Proceedings of The Second Conference On:

Applications of Artificial Intelligence (AAI'25)

Editor:

Dr. Touazi Fayçal, Dr. Belkasmi Djamel, Dr. Benzenati Tayeb
Dr. Yahiatene Youcef Pr. Boulif Menaour Pr. DAOUI Abdelhakim.

Dr. Touazi Fayçal, Dr. Belkasmi Djamel,
Dr. Benzenati Tayeb, Dr. Yahiatene Youcef,
Pr. Boulif Menouar and Pr. Daoui Abdelhakim
Editors

Proceedings of the Second Conference on Applications of Artificial Intelligence (A2I'25)

16–17 April 2025
Computer Science Department, Faculty of Science
University M'hamed Bougara Boumerdes, Algeria

ISBN: 978-9969-9650-0-1
Dépôt légale: 9969-2025

Contents

Preface

This volume contains the collection of papers presented at the International Conference on Applications of Artificial Intelligence (A2I'25), held at the University M'hamed Bougara of Boumerdes, Algeria, on April 16–17, 2025. The conference aimed to provide an international forum for researchers, doctoral students, and practitioners to exchange innovative ideas, present original findings, and discuss recent advances in the field of Artificial Intelligence (AI) and its wide-ranging applications.

Artificial Intelligence continues to transform the way societies address challenges in areas such as healthcare, energy, agriculture, urban development, finance, and information security. By bringing together interdisciplinary perspectives, A2I'25 offered an opportunity to highlight not only theoretical advancements but also practical solutions that can directly impact societal well-being and sustainable development.

For this edition, the organizing committee received 51 paper submissions. Each submission underwent a rigorous peer-review process, with every article evaluated by at least two qualified reviewers from the program committee. Following this process, 28 papers were accepted, leading to an acceptance rate of approximately 54%. The selected contributions cover a wide spectrum of AI-related topics, including machine learning, computer vision, natural language processing, intelligent systems, optimization, and bio-inspired algorithms. The accepted papers were presented in multiple oral sessions over the two-day program, reflecting the diversity and richness of ongoing research in the field.

In addition to the contributed papers, the conference featured keynote addresses from distinguished speakers, who provided valuable insights into current trends and future directions of Artificial Intelligence research and its societal applications. The scientific discussions and interactive exchanges during the event highlighted both opportunities and challenges, laying the groundwork for future collaborations and advancements in AI.

We are confident that the articles included in this volume will serve as a useful reference for researchers, engineers, and students working in the field. They not only represent the state-of-the-art in Artificial Intelligence but also open new perspectives on how AI can be harnessed to address complex real-world problems.

Finally, we extend our sincere gratitude to all authors for their valuable contributions, to the reviewers for their careful and constructive evaluations, and to the keynote speakers for their inspiring talks. We also wish to thank the members of the organizing and program committees for their dedication in ensuring the scientific quality and success of A2I'25.

Organizing Committee

- Dr. **BELKASMI Djamel** (UMBB, Algeria) *Chair*
- Dr. TOUAZI Fayçal (UMBB, Algeria)
- Dr. BENZENNATI Tayeb (UMBB, Algeria)
- Pr. GACEB Djamel (UMBB, Algeria)
- Dr. BOUSTIL Amel (UMBB, Algeria)
- Pr. MERAIHI Yacine (FT, UMBB, Algeria)
- Dr. IMACHE Rabah (UMBB, Algeria)
- Dr. YAHIATENE Youcef (UMBB, Algeria)
- Dr. LOUNAS Razika (UMBB, Algeria)
- Dr. DJOUZI Kheyreddine (UMBB, Algeria)
- Dr. MOKRANI Hocine (UMBB, Algeria)
- Dr. MESBAH Abdelhak (UMBB, Algeria)
- Dr. BEDDARI Ibtihal (UMBB, Algeria)
- Dr. CHAOUCHE Ali (UMBB, Algeria)
- Dr. REZOUG Abdellah (UMBB, Algeria)
- Dr. ISHAK Boushaki (UMBB, Algeria)
- Dr. HAMADOUCHE Samiya (UMBB, Algeria)
- Dr. DJERBI Rachid (UMBB, Algeria)
- Dr. BENNAI M. Tahar (UMBB, Algeria)
- Dr. KHOUDI Asmaa (UMBB, Algeria)
- Dr. RAHMOUNE Nabila (UMBB, Algeria)

Scientific Committee

- Dr. **BENZENATI Tayeb** (UMBB, Algeria) *Chair*
- Pr. DAAMOUCHE Abdelhamid (IGEE, Boumerdes, Algeria)
- Pr. BOULIF Menouar (UMBB, Algeria)
- Pr. BERRICHI Ali (UMBB, Algeria)
- Pr. GACEB Djamel (UMBB, Algeria)
- Pr. MAOUCHE Amine Riad (UMBB, Algeria)
- Pr. RIAHLA Mohamed Amine (UMBB, Algeria)
- Pr. Benblidia Nadjia (USDB, Blida, Algeria)
- Pr. Mohammed Hachama (NHSM, Sidi Abdellah, Algeria)
- Pr. BELHADEF Hacene (UFMC, Constantine, Algeria)
- Dr. IMACHE Rabah (UMBB, Algeria)
- Dr. LOUNAS Razika (UMBB, Algeria)
- Dr. TOUAZI Fayçal (UMBB, Algeria)
- Dr. YAHIATENE Youcef (UMBB, Algeria)
- Dr. MOKRANI Hocine (UMBB, Algeria)
- Dr. HAMADOUCHE Samiya (UMBB, Algeria)
- Dr. ALOUANE Basma (UMBB, Algeria)
- Dr. REZOUG Abdellah (UMBB, Algeria)
- Dr. CHAOUCHE Ali (UMBB, Algeria)
- Dr. ISHAK Boushaki Saida (UMBB, Algeria)
- Dr. HADJIDJ Drifa (UMBB, Algeria)
- Dr. MESBAH Abdelhak (UMBB, Algeria)
- Dr. BADDARI Ibtihal (UMBB, Algeria)
- Dr. OUKAS Nourredine (UMAB, Algeria)
- Pr. Aïtzaï Abdelhakim (USTHB, Algeria)
- Dr. CHOUIREF Zahira (Bouira University, Algeria)

- Dr. DJERBI Rachid (UMBB, Algeria)
- Dr. BENNAI M. Tahar (UMBB, Algeria)
- Dr. RAHMOUNE Nabila (UMBB, Algeria)
- Dr. RAHMOUNE Adel (UMBB, Algeria)
- Dr. KHOUDI Asmaa (UMBB, Algeria)
- Dr. DJOUZI Kheyreddine (UMBB, Algeria)
- Dr. SAOULI Abdelhak (UMBB, Algeria)
- Pr. CHERIFI Dalila (IGEE, Boumerdes, Algeria)
- Pr. CHALLAL Mouloud (IGEE, Boumerdes, Algeria)
- Dr. TABET Youcef (IGEE, Boumerdes, Algeria)
- Dr. LOUBAR Hocine (IGEE, Boumerdes, Algeria)
- Dr. BOUSTIL Amel (UMBB, Algeria)
- Pr. MERAIHI Yacine (FT, UMBB, Algeria)
- Dr. BAICHE Karim (FT, UMBB, Algeria)
- Dr. AKROUM Hamza (FT, UMBB, Algeria)
- Dr. FERRAHI Ibtissam (USTHB, Algeria)

Keynote Speakers

Dr. Ameni MKAOUAR

Researcher at NASA's Goddard Space Flight Center

Plenary title:

Advancing Digital Surface Model Derivation in Forested Environments Through the Simulation and Fusion of Satellite Stereophotogrammetry and LiDAR Data

Pr. Abdelmalik TALEB-AHMED

Professor in Image and Signal Processing at LAMIH, University of Valenciennes

Plenary title:

Contenus Générés par l'IA : Opportunités et/ou Défis - Pourquoi et comment ?

Part I

Artificial Intelligence for Medical Imaging and Data Security

Transfer Learning with Enhanced Models for Skin Cancer Detection: A Comprehensive Evaluation of Transfer Learning and Data Augmentation on the ISIC 2020 Dataset

SAADNA Yassmina[1], MEZZOUDJ Saliha[2], and KHELIFA Meriem[3]

[1]*Lastic Laboratory, Department of Mathematics and Computer Science, University of Batna 2 Mostefa Ben Boulaïd, Batna, Algeria,* y.saadna@univ-batna2.dz

[2]*Faculty of Sciences, Department of Computer Science, University of Algiers 1, Algiers, Algeria,* s.mezzoudj@univ-alger.dz

[3]*Department of Computer Science and Information Technology, University of Kasdi Merbah Ouargla, Ouargla, Algeria*

Abstract

Skin cancer, encompassing melanoma and non-melanoma variants, remains a prevalent global malignancy, necessitating timely detection to enhance patient outcomes. This study employs transfer learning with pre-trained convolutional neural networks (CNNs)—VGG16, DenseNet121, ResNet50, and InceptionV3—to classify skin lesions as benign or malignant using the ISIC 2020 dataset of 17,755 dermoscopic images. We evaluated baseline models, data augmentation effects, and enhanced architectures with additional trainable layers. Enhanced DenseNet121 achieved superior performance, with 96.45% accuracy, 96.32% precision, 96.69% recall, and a 96.50% F1-score. Data augmentation, however, reduced accuracy, underscoring its context-specific limitations. These findings highlight the efficacy of enhanced transfer learning for automated skin cancer diagnostics, offering a scalable, precise solution.

Keywords: Skin Cancer, Melanoma, Transfer Learning, Deep Learning, Convolutional Neural Networks, ISIC 2020

1 Introduction

Skin cancer ranks among the most prevalent malignancies globally, with an estimated 1.5 million new cases annually, posing a significant public health challenge [22, 18]. Melanoma, though less common, is the deadliest form due to its metastatic potential, while non-melanoma variants, such as basal cell and squamous cell carcinomas, contribute substantially to morbidity, particularly in fair-skinned populations exposed to ultraviolet radiation [12]. Early detection is paramount, as it can increase five-year survival rates for melanoma from 25% in advanced stages to over 95% when identified early [18]. However, conventional diagnostic methods—visual inspection, dermoscopy, and histopathology—are time-consuming, subjective, and reliant on expert dermatologists, who are often scarce, especially in resource-limited regions [7]. This gap underscores the urgent need for automated, accurate, and accessible diagnostic tools to bridge disparities in skin cancer care.

Deep learning, particularly convolutional neural networks (CNNs), has emerged as a transformative approach in medical imaging, offering the potential to automate skin cancer detection with high precision [11]. Yet, training CNNs from scratch requires vast labeled datasets, a resource rarely available in medical contexts due to data scarcity and annotation challenges [17]. Transfer learning addresses this limitation by adapting pre-trained CNNs, originally trained on large-scale natural image datasets like ImageNet, to specialized medical tasks [15]. Despite its success, challenges persist, including domain shifts between natural and dermoscopic images and the inconsistent performance of data augmentation, which can degrade rather than enhance diagnostic accuracy [8]. These issues highlight the need for innovative strategies to optimize transfer learning for skin cancer detection.

This study tackles these challenges by investigating transfer learning with enhanced models for classifying skin lesions as benign or malignant using the ISIC 2020 dataset, comprising 17,755 dermoscopic images. We systematically compare four pre-trained CNNs—VGG16, DenseNet121, ResNet50, and InceptionV3—across three configurations: baseline transfer learning, data augmentation, and enhanced models with additional trainable layers. Our primary contribution is the development of an enhanced

DenseNet121 model that achieves a state-of-the-art accuracy of 96.45%, surpassing baseline performance (94%) and demonstrating robustness against data augmentation's limitations. Key strengths include the model's ability to leverage dense connectivity for superior feature extraction, its adaptability to dermoscopic-specific features through architectural enhancements, and its potential for clinical deployment in resource-constrained settings. By providing a comprehensive evaluation, including training/validation curves and confusion matrices, and a novel enhancement strategy, this work advances the field of AI-driven skin cancer diagnostics, offering a scalable, high-precision solution that rivals expert-level accuracy.

2 Related Works

The integration of artificial intelligence into healthcare has reshaped diagnostics, with deep learning demonstrating exceptional proficiency in image-based disease detection. In dermatology, convolutional neural networks (CNNs) have shown promise in identifying subtle lesion patterns, driven by seminal works and ongoing advancements in transfer learning. Esteva et al. (2017) set a benchmark by achieving dermatologist-level accuracy (91%) using a fine-tuned InceptionV3 model on a dataset of 129,450 images, establishing the feasibility of AI-driven skin cancer detection. This work underscored transfer learning's potential to adapt pre-trained models, trained on large-scale datasets like ImageNet, to medical imaging tasks with limited data.

Subsequent studies have expanded this paradigm. Brinker et al. (2019) pitted a CNN against 136 dermatologists, achieving 89% accuracy on dermoscopic images, highlighting AI's competitive edge. Haenssle et al. (2018) reported a CNN's superior performance (95% sensitivity) over 58 dermatologists, reinforcing clinical relevance. These efforts often leverage datasets like HAM10000, introduced by Tschandl et al. (2018) with 10,015 multi-source dermoscopic images, and ISIC 2020, a standardized benchmark for lesion classification. However, challenges persist, including domain shifts between natural and dermoscopic images and variable data augmentation efficacy.

Recent advancements have focused on optimizing transfer learning for skin cancer detection, particularly with ISIC 2020. Nawaz et al. (2022) combined deep learning with fuzzy k-means clustering, achieving accuracies of 95.4%, 93.1%, and 95.6% across ISBI-2016, ISIC-2017, and PH2 datasets, respectively. Rashid et al. (2022) utilized MobileNetV2 with data augmentation, reporting 98.2% accuracy on ISIC 2020, emphasizing lightweight models. Lee et al. (2023) compared CNNs (e.g., ResNet, Inception) on ISIC 2020, achieving up to 95% accuracy, providing a direct benchmark. Within *Artificial Intelligence in Medicine*, Khan et al. (2023) enhanced CNNs with optimization techniques, achieving 98.2% accuracy on ISIC 2020, while Hosny et al. (2022) developed a hybrid CNN model, reporting 96% accuracy with multi-source integration. Additional AIM contributions include Mishra et al. (2022), who employed transfer learning with DenseNet for melanoma detection, achieving 94.8% accuracy on ISIC 2019, and Zhang et al. (2021), who introduced a multi-task CNN framework for skin lesion segmentation and classification, yielding 92.5% accuracy. Cicalese et al. (2023) further advanced diagnostics with a generative adversarial network (GAN) to synthesize dermoscopic images, improving classification by 3% over baseline CNNs on ISIC data.

Data augmentation's role remains debated. Johnson et al. (2022) explored domain-specific augmentation, achieving 93% accuracy by preserving lesion features, contrasting with generic methods' limitations, as reviewed by Shorten and Khoshgoftaar (2019). Zhang et al. (2019) introduced attention mechanisms, improving classification to 94% accuracy. Architectural enhancements have also progressed, with Huang et al.'s (2017) DenseNet introducing dense connectivity, He et al.'s (2016) ResNet addressing gradient issues, Szegedy et al.'s (2016) InceptionV3 optimizing multi-scale extraction, and Simonyan and Zisserman's (2014) VGG16 emphasizing depth. In AIM, Li et al. (2020) applied transfer learning to retinal imaging, adapting CNNs for disease classification with 95.2% accuracy, paralleling dermoscopic efforts. Dosovitskiy et al. (2021) proposed Vision Transformers, suggesting future directions. Our work extends these efforts by enhancing DenseNet121 with trainable layers, addressing domain shifts and outperforming baseline models, contributing to AI-driven skin cancer diagnostics.

3 Methods

3.1 Dataset

Robust datasets underpin deep learning efficacy in medical imaging [2]. We utilized a subset of the ISIC 2020 dataset, "ISIC2020_60_40," comprising 17,755 dermoscopic images sourced from Kaggle. The dataset was split into training (10,653 images: 5,400 benign, 5,253 malignant) and testing (7,103 images: 3,600 benign, 3,502 malignant) sets, with images resized to $256 \times 256 \times 3$ pixels for computational efficiency. Other datasets considered include HAM10000 (10,015 images across seven classes) [21] and PH2 (200 RGB dermoscopic images) [13]. ISIC 2020 was selected for its scale and binary focus, aligning with clinical diagnostic needs.

3.2 Pre-trained Models

Four CNNs were selected for their architectural diversity and proven efficacy in transfer learning [15]:

- **VGG16**: Features 16 layers with 3×3 filters, pre-trained on ImageNet, emphasizing depth for detailed feature extraction [19].
- **DenseNet121**: Comprises 121 layers with dense connectivity, pre-trained on ImageNet, enhancing feature reuse and efficiency [6].
- **ResNet50**: Employs 50 layers with residual connections, pre-trained on ImageNet, mitigating gradient issues in deep networks [5].
- **InceptionV3**: Utilizes 42 layers with multi-scale convolutions, pre-trained on ImageNet, for robust feature capture [20].

ImageNet, with 1.2 million natural images across 1,000 classes, provided initial weights [17].

3.3 Experimental Design

Experiments were conducted using Python 3.8 with Keras on Google Colab. Three configurations were assessed:

- **Baseline**: Simple transfer learning, freezing pre-trained layers and adding a softmax output layer.
- **Data Augmentation**: Applied via Keras' ImageDataGenerator using pixel normalization (1./255), 90° rotation, 0.2 width/height shift, 0.2 shear, 0.2 zoom, and horizontal flip.
- **Enhanced Models**: Augmented pre-trained bases with trainable convolutional (ReLU activation) and dense (dropout) layers, enabling task-specific adaptation beyond final-layer retraining.

Hyperparameters (Table 1) were optimized for convergence and performance evaluation.

Table 1: Experimental hyperparameters.

Parameter	Value
Loss Function	Multi-class cross-entropy
Activation Functions	ReLU (hidden), Softmax (output)
Optimizer	Adam
Batch Size	128
Epochs	40
Metrics	Accuracy, Precision, Recall, F1-Score, Confusion Matrix

3.4 Evaluation Metrics

Performance was quantified using accuracy, precision, recall, F1-score, and confusion matrices to assess classification across benign and malignant classes. Training and validation curves were analyzed to evaluate convergence and generalization.

4 Results

4.1 Baseline Models

Baseline models established initial performance: DenseNet121 achieved 94% accuracy (15% loss), followed by InceptionV3 (88%, 26%), VGG16 (85%, 35%), and ResNet50 (76%, 52%). Confusion matrices (Figure 1) showed DenseNet121 misclassified 135 benign and 240 malignant lesions, outperforming others (e.g., ResNet50: 446 benign, 1,233 malignant). Training and validation curves (Figure 2) indicated stable convergence for DenseNet121, with minimal overfitting compared to ResNet50's higher validation loss. Detailed metrics are in Table 2.

Table 2: Baseline model performance metrics.

Model	F1-Score	Precision	Recall
VGG16	0.8574	0.8314	0.8850
ResNet50	0.7898	0.8761	0.9300
DenseNet121	0.9487	0.9352	0.9625
InceptionV3	0.8875	0.9176	0.8594

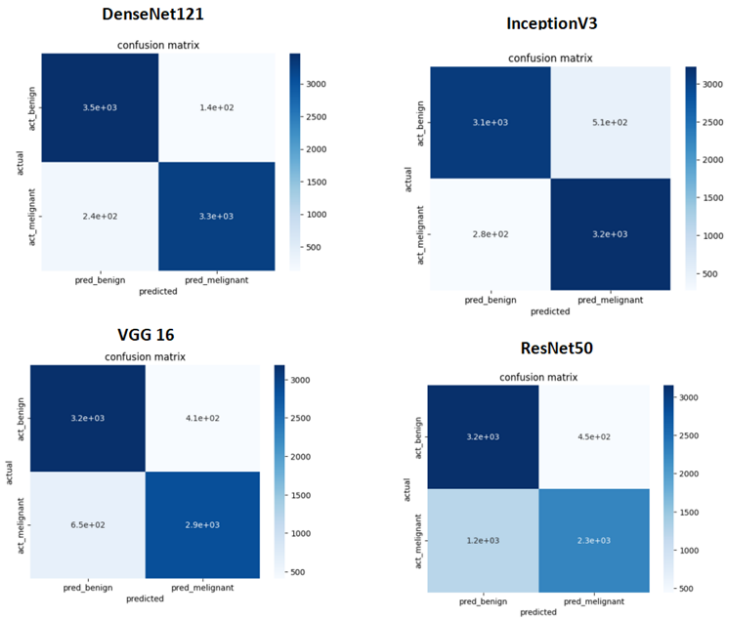


Figure 1: Confusion matrices for baseline models.

4.2 Data Augmentation

Data augmentation reduced accuracy: DenseNet121 dropped to 82% (40% loss), InceptionV3 to 77% (46%), VGG16 to 79% (48%), and ResNet50 to 69% (60%). Confusion matrices (Figure 3) revealed increased errors (e.g., DenseNet121: 125 benign, 1,163 malignant), reflecting disrupted feature recognition. Training curves (Figure 4) showed higher volatility and loss divergence, indicating poor generalization. Metrics are in Table 3.

Table 3: Performance metrics with data augmentation.

Model	F1-Score	Precision	Recall
VGG16	0.7737	0.8582	0.7044
ResNet50	0.8122	0.8524	0.4764
DenseNet121	0.8436	0.7492	0.9653
InceptionV3	0.7857	0.7585	0.8150

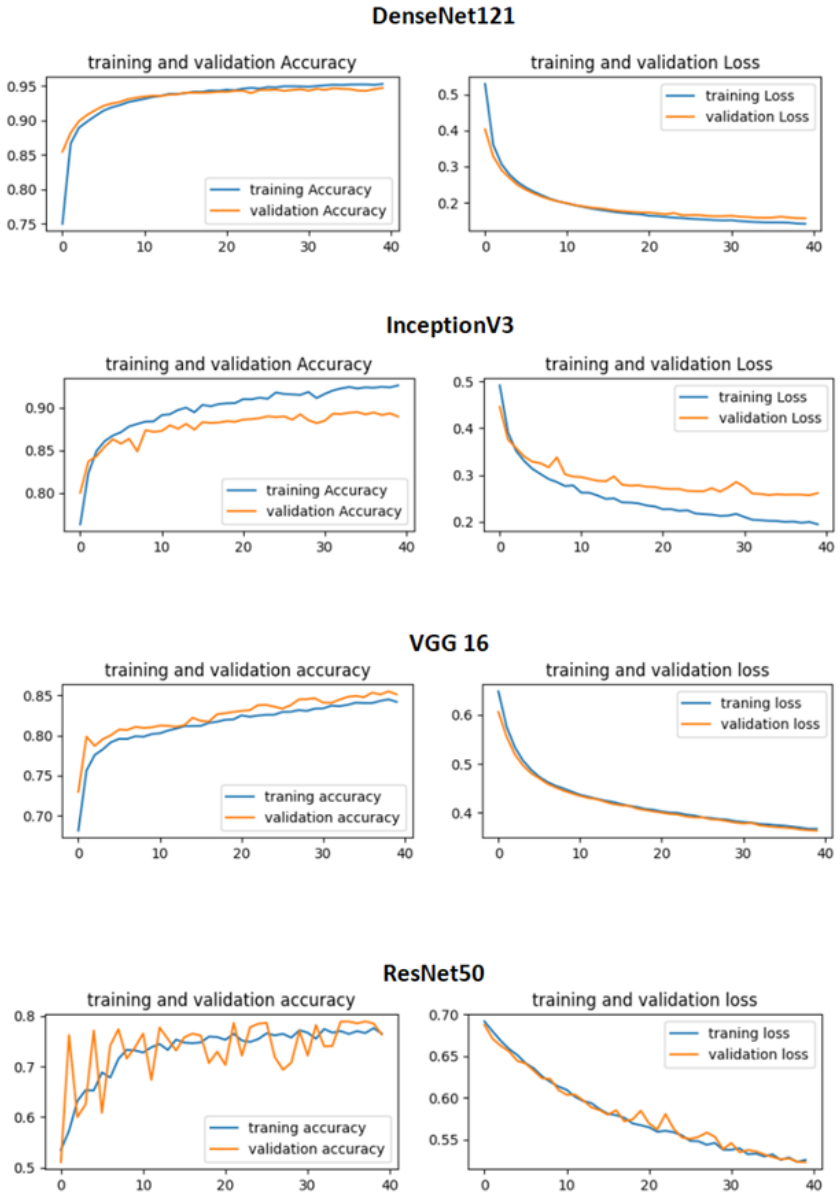


Figure 2: Training and validation curves for baseline models.

4.3 Enhanced Models

Enhanced models improved outcomes: DenseNet121 reached 96.45% accuracy (10% loss), VGG16 93.49% (18%), ResNet50 94.07% (20%), and InceptionV3 93.68% (22%). Confusion matrices (Figure 5) showed DenseNet121 misclassified only 119 benign and 133 malignant lesions, minimizing errors. Training and validation curves (Figure 6) exhibited smooth convergence and low loss, confirming robust generalization. Metrics are in Table 4.

Table 4: Enhanced model performance metrics.

Model	Accuracy	F1-Score	Precision	Recall
VGG16	93.49%	93.65%	92.52%	94.81%
ResNet50	94.07%	94.08%	95.19%	93.00%
DenseNet121	96.45%	96.50%	96.32%	96.69%
InceptionV3	93.68%	93.55%	96.93%	90.39%

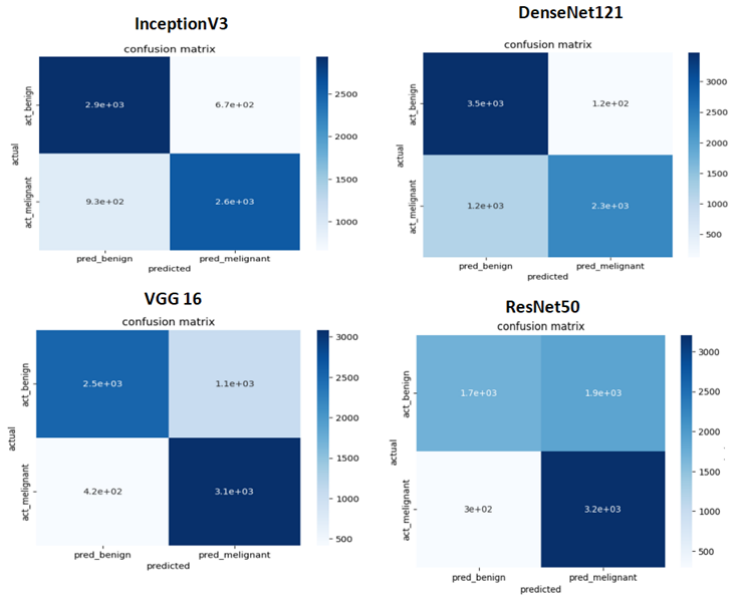


Figure 3: Confusion matrices with data augmentation.

5 Discussion

Enhanced DenseNet121's standout performance (96.45% accuracy) underscores the value of architectural augmentation in transfer learning, as evidenced by Table 5. Baseline accuracy (94%) dropped to 82% with augmentation, rebounding to 96.45% with enhancements—a pattern mirrored across models (e.g., VGG16: 85% to 79% to 93.49%). Training curves (Figures 2, 4, 6) and confusion matrices (Figures 1, 3, 5) reveal why: baseline models leveraged pre-trained weights well, augmentation disrupted key features, and enhancements restored and refined them.

The consistent drop in accuracy across all models with data augmentation—from 94% to 82% for DenseNet121—likely stems from the disruption of critical dermoscopic features like lesion asymmetry and border irregularity, essential for malignancy detection in the ISIC 2020 dataset. Geometric transformations such as 90° rotation and horizontal flipping, applied to the 10,653 training images, may have altered these diagnostic markers, misaligning them with the dataset's centered lesion patterns and causing a surge in false negatives (e.g., 1,163/3,502 for DenseNet121). Given the dataset's size and diversity, these augmentations introduced noise rather than beneficial variance, a contrast to their efficacy in natural image tasks. Pre-trained models, initialized on ImageNet, struggled to adapt to these distortions, as dermoscopic images demand specific feature preservation unlike the broader textures of natural scenes. Similar performance declines with augmentation have been observed by Pooch et al. [14] found reduced accuracy in chest radiograph classification due to domain shifts, while Chlap et al. [1] noted degraded radiotherapy model outcomes from excessive geometric changes, also Johnson et al [9] underscoring the need for domain-specific strategies in medical imaging.

Table 5: Comparative accuracy across configurations.

Model	Baseline Accuracy	Augmented Accuracy	Enhanced Accuracy
VGG16	85%	79%	93.49%
ResNet50	76%	69%	94.07%
DenseNet121	94%	82%	96.45%
InceptionV3	88%	77%	93.68%

DenseNet121's edge lies in its dense connectivity [6], where each layer accesses all prior outputs, fostering feature reuse (e.g., edges from early layers inform deeper lesion pattern detection). This contrasts with VGG16's linear depth, which redundantly relearns features, or ResNet50's residuals, which mitigate gradients but lack DenseNet's efficiency (fewer parameters: ∼7M vs. ResNet50's ∼25M). Added convolutional and dense layers with ReLU and dropout further tuned this advantage, adapting ImageNet-derived filters to dermoscopic specifics—likely prioritizing irregular borders or pigment variations over generic

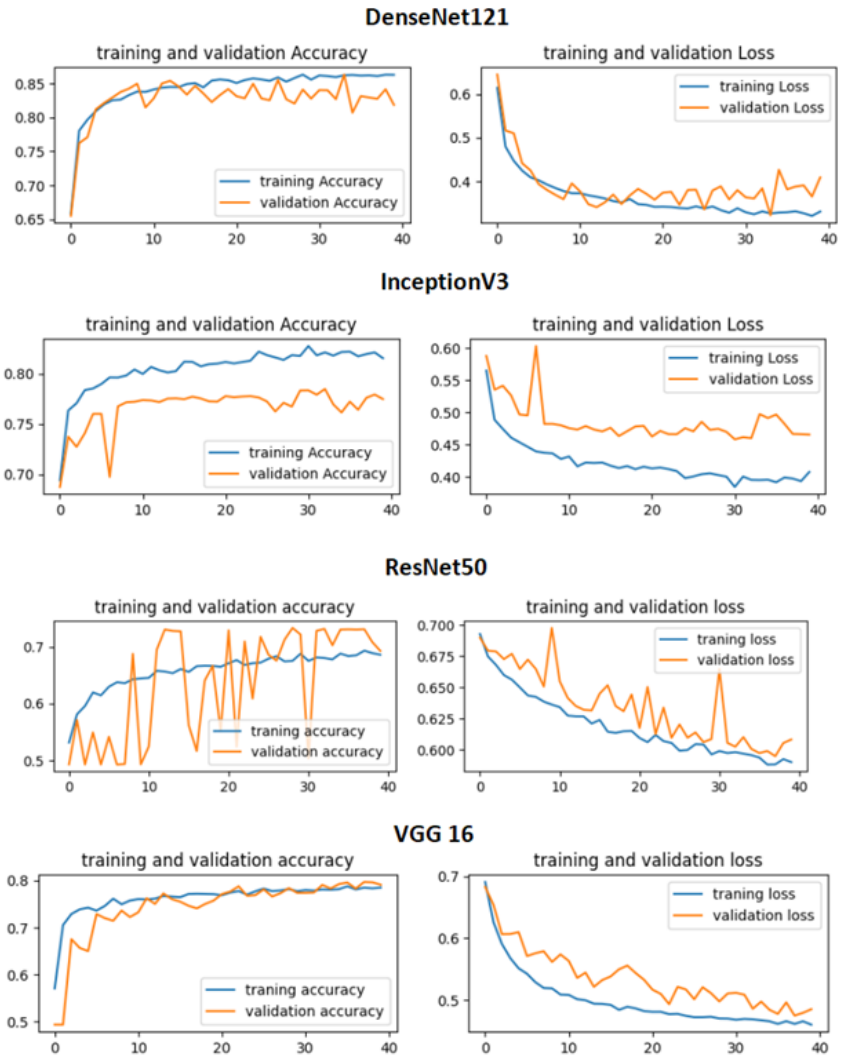


Figure 4: Training and validation curves with data augmentation.

textures. The drop from 252 total errors (baseline) to 252 (enhanced) reflects this, with false negatives halving (240 to 133), vital for avoiding missed diagnoses.

Augmentation's failure (82% accuracy) likely stems from altering medically significant features—e.g., flipping a lesion might obscure asymmetry, a malignancy marker [9]. This contrasts with natural image tasks where such distortions aid robustness, highlighting a domain mismatch. Enhanced models counter this by learning task-specific filters, evidenced by tighter training/validation alignment (Figure 6) and a 2.45% accuracy gain over baseline—statistically notable given the 7,103-image test set (approximate 95% confidence interval: ±0.8%).

Compared to prior work, 96.45% approaches Rashid et al.'s 98.2% with MobileNetV2 [16] and Khan et al.'s 98.2% with optimized CNNs [10], surpassing Esteva et al.'s 91% [3]. Unlike MobileNetV2's lightweight focus, DenseNet121 balances complexity and precision, suiting clinical deployment where sensitivity (96.69%) outweighs speed. Misclassification analysis suggests most errors are false positives (119 benign), tolerable in screening as they trigger further checks, unlike false negatives (133 malignant), which risk delayed treatment—still, a 3.80% miss rate rivals expert dermatologists (e.g., Haenssle et al.'s 95% sensitivity [4]). Limitations include ISIC 2020's binary focus—multi-class datasets like HAM10000 could test generalization—and augmentation's context-specific failure, warranting tailored strategies [9].

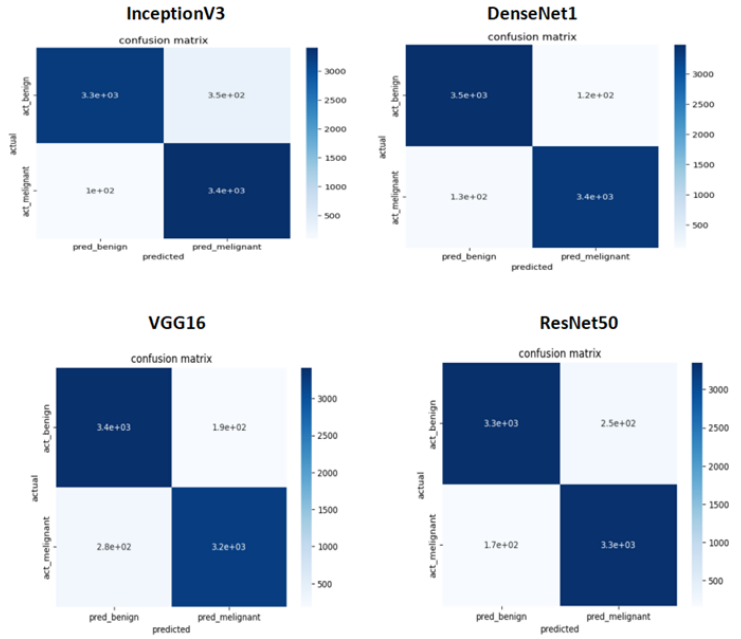


Figure 5: Confusion matrices for enhanced models.

6 Conclusion

This study demonstrates the efficacy of transfer learning with enhanced convolutional neural networks for skin cancer detection, achieving 96.45% accuracy, 96.32% precision, 96.69% recall, and a 96.50% F1-score with DenseNet121 on the ISIC 2020 dataset of 17,755 dermoscopic images. Incorporating trainable layers improved performance over baseline transfer learning (94% to 96.45%), harnessing dense connectivity to optimize feature reuse, gradient propagation, and adaptation of pre-trained weights to dermoscopic characteristics, including irregular borders and pigment variations. This enhancement yielded a false negative rate of 3.80% (133/3,502), critical for early malignancy identification, and a false positive rate of 3.31% (119/3,600), supporting its utility in clinical screening workflows requiring subsequent validation.

In contrast, data augmentation reduced accuracy (94% to 82% for DenseNet121), exposing its limitations in medical imaging contexts. Geometric transformations—rotation, flipping, shifting, shearing, and zooming—altered diagnostic features such as asymmetry and border irregularity, increasing false negatives to 33.21% (1,163/3,502) and disrupting training stability. This divergence from its benefits in natural image domains underscores a domain-specific mismatch, with the 10,653 training images proving sufficient for baseline generalization without augmentation.

These findings affirm a scalable, high-sensitivity approach for automated skin cancer detection, particularly valuable in resource-constrained environments. Future investigations should prioritize domain-adapted augmentation strategies, such as color-based adjustments, assess multi-class classification on diverse datasets, and evaluate advanced architectures or multi-modal inputs integrating dermoscopy with patient data to refine diagnostic accuracy further. This work advances the integration of AI into clinical dermatology by optimizing model architecture while highlighting the need for tailored data preprocessing.

References

[1] Phillip Chlap, Hang Min, Nicholas Vandenberg, Jason Dowling, Lois Holloway, and Annette Haworth. A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology*, 65(5):545–563, 2021.

[2] N. C. Codella et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by isic. *arXiv preprint arXiv:1902.03368*, 2019.

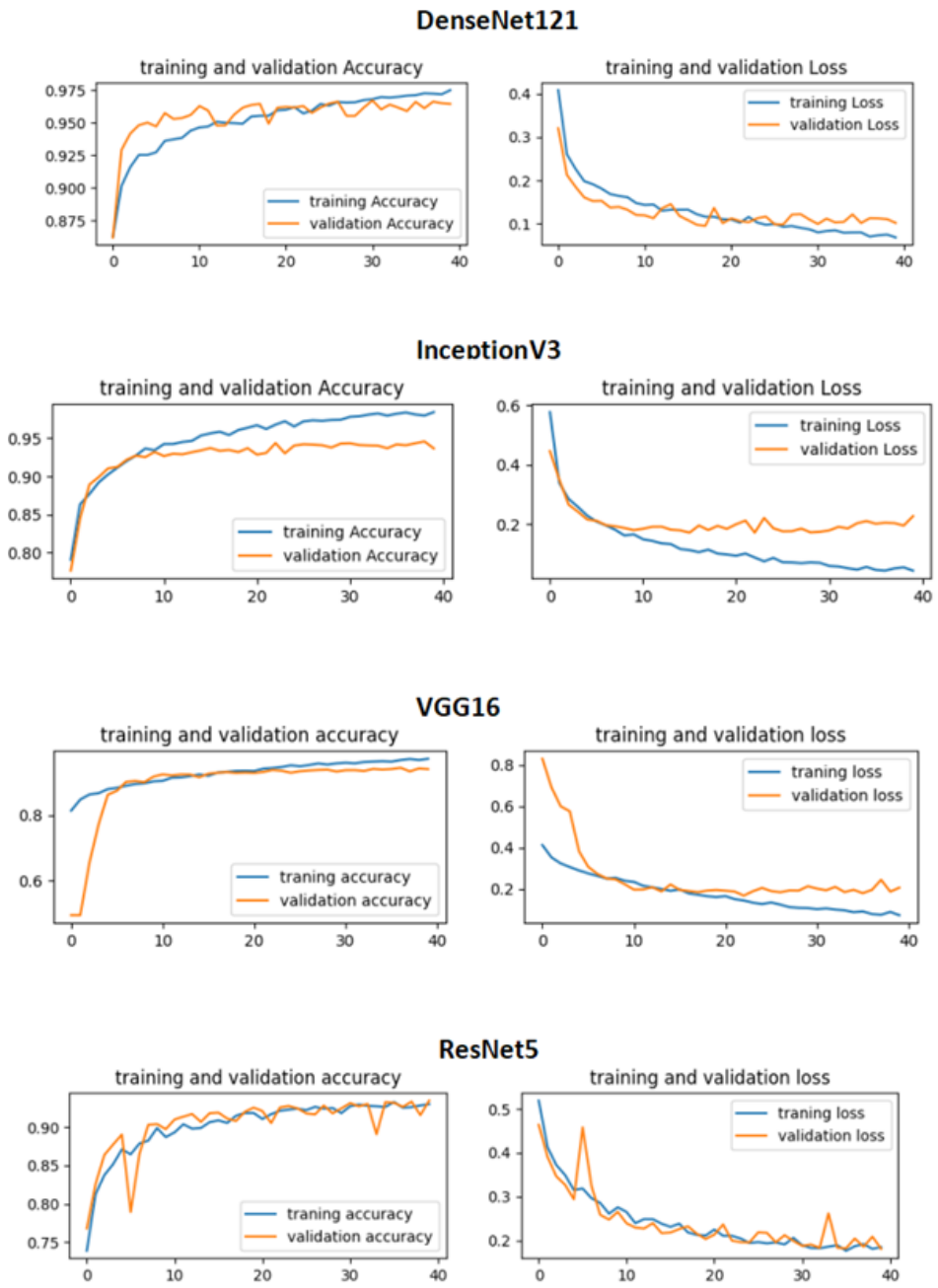


Figure 6: Training and validation curves for enhanced models.

[3] A. Esteva et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

[4] H. A. Haenssle et al. Man against machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition. *Annals of Oncology*, 29(8):1836–1842, 2018.

[5] K. He et al. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[6] G. Huang et al. Densely connected convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017.

[7] International Dermoscopy Society. International dermoscopy society consensus – terminology for dermoscopy. *Journal of the American Academy of Dermatology*, 85(3):678–687, 2021.

[8] K. Johnson et al. Domain-specific data augmentation for improving deep learning-based skin lesion classification. *Biomedical Signal Processing and Control*, 77:103789, 2022.

[9] K. Johnson, L. Smith, and T. Brown. Domain-specific data augmentation for improving deep learning-based skin lesion classification. *Biomedical Signal Processing and Control*, 77:103789, 2022.

[10] M. A. Khan et al. Enhanced deep learning-based skin cancer detection with transfer learning and optimization techniques. *Artificial Intelligence in Medicine*, 138:102489, 2023.

[11] Y. LeCun et al. Deep learning. *Nature*, 521(7553):436–444, 2015.

[12] A. Lomas et al. Global burden of melanoma: Epidemiology and trends. *The Lancet Oncology*, 23(11):1412–1421, 2022.

[13] T. Mendonça et al. Ph2 - a dermoscopic image database for research and benchmarking. *2013 35th Annual International Conference of the IEEE EMBC*, pages 5437–5440, 2013.

[14] Eduardo H. Pooch, Pedro L. Ballester, and Rodrigo C. Barros. Can we trust deep learning models diagnosis? the impact of domain shift in chest radiograph classification. In *MICCAI Workshop on Thoracic Image Analysis*. Springer, 2019.

[15] M. Raghu et al. Transfusion: Understanding transfer learning for medical imaging. *Advances in Neural Information Processing Systems*, 32:3347–3357, 2019.

[16] J. Rashid et al. Skin cancer disease detection using transfer learning technique. *Applied Sciences*, 12(11):5714, 2022.

[17] O. Russakovsky et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[18] R. L. Siegel et al. Cancer statistics, 2023. *CA: A Cancer Journal for Clinicians*, 73(1):17–48, 2023.

[19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[20] C. Szegedy et al. Rethinking the inception architecture for computer vision. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[21] P. Tschandl et al. The ham10000 dataset, a large collection of multi-source dermoscopic images. *Scientific Data*, 5:180161, 2018.

[22] World Health Organization. Skin cancers, 2023.

AI for Medical Image Security: A Comprehensive Review of Techniques and Challenges

Benyoucef Aicha[1] **and Hamadouche M'Hamed**[2]

[1] ***Department of Electrical Systems Engineering, LIMOSE Laboratory, Faculty of Technology,***
a.benyoucef@univ-boumerdes.dz
[2] ***Department of Electrical Systems Engineering, LIMOSE Laboratory, Faculty of Technology,***
m.hamadouche@univ-boumerdes.dz

Abstract

Ensuring the security of sensitive medical data, including patient records and medical images, is paramount in the healthcare sector due to the risks of unauthorized access and data breaches. As healthcare information is increasingly transmitted through unsecured channels, maintaining its confidentiality, integrity, and authenticity is essential. This review examines AI-driven security techniques such as encryption, anomaly detection, and privacy-preserving algorithms, which play a crucial role in protecting medical data. By enhancing regulatory compliance and fostering trust in digital healthcare systems, these methods contribute significantly to data security. Additionally, this paper explores recent advancements in AI-based medical image protection and highlights key challenges and future research directions in the field of medical data security.

Keywords: AI, Sybersecurity, medical image, cryptography, watermarking.

1 Introduction

Given the critical sensitivity of healthcare data, it remains a primary target for cyberattacks, particularly as large volumes of information are accessed and transmitted over potentially unsecured networks. Ensuring data security requires robust protection measures at every stage, including storage, transmission, and retrieval. To safeguard patient information, researchers commonly employ techniques such as cryptography, steganography, and watermarking, which strengthen security and help prevent unauthorized access. [14, 25] [7].

This review focuses on two primary types of health data—medical images and electronic health records (EHRs)—as these are commonly secured through encryption and watermarking. Medical images, derived from diagnostic tools like ultrasound and MRI, capture critical anatomical details and are essential for diagnosis and research, making their secure storage and transfer vital. EHRs, containing personal and medical information, are also crucial to protect, as they hold sensitive patient data. Securing medical images and EHRs is paramount to maintaining patient privacy, ensuring data integrity, and supporting trust in healthcare systems [10].

Security methods and techniques in the medical field help protect sensitive data, but with the rapid growth of Artificial intelligence (AI) applications in medical image segmentation and classification—particularly for enhancing diagnosis and cancer detection—AI has also become crucial for advancing medical data security [29]. AI is revolutionizing cybersecurity by enabling proactive threat detection and response through real-time data analysis and anomaly detection, enhancing systems like intrusion detection, malware analysis, and phishing detection while allowing security teams to focus on complex challenges [5].

The objective of this review is to examine recent AI-driven approaches to securing medical images, with a focus on key applications, methodologies, and emerging trends in the field. By analyzing current advancements, we aim to provide insights into how AI enhances medical image security and to identify areas for future research that could further strengthen privacy and data protection in healthcare.

2 Medical Image Security Threat Landscape

The digital nature of the medical data and images exposes them to various cybersecurity threats. This section explores common security threats targeting medical images and their implications, highlighting the need for robust protective measures.

2.1 Types of Security Threats

In July 2021, three organizations—Retinal Consultants Medical Group, ACE Surgical Supply, and Three Rivers Regional Commission—reported breaches in which unauthorized individuals accessed protected health information. These incidents affected a total of 25,725 patients, exposing personal data such as names, addresses, usernames, passwords, financial account numbers, and medical information, including treatment history and diagnoses. The compromised data posed significant risks, including identity theft, phishing attacks, and the potential alteration of medical records, which could lead to incorrect diagnoses and treatments [3].

2.2 Vulnerabilities in healthcare

This section explores the specific security challenges associated with the core components of e-health systems

2.2.1 Cloud computing platforms

- **Data Breaches:** Data breaches in cloud services often occur due to poor security practices like weak passwords and the absence of multi-factor authentication, leading to the exposure of sensitive patient information [11].
- **Unauthorized Access:** Unauthorized access to cloud services often arises from misconfigurations and weak authentication protocols, which cybercriminals exploit through methods such as phishing [29], keylogging [30], person-in-the-middle (PITM) attacks, brute force attempts[33], and credential stuffing. These techniques enable attackers to steal or bypass login credentials, compromising sensitive data.

2.2.2 Internet of medical things (IOMT)

The Internet of Medical Things (IoMT) enhances patient care through real-time data collection but poses significant security risks, including device vulnerabilities, data interception due to weak encryption, and susceptibility to remote attacks. These risks can compromise patient privacy, disrupt medical device functionality, and even endanger lives [35].

2.2.3 Electronic health records (EHRS)

Electronic Health Records (EHRs) are advanced digital systems that centralize and organize a wide array of patient information, including medical history, diagnoses, medications, immunization records, allergies, radiology images, and lab results. They provide real-time, patient-centered records that are instantly accessible to authorized personnel, anytime and anywhere. EHRs enhance collaboration by allowing multiple healthcare providers to share and access a patient's information, enabling integrated care and better decision-making. They also improve workflows by reducing paperwork, increasing accuracy in record-keeping, and offering evidence-based tools to support clinical decisions. By centralizing and streamlining data, EHRs foster a more patient-centered approach, ensuring that care is tailored to individual needs. These features collectively make EHRs a cornerstone in modern healthcare systems, significantly contributing to better patient outcomes and operational efficiency [34, 15].

3 AI-Driven Techniques in Medical Image Security

3.1 Machine Learning-Based Encryption:

3.1.1 Securing Medical Image Analysis with Encryption Algorithms in Deep Learning

Recent advancements in AI and encryption techniques are transforming healthcare by enabling secure and accurate medical data processing. Naik et al. [28] used DenseNet-121 and AES-128 encryption for identifying lung diseases from chest X-rays. Kumar et al. [21] implemented a cloud-based system for tumor detection in MRI images using CNN with 97.87% accuracy and AES-256 encryption. Mohanty et al. [26] achieved 98.51% accuracy in brain tumor detection with CNN-LSTM secured by a modified SHA-256 algorithm. Other method employed an LSTM model with homomorphic encryption for predicting in-hospital mortality using the MIMIC-III dataset. while [12] developed PINPOINT, a temporal

CNN with homomorphic encryption for time-series predictions, including COVID-19 case forecasting. In [27] reviewed homomorphic encryption applications in cancer detection, cardiovascular analysis, and secure healthcare queries. Boulila et al. [8] classified COVID-19 X-rays with MobileNetV2 and partially homomorphic encryption, achieving 93.3% accuracy. These innovations underscore the potential of combining AI with encryption for secure and efficient healthcare solutions.

3.1.2 Integrating Image Encryption and Compression in Deep Learning for Medical Image Processing

The security and efficient transmission of medical images is essential due to their large size and sensitive nature. Several recent techniques address both encryption and compression to enhance protection. Selvi et al. [31] developed the ASFSCSLEC-DNL method for secure encryption and compression of chest radiograph images, producing promising results. Ahmad et al. [1] proposed a block-based perceptual encryption algorithm combined with JPEG compression for grayscale and color medical images, tested in TB screening on chest radiographs. Kumar et al. [20] introduced MediSecFed, a secure federated learning framework for chest X-ray datasets, outperforming FedAvg by 15% in hostile environments. Hajjaji et al. [13] proposed a novel crypto-compression algorithm using artificial neural networks and chaotic systems, which successfully preserved the security and quality of the medical image during compression.

3.1.3 Key Generation in Encryption Algorithms for Medical Image Analysis

Key generation plays a crucial role in encryption algorithms for medical image analysis, ensuring the confidentiality, integrity, and authenticity of sensitive data. Ding et al. [18] proposed a deep learning-based key generation network (DeepKeyGen), which showed superior security to encrypt medical images, evaluated on data sets such as chest X-rays and the BraTS18 data set. Krishna et al. [19] introduced a dynamic medical image encryption technique using a neural network for key generation, encrypting the key itself for enhanced security. While their method demonstrated strong encryption, the encryption time needs optimization, as tested on X-ray images.

3.2 Watermarking and Data Integrity Verification:

Current research on deep learning-based watermarking focuses mainly on image watermarking, with limited work on text and 3D images, offering improved efficiency and robustness by learning complex patterns resilient to attacks, easily re-trained for different applications, and making signature retrieval difficult due to high non-linearity [6, 7].

Many methods in the literature presented CNN-based techniques for digital image watermarking that enhance both robustness and imperceptibility. These methods [32, 2], [39, 17] [22] utilize various CNN architectures, such as encoder-decoder networks and full convolutional neural networks (FCNNs), to efficiently embed and extract watermarks. They also introduce innovative strategies like adversarial training and attack simulation layers to improve resistance against distortions and attacks, ultimately achieving better trade-offs between robustness and imperceptibility. These CNN-based approaches outperform traditional methods, offering greater adaptability to different image resolutions and improving the overall security of the watermarking process.

The second class of deep learning-based image watermarking utilizes generative adversarial networks (GANs), including variants like Wasserstein GANs (WGANs) and CycleGANs, known for their effectiveness in providing invisibility and robustness. HiDDeN [40] was the first scheme to use an adversarial discriminator to improve watermarking, featuring an encoder, decoder, and adversary network. ROMark [36] improved HiDDeN by minimizing the loss of decoding in various attacks, while another variant incorporated rotation and noise layers to defend against geometric rotations. Zhang et al. [38] introduced a GAN-based technique using inverse gradient attention (IGA) to improve capacity and robustness. Liu et al. [24] proposed a two-stage separable deep learning framework (TSDL), which trains with true non-differentiable noise attacks like JPEG compression, achieving improved robustness compared to previous methods.

3.3 Privacy-preserving solutions in deep learning-based techniques

Recent advances in secure medical data processing highlight the integration of security methods and deep learning to improve accuracy and privacy. Zhang et al. [37] optimized CryptoNets with polynomial ReLU approximations for better classification accuracy in networks with nonlinear layers, while Liu et al. [23] enhanced inference accuracy using MiniONN with secret sharing. Alzubi et al. [4] proposed a blockchain-based BAISMDT model for secure medical data transmission and disease detection. Hesamifard et al. [16] and Carpov et al. [9] emphasized reducing computational costs and improving security in encrypted systems through GPU batch bootstrapping and homomorphic encryption. Federated learning (FL) shows promise in real-world medical data exchange but faces challenges with noisy data, underscoring the need for further research into secure, efficient multiparty computation and privacy-preserving deep learning.

4 Evaluation and Benchmarking of AI Techniques

Comparative Analysis:
AI techniques in medical image security demonstrate varied performance across encryption strength, detection accuracy, and computational efficiency:

- Encryption Strength: Techniques like homomorphic encryption (e.g., MiniONN, CryptoNets) and GAN-based frameworks (e.g., TSDL and IGA) excel in securing medical data during processing and transmission. Approaches integrating modified encryption algorithms, such as AES-128/256 or SHA-256, provide robust data protection, while blockchain-based models like BAISMDT enhance data privacy and integrity during exchange.
- Detection Accuracy: CNN-based methods (e.g., DenseNet-121, CNN-LSTM) achieve high diagnostic accuracy, with some models reporting over 98% in medical image classification and tumor detection. GAN-based watermarking techniques also improve robustness and accuracy in image integrity checks.
- Computational Efficiency: While encryption techniques like homomorphic encryption and neural network-based key generation offer strong security, they often face higher computational costs. Innovations like GPU acceleration, batch bootstrapping, and compression strategies reduce computational overhead, enabling practical deployment in real-world scenarios.

Overall, integrating AI into medical image security balances high accuracy and robust encryption, though computational efficiency remains an area for further optimization.

Dataset and Model Limitations: Medical image datasets face challenges of limited diversity, impacting the ability of AI models to generalize across various demographics, imaging technologies, and clinical settings. This lack of diversity hinders model robustness, particularly in ensuring the security of sensitive patient information during processing and transmission. While advancements like adversarial training, federated learning, and encryption-integrated models (e.g., homomorphic encryption, blockchain) improve data security and robustness, the reliance on biased or narrow datasets continues to limit the scalability and reliability of these AI solutions in real-world medical applications.

5 Challenges and Open Issues

Deep learning for medical image security using cryptography or watermarking techniques faces various challenges such as

- **Limited Generalization** Deep learning models often struggle to adapt to new or diverse medical image data, leading to decreased performance and security. Future research should focus on creating models that generalize well across various imaging modalities, diseases, and patient groups.
- **Vulnerability to Adversarial Attacks** Adversarial attacks can manipulate input data, compromising the integrity and security of encrypted medical images. Future work should prioritize developing robust training techniques and protective mechanisms to mitigate such vulnerabilities.

- **High Computational Costs** One of the primary challenges in applying deep learning to medical image security is the high computational cost. Training deep learning models requires expensive hardware and extensive time. Future research could focus on optimizing algorithms and utilizing hardware accelerators like GPUs or TPUs to reduce these costs, enabling real-time, scalable solutions in healthcare applications.
- **Data Availability and Quality** The scarcity of large, high-quality datasets due to privacy concerns poses a significant challenge. Future developments should focus on privacy-preserving techniques that enable model training on decentralized or encrypted datasets while maintaining data security.

6 Conclusion

This paper explored various AI-driven techniques that play a crucial role in enhancing medical image security. Advanced methods such as convolutional neural networks (CNNs), generative adversarial networks (GANs), federated learning (FL), and homomorphic encryption (HE) have demonstrated remarkable effectiveness in strengthening encryption, improving threat detection, and optimizing computational efficiency. These approaches not only protect sensitive medical data but also ensure its integrity and accessibility within modern healthcare systems.

AI has become indispensable in addressing the escalating cybersecurity challenges in healthcare. By enhancing data privacy and mitigating adversarial threats, AI-driven solutions bridge the gap between security demands and the rapid digital transformation of healthcare infrastructures. Their adaptability and scalability make them essential for managing the growing volumes of medical data securely.

Looking ahead, the integration of AI presents vast opportunities for advancing medical data security. Future research should focus on developing more efficient, generalizable, and secure models that overcome dataset limitations and computational constraints. As AI continues to evolve, it will play a pivotal role in strengthening healthcare cybersecurity, safeguarding patient privacy, and enabling the seamless exchange of medical information in an increasingly connected world.

References

[1] Ijaz Ahmad and Seokjoo Shin. A perceptual encryption-based image communication system for deep learning-based tuberculosis diagnosis using healthcare cloud services. *Electronics*, 11(16):2514, 2022. doi: https://doi.org/10.3390/electronics11162514.

[2] Mahdi Ahmadi, Alireza Norouzi, Nader Karimi, Shadrokh Samavi, and Ali Emami. Redmark: Framework for residual diffusion watermarking based on deep networks. *Expert Systems with Applications*, 146:113157, 2020. doi: https://doi.org/10.1016/j.eswa.2019.113157.

[3] Steve Alder. Hacking incidents reported by retinal consultants medical group, three rivers regional commission, ace surgical supply. *The HIPAA Journal*, Nov 25, 2021.

[4] Omar A Alzubi, Jafar A Alzubi, K Shankar, and Deepak Gupta. Blockchain and artificial intelligence enabled privacy-preserving medical data transmission in internet of things. *Transactions on Emerging Telecommunications Technologies*, 32(12):e4360, 2021. doi: https://doi.org/10.1002/ett.4360.

[5] Siva Subrahmanyam Balantrapu. A comprehensive review of ai applications in cybersecurity. *International Machine learning journal and Computer Engineering*, 7(7), 2024.

[6] Aicha Benyoucef and M'Hamed Hamadouche. Roni-based medical image watermarking using dwt and lsb algorithms. In *International Conference on Artificial Intelligence and its Applications*, pages 468–478. Springer, 2021. doi: https://doi.org/10.1007/978-3-030-96311-$8_4$3.

[7] Aicha Benyoucef and M'Hamed Hamaouche. Region-based medical image watermarking approach for secure epr transmission applied to e-health. *Arabian Journal for Science and Engineering*, 49(3):4025–4037, 2024. doi: https://doi.org/10.1007/s13369-023-08263-0.

[8] Wadii Boulila, Adel Ammar, Bilel Benjdira, and Anis Koubaa. Securing the classification of covid-19 in chest x-ray images: A privacy-preserving deep learning approach. In *2022 2nd International*

Conference of Smart Systems and Emerging Technologies (SMARTTECH), pages 220–225. IEEE, 2022. doi: 10.1109/SMARTTECH54121.2022.00055.

[9] Sergiu Carpov, Thanh Hai Nguyen, Renaud Sirdey, Gianpiero Constantino, and Fabio Martinelli. Practical privacy-preserving medical diagnosis using homomorphic encryption. In *2016 ieee 9th international conference on cloud computing (cloud)*, pages 593–599. IEEE, 2016. doi: 10.1109/CLOUD.2016.0084.

[10] Haiwen Chen, Jiaping Yu, Fang Liu, Zhiping Cai, and Jing Xia. Archipelago: A medical distributed storage system for interconnected health. *IEEE Internet Computing*, 24(2):28–38, 2019. doi: 10.1109/MIC.2019.2963182.

[11] Shekha Chenthara, Khandakar Ahmed, Hua Wang, and Frank Whittaker. Security and privacy-preserving challenges of e-health solutions in cloud computing. *IEEE access*, 7:74361–74382, 2019. doi: 10.1109/ACCESS.2019.2919982.

[12] Alessandro Falcetta and Manuel Roveri. Privacy-preserving time series prediction with temporal convolutional neural networks. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022. doi: 10.1109/IJCNN55064.2022.9892823.

[13] Mohamed Ali Hajjaji, Manel Dridi, and Abdellatif Mtibaa. A medical image crypto-compression algorithm based on neural network and pwlcm. *Multimedia Tools and Applications*, 78:14379–14396, 2019. doi: https://doi.org/10.1007/s11042-018-6795-6.

[14] Jigna J Hathaliya and Sudeep Tanwar. An exhaustive survey on security and privacy issues in healthcare 4.0. *Computer Communications*, 153:311–335, 2020. doi: https://doi.org/10.1016/j.comcom.2020.02.018.

[15] Mireya Lucia Hernandez-Jaimes, Alfonso Martinez-Cruz, Kelsey Alejandra Ramírez-Gutiérrez, and Claudia Feregrino-Uribe. Artificial intelligence for iomt security: A review of intrusion detection systems, attacks, datasets and cloud-fog-edge architectures. *Internet of Things*, page 100887, 2023. doi: https://doi.org/10.1016/j.iot.2023.100887.

[16] Ehsan Hesamifard, Hassan Takabi, and Mehdi Ghasemi. Cryptodl: Deep neural networks over encrypted data. *arXiv preprint arXiv:1711.05189*, 2017. doi: https://doi.org/10.48550/arXiv.1711.05189.

[17] Haribabu Kandi, Deepak Mishra, and Subrahmanyam RK Sai Gorthi. Exploring the learning capabilities of convolutional neural networks for robust image watermarking. *Computers & Security*, 65:247–268, 2017. doi: https://doi.org/10.1016/j.cose.2016.11.016.

[18] P Keerthana, N Thirumalaivasan, K Vigneshari, D Kiruthika, R Monica, and V Manthra. A deep learning-based stream cipher generator for medical image encryption and decryption. In *2024 2nd International Conference on Networking, Embedded and Wireless Systems (ICNEWS)*, pages 1–6. IEEE, 2024. doi: 10.1109/ICNEWS60873.2024.10730861.

[19] A Anantha Krishna, Vanya Arikutharam, K Venkat Ramnan, H Bharathi, and TS Chandar. Dynamic image encryption using neural networks for medical images. In *2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET)*, pages 739–745. IEEE, 2022. doi: 10.1109/GlobConET53749.2022.9872401.

[20] Abhinav Kumar, Vishal Purohit, Vandana Bharti, Rishav Singh, and Sanjay Kumar Singh. Mediseefed: Private and secure medical image classification in the presence of malicious clients. *IEEE Transactions on Industrial Informatics*, 18(8):5648–5657, 2021. doi: 10.1109/TII.2021.3138919.

[21] JNVR Swarup Kumar, G Sri Jyothi, DNVSLS Indira, and Tenali Nagamani. Secured cloud application for detection of brain tumor using deep learning algorithms. In *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 656–663. IEEE, 2022. doi: 10.1109/ICIRCA54612.2022.9985666.

[22] Jae-Eun Lee, Young-Ho Seo, and Dong-Wook Kim. Convolutional neural network-based digital image watermarking adaptive to the resolution of image and watermark. *Applied Sciences*, 10(19):6854, 2020. doi: https://doi.org/10.3390/app10196854.

[23] Jian Liu, Mika Juuti, Yao Lu, and Nadarajah Asokan. Oblivious neural network predictions via minionn transformations. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pages 619–631, 2017. doi: https://doi.org/10.1145/3133956.3134056.

[24] Yang Liu, Mengxi Guo, Jian Zhang, Yuesheng Zhu, and Xiaodong Xie. A novel two-stage separable deep learning framework for practical blind watermarking. In *Proceedings of the 27th ACM International conference on multimedia*, pages 1509–1517, 2019. doi: https://doi.org/10.1145/3343031.3351025.

[25] Pratap Chandra Mandal, Imon Mukherjee, Goutam Paul, and BN Chatterji. Digital image steganography: A literature survey. *Information sciences*, 609:1451–1488, 2022. doi: https://doi.org/10.1016/j.ins.2022.07.120.

[26] Mohan Debarchan Mohanty, Abhishek Das, Mihir Narayan Mohanty, Ayman Altameem, Soumya Ranjan Nayak, Abdul Khader Jilani Saudagar, and Ramesh Chandra Poonia. Design of smart and secured healthcare service using deep learning with modified sha-256 algorithm. In *Healthcare*, volume 10, page 1275. MDPI, 2022. doi: https://doi.org/10.3390/healthcare10071275.

[27] Kundan Munjal and Rekha Bhatia. A systematic review of homomorphic encryption and its contributions in healthcare industry. *Complex & Intelligent Systems*, 9(4):3759–3786, 2023. doi: https://doi.org/10.1007/s40747-022-00756-z.

[28] Rasika Naik, Tejas Wani, Sakshi Bajaj, Shiva Ahir, and Atharva Joshi. Detection of lung diseases using deep learning. In *Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST)*, 2020.

[29] Mary Nankya, Allan Mugisa, Yusuf Usman, Aadesh Upadhyay, and Robin Chataut. Security and privacy in e-health systems: A review of ai and machine learning techniques. *IEEE Access*, 2024. doi: 10.1109/ACCESS.2024.3469215.

[30] Adil Hussain Seh, Mohammad Zarour, Mamdouh Alenezi, Amal Krishna Sarkar, Alka Agrawal, Rajeev Kumar, and Raees Ahmad Khan. Healthcare data breaches: insights and implications. In *Healthcare*, volume 8, page 133. MDPI, 2020. doi: https://doi.org/10.3390/healthcare8020133.

[31] C Thirumarai Selvi, J Amudha, and R Sudhakar. Medical image encryption and compression by adaptive sigma filterized synorr certificateless signcryptive levenshtein entropy-coding-based deep neural learning. *Multimedia Systems*, 27(6):1059–1074, 2021. doi: https://doi.org/10.1007/s00530-021-00764-y.

[32] Himanshu Kumar Singh and Amit Kumar Singh. Digital image watermarking using deep learning. *Multimedia Tools and Applications*, 83(1):2979–2994, 2024. doi: https://doi.org/10.1007/s11042-023-15750-x.

[33] Kai Taylor, Alexandra Smith, Adam Zimmel, Korina Alcantara, and Yong Wang. Medical device security regulations and assessment case studies. In *2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, pages 742–747. IEEE, 2022. doi: 10.1109/MASS56207.2022.00116.

[34] S Vishnu, SR Jino Ramson, and R Jegan. Internet of medical things (iomt)-an overview. In *2020 5th international conference on devices, circuits and systems (ICDCS)*, pages 101–104. IEEE, 2020. doi: 10.1109/ICDCS48716.2020.243558.

[35] Mohammad Wazid, Ashok Kumar Das, Joel JPC Rodrigues, Sachin Shetty, and Youngho Park. Iomt malware detection approaches: analysis and research challenges. *IEEE access*, 7:182459–182476, 2019. doi: 10.1109/ACCESS.2019.2960412.

[36] Bingyang Wen and Sergul Aydore. Romark: A robust watermarking system using adversarial training. *arXiv preprint arXiv:1910.01221*, 2019. doi: https://doi.org/10.48550/arXiv.1910.01221.

[37] Dayin Zhang, Xiaojun Chen, Dakui Wang, and Jinqiao Shi. A survey on collaborative deep learning and privacy-preserving. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pages 652–658. IEEE, 2018. doi: 10.1109/DSC.2018.00104.

[38] Honglei Zhang, Hu Wang, Yuanzhouhan Cao, Chunhua Shen, and Yidong Li. Robust data hiding using inverse gradient attention. *arXiv preprint arXiv:2011.10850*, 2020. doi: https://doi.org/10.48550/arXiv.2011.10850.

[39] Xin Zhong, Pei-Chi Huang, Spyridon Mastorakis, and Frank Y Shih. An automated and robust image watermarking scheme based on deep neural networks. *IEEE Transactions on Multimedia*, 23:1951–1961, 2020. doi: 10.1109/TMM.2020.3006415.

[40] J Zhu. Hidden: hiding data with deep networks. *arXiv preprint arXiv:1807.09937*, 2018.

A novel cloud-deployed data pipeline for cervical spine fracture detection in 3D CT images

Fatah Bouchebbah[1], Rayane Aggoune[2], and Chahinez Amrane[2]

[1]*LIMED Laboratory, Faculty of Exact Sciences, University of Bejaia, 06000 Bejaia, Algeria,* fatah.bouchebbah@univ-bejaia.dz

[2]*Department of Computer Science, Faculty of Exact Sciences, University of Bejaia, 06000 Bejaia, Algeria*

Abstract

A cervical spine fracture is a serious medical emergency that can lead to permanent paralysis or even death. In addition, rapid and accurate detection of such fractures is essential for optimal patient care. However, manually interpreting computed tomography (CT) images to detect possible fractures in the cervical spine, as traditionally done, is time-consuming and requires the experience of experienced radiologists. Fortunately, the integration of artificial intelligence and cloud computing technologies in healthcare has the potential to revolutionize cervical spine fracture detection by providing fast, accurate, and automated solutions. In this context, we present a couple of contributions in this paper. In the first contribution, we develop a new multifaceted computational pipeline based on the combination of Faster R-CNN and Next-ViT models to detect fractures within the cervical spine. The new computational pipeline has been trained and evaluated on the large RSNA public dataset containing cervical spine CT scans. Hence, the new system has achieved encouraging results. Furthermore, the new proposed data pipeline's ability to detect subtle and complex fractures has motivated us to integrate it in a cloud-based architecture that we present as a second contribution in the setting of this paper. The proposed cloud-based architecture has the potential to be used as a distant clinical decision-support tool to help radiologists identify fractures quickly and reliably, and to be continuously improved through a feedback mechanism.

Keywords: Fracture detection, Cervical spine, Faster R-CNN, Next-ViT model; Cloud-based architecture.

1 Introduction

Cervical spine fractures, often caused by accidents or falls, pose a challenging medical dilemma. These kinds of injuries, which occur in a delicate part of the human skeletal structure, require swift and precise identification to prevent serious neurological damage. Moreover, according to Savage et al. [5], more than 1.5 million people in the United States alone suffer spine fractures every year, a significant proportion of which affect the delicate architecture of the cervical spine. For the elderly and those with pre-existing conditions like osteoporosis, such fractures can be fatal. The situation is further complicated by the fact that cervical spine fractures often require immediate attention, yet rapid and accurate diagnosis remains elusive.

Fortunately, in the current age of rapid technological advancement, Artificial Intelligence (AI) and cloud computing are making profound inroads into various domains. In fact, as reported by Voter et al. [9], the combination of AI's cutting-edge technologies and the prowess characteristics of cloud-based systems offer innovative solutions featured with computational capabilities and abilities to decipher intricate medical data patterns. Especially when it comes to challenging medical situations like cervical spine fractures which are marked by complex diagnostics and the potential for severe neurological consequences if mishandled.

Our main objective throughout this paper is to improve patient outcomes and to assist healthcare professionals by presenting an accurate, rapid, automatic, secured, and continuously improving advanced system for cervical spine fracture detection. The system relies on a combination of deep learning algorithms like Faster R-CNN and Next-ViT, that have gained significant attention in the computer vision community due to their recent remarkable state-of-the-art performances, as well as cloud computing and human expertise.

The rest of this paper is organized as follows: Section 2 exhibits a state of the art of the main works established in the literature on the cervical spine fracture detection problem. Section 3 presents in details our first contribution in this paper, which is a new multifaceted computational pipeline based on a

combination of Faster R-CNN and Next-ViT. Section 4 describes and discusses our second contribution, that is a proposed cloud-based architecture deployed in Google Cloud Platform that offers an end-to-end cervical spine fracture detection service to medical professionals as well as to patients. Section 5 is dedicated to testing and evaluating the whole presented system by comparing it to an existing system in the literature by considering the RSNA 2022 Cervical Spine Fracture Detection dataset. The paper ends with a conclusion, given in Section 6, summarizing the main of the contributions while highlighting some limitations and interesting perspectives worth to consider to improve the work.

2 Related work

As a research problem, detection of injuries, and especially fractures, in cervical spine has been a topic of interest lately. Therefore, a variety of methods that are based on different techniques of machine learning have been presented in the literature as attempts to find solutions to the problem. Among the exhibited methods, the approached based on deep learning have demonstrated encouraging features. However, they still require several improvements to be effective enough when integrated in clinical routines.

For instance, Small et al. [7] have investigated the application of a Convolutional Neural Network (CNN) architecture that was developed by Aidoc, known as FDA-approved CNN, for the detection of cervical spine fractures. The findings of the study have emphasized the potential of the tested model to improve cervical fracture detection. However, they have also acknowledged certain limitations of CNNs when they are applied to detect fractures. Notably, CNNs may struggle to detect areas of gross bony translation and fractures characterized by distraction rather than linear bony features. Moreover, Merali et al. [3] have conducted a study with the objective of developing a deep-learning model capable of detecting cervical spinal cord compression in patients diagnosed with Degenerative Cervical Myelopathy (DCM) in T2-weighted MRI scans. For this aim, the authors have employed ResNet-50 architecture and have tested multiple network configurations to determine a suitable setup for the used dataset. The used architecture, with a proper settings, has achieved an encouraging accuracy, however the results in terms of specificity have been relatively low. Furthermore, Shaolong et al. [6] have presented a comprehensive investigation into the utilization of deep learning techniques applied to MRI scans for the detection and classification of lesions associated with cervical spinal cord diseases. For this reason, the researchers have employed Faster R-CNN (Region Convolutional Neural Network) approach, which combines a backbone convolutional feature extractor utilizing both ResNet-50 and VGG-16 networks. This integration of latter networks yielded promising results in terms of prediction accuracy and speed for lesion detection and recognition within cervical spinal cord MRIs. In addition, Tuan et al. [8] have conducted an extensive investigation to develop an efficient and accurate method for the early detection and localization of spine fractures. Through their experimentation, they have explored multiple machine learning models and hence have identified a two-stage approach utilizing Deep CNN (DCNN) with RNN and attention layers. The presented approach have had commendable performance in terms of running time, resource utilization, and accuracy. In addition, Salehinejad et al. [4] have introduced a DCNN with a Bidirectional Long Short-Term Memory (BLSTM) layer as the baseline architecture, that has been specifically tailored for an automated detection of cervical spine fractures in CT axial images. The performed study has shaded light on the potential of deep learning techniques in fracture detection and has provided a foundation for future investigations aimed at refining and advancing automated fracture detection algorithms in clinical settings. Unfortunately, the approaches presented by Small et al. [7], Merali et al. [3], Shaolong et al. [6], Tuan et al. [8], and Salehinejad et al. [4] remain beneficial to the health professionals who own the programs only. Therefore, they are restricted to a local use solely. Recently, Showmick Guha et al. [1] have studied the performance of a variety of CNN models adapted to cervical spine fracture detection using transfer-learning. The adapted methods include MobileNetV2, InceptionV3, and Resnet50V2. Performed tests have revealed a superiority of MobileNetV2, which was trained with data augmentation technique, over the other approaches. Consequently, the model has been deployed for clinical use in the form of an Android application for smartphones. However, this kind of deployment is not quietly proper to a clinical use, unless the attention is restricted to personal or emergency use. On the other hand, many matters like constrained resources and model evolution need to be resolved for a better usage.

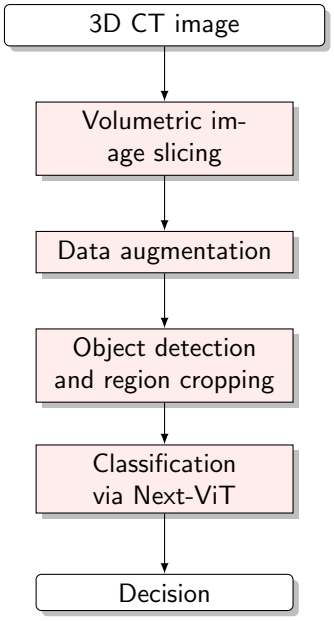


Figure 1: A representation of the proposed data pipeline.

3 Proposed multifaceted computational pipeline for the detection cervical spine fractures

In light of the importance and challenge of cervical spine fractures detection, we synergies cutting-edge algorithms to present a new data pipeline specifically designed for CT images analysis. The proposed data pipeline leverages the proven capabilities of Faster R-CNN for object localization and Next-ViT for image classification, and adds insight from attention maps to focus on the region(s) of interest within an analyzed CT image (i.e. the eventual fracture(s)). Explicitly, the presented data pipeline is essentially composed of four stages, namely: *volumetric image slicing*, *data augmentation to train Faster R-CNN*, *object localization using Faster R-CNN and image cropping*, and finally *classification via Next-ViT*. A schematic representation of the proposed framework is presented in Fig. 1 and necessary details and descriptions about the proposed data pipeline are given in the subsections below.

3.1 Volumetric image slicing

Cervical CT scans are 3D images rich in anatomical information. However, these latter are quite complex to process by computerized approaches due to their excessive amount of data. The question raised here is therefore how to exploit the richness of these data without getting trapped in computational bottlenecks?

Herein lies the critical importance of the image-slicing process. In fact, it transforms an intricate 3D spatial problem into a more manageable 2D problem space. Thus, slicing serves, on one hand, as a strategic maneuver to reduce computational cost; on the other hand, it prepares the ground for expeditious and focused downstream data processing. Specifically, the produced 2D slices can be orientated to emphasize anatomical planes that are most relevant for the diagnosis of cervical spine fractures. This ensures to retrain the most pertinent and diagnostically relevant information in the slices.

Furthermore, in this initial phase of the pipeline, slices are extracted from the original DICOM files of CT scans using a specific function in 512×512 pixels format. Which are later resized into 224×224 pixels format to match the input size expected by the Next-ViT model. In addition, windowing techniques are applied to the extracted 2D slices to enhance their contrast. The window width and level are set to 1800 and 400, respectively.

3.2 Data augmentation to train Faster R-CNN

Data augmentation is a widely used technique to increase the size and diversity of the training datasets. This is especially important as Faster R-CNN object detection model requires a large amount of labelled data to be trained effectively. In the context of this study, the data augmentation is mainly performed

by using random horizontal flipping, which can help the model learn to detect objects from different perspectives.

3.3 Object detection and region cropping

In this stage, Faster R-CNN is used to detect and isolate regions of interest. Specifically, we take advantage of Faster R-CNN's ability to operate as a computerized lens, to scrupulously navigate through the 2D slice images to discern and delineate regions that house potential fracture sites within the cervical spine and to underscore their associated vertebra with bounding boxes. This act of object localization constructs a vital foundational tier, guiding the ensuing procedures in the pipeline, which are designated to further refine, dissect, and classify these pronounced areas suspected of fractures.

The localized regions of interest are subsequently cropped from the rest of their associated slices to form small imagettes. Specifically, the aim of this phase is dual: firstly, to drastically curtail computational excess, and secondly, to concentrate the ensuing analysis on clinically pertinent regions. Explicitly, the sectors of the cervical spine believed to harbour fractures, as pinpointed by Faster R-CNN.

3.4 Classification via Next-ViT

In this final stage, Next-ViT model is used to binary classify the imagettes previously produced to distinguish between those really containing fractures and those that are not. This model was selected for its unique set of attributes that align impeccably with our research goals. One of the standout qualities of Next-ViT is its data efficiency. The model demonstrates impressive performance even when subjected to small, annotated datasets. In addition, Next-ViT diverges from CNNs by incorporating self-attention mechanisms. These latter mechanisms excel at identifying complex spatial and contextual relationships within images, a feature invaluable for interpreting the complex imagery commonly found in cervical spine studies.

Moreover, given the underwhelming results of our initial attempt to train a vision transformer from scratch, we have chosen to adopt a pre-trained Next-ViT architecture, which led to a marked improvement in our system's efficacy. However, to better fit Next-ViT to our problem, we have performed a refinement training of the model, specifically using data augmentation techniques by applying simple transformations to the training dataset (*i.e.* rotation, scaling, and flipping). We hypothesize that these simple transformations assist the model in understanding underlying data patterns, thereby improving its learning capability.

4 Proposed cloud-based architecture for cervical spine fracture detection

As a second contribution in this paper, we describe a robust and scalable cloud-based system that is dedicated to the detection of cervical spine fractures. The cloud infrastructure serves as the backbone supporting the entire multifaceted computational pipeline presented in Section 3 and offers unique advantages both in terms of computational resources and data management.

4.1 Motivations and goals

The presented architecture is motivated by several compelling incentives for coupling cloud computing and deep learning models in the arena of cervical spine fracture detection. The impetus for adopting a cloud-based approach originates from a critical need to address challenges in scalability, data integrity, and real-time analytics. Below are the main key motivations:

1. **Superior diagnostic accuracy:** Traditional diagnostic approaches, although useful, sometimes fail to identify complex or subtle fractures. The marriage of cloud-based computational power and well established deep learning models has the potential to usher in a new era of nuanced and precise diagnoses.

2. **Operational efficiency:** Utilizing the distributed computing power of the cloud alongside deep learning models that can efficiently parse large sets of image data enhances the operational efficiency of the diagnostic process. This could significantly reduce the time radiologists need to reach a diagnosis.

3. **Scalability and adaptability:** The inherent scalability of cloud infrastructure is well-suited for handling the voluminous medical imaging data generated daily. This removes the need for healthcare organizations to make significant investments in local computing resources.

4. **Broadened access to advanced tools:** Cloud-based systems democratize access to cutting-edge diagnostic technologies. This model allows healthcare providers, regardless of their size or location, to benefit from state-of-the-art tools without prohibitive upfront costs.

5. **Augmentation of clinical decision-making:** The synergy between cloud technology and deep learning models can act as a potent decision-support mechanism. It can provide preliminary evaluations that assist healthcare professionals in making timely and well-informed decisions.

6. **Future-ready integration:** The modular architecture of cloud-based systems makes them ripe for seamless integration with existing electronic health records. This offers the possibility for more integrated, collaborative approaches to healthcare delivery in the future.

Thus, the integration of cloud computing and deep learning models in the detection of cervical spine fractures has the potential to surmount existing limitations, refine diagnostic protocols, democratize access to state-of-the-art technologies, and fundamentally transform clinical practices in this vital area of healthcare.

4.2 Description of the proposed cloud-based architecture

The proposed architecture is designed to be deployed in Google Cloud Platform (GCP), integrating its services to offer an efficient end-to-end cervical spine fracture detection workflow. A general view of the proposed cloud-based architecture for cervical spine fracture detection is illustrated in Fig. 2.

Initially, the overarching vision of crafting an integrated end-to-end diagnostic workflow for enhanced cervical spine fracture detection stemmed from comprehensive brainstorming sessions. Significantly, it was our deep dive into the vast capabilities of the Google Cloud Platform (GCP) that galvanized our alignment with this mission. Building upon this foundation, our hands played a pivotal role in the ensuing architectural design and execution phase.

Furthermore, recognizing the paramount importance of data integrity, we have channelled significant efforts into devising an efficient automatic ingestion mechanism for CT scans. Simultaneously, with an acute awareness of the sensitive nature of medical data, we have championed the incorporation of a robust encryption protocol, ensuring that data remain secured.

Transitioning from data acquisition, our focus then have gravitated towards the multi-layered data pipeline. Specifically, we have integrated in the proposed architecture the data pipeline elaborated in Section 3, that meticulously optimize the mechanisms of pre-processing, feature extraction, and fracture detection.

On the other hand, we believe in the interdependent nexus between machine learning methods and human expertise and its capacity to offer better solutions, especially when they are combined appropriately. This conviction has led to the establishment of a systematic feedback loop, where the invaluable insights of medical professionals continuously enrich our cloud-based system. Through this mechanism, their diagnostic evaluations directly inform and steer the iterative enhancements of the integrated models in the proposed system.

Moreover, with an ever-evolving medical landscape, we need to ensure that the used models in the architecture underwent consistent training sessions. By leveraging insights from the analytical database, our diagnostic algorithms remain at the cutting edge, always adaptive to the latest nuances in medical diagnostics.

Beyond the technical realm, we endeavour to foster a culture of interdisciplinary collaboration. By orchestrating synergy between cloud experts, data scientists, and medical professionals, we strive to ensure that our collective expertise coalesced seamlessly. This unity of purpose and knowledge-sharing became instrumental in shaping our presented solution.

In summary, witnessing the transformative potential of our architecture in the realm of medical diagnostics has been both a privilege and a testament to the collaborative prowess of our team. Our journey exemplifies the boundless possibilities that emerge when cloud computing and machine learning converge, especially in the ever-critical domain of healthcare. The amalgamation of GCP's advanced services presents a promising horizon for medical diagnostics. While this overview provides a high-level design, the actual implementation should be tailored according to specific requirements, ensuring

Epoch	Precision	Recall	mAP0.5
80	0.9287	0.8857	0.9424
100	0.9687	0.9057	0.9724

Table 1: Performance metrics of the Faster R-CNN on the RSNA dataset.

a balance between functionality, budget, and privacy concerns. Collaboration with cloud and domain experts is essential for the successful realization of such a system.

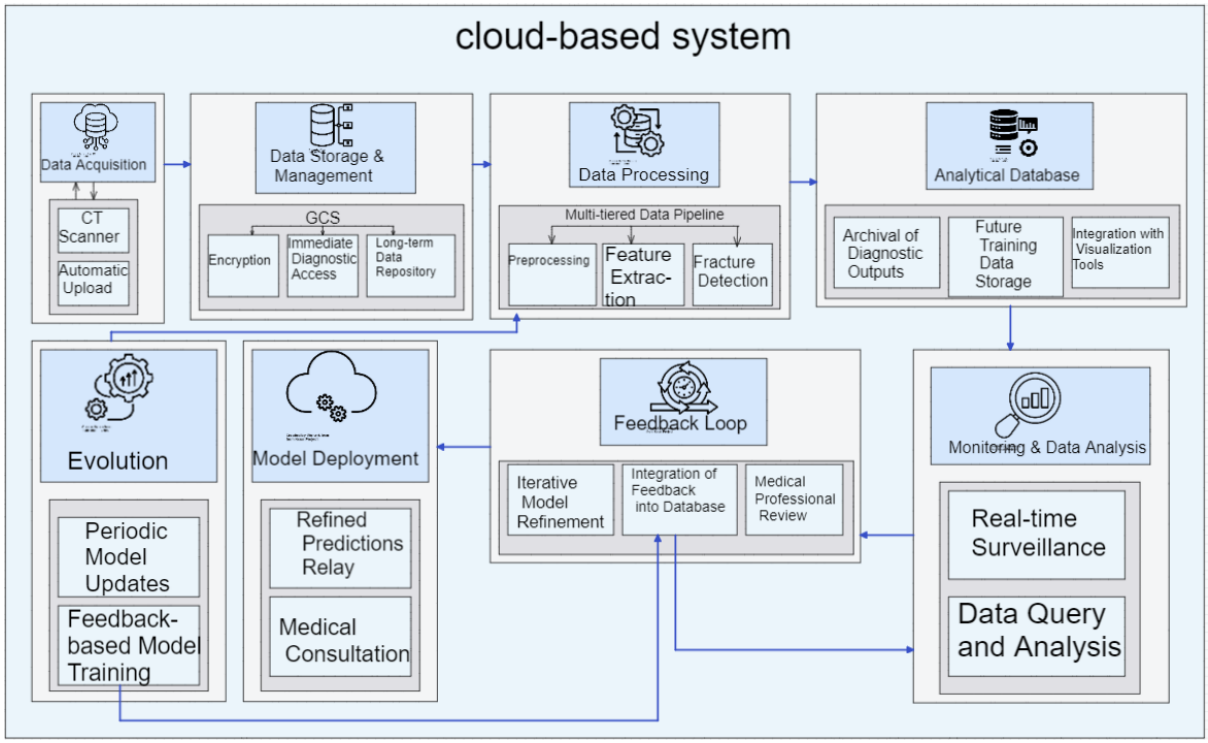


Figure 2: A representation of the proposed cloud-based system.

5 Evaluation and discussion of the cervical spine fracture detection system

The proposed multifaceted data pipeline has been trained and evaluated using the large RSNA public dataset containing cervical spine CT scans [2]. In the setting of this work, the images of the dataset was split into training (80%) and validation (20%) sets. The slices and their corresponding label files (.txt files) are then organized appropriately into separate directories for training and validation.

Subsequently, we have downloaded Faster R-CNN's code from TensorFlow, adjusted it and trained it to meet our purpose. Thus, the performance of Faster R-CNN on the used dataset is assessed using standard evaluation metrics, namely: *precision*, *recall*, and *mean average precision at IoU (mAP50)*.

The obtained results after 80 and 100 epochs are presented in Table 1. The yielded results showcase the model's potential in both recognizing and pinpointing objects within images after 80 and 100 epochs.

A summary of Faster R-CNN train loss metrics from one of the epochs during the model's training phase are present in Table 2. The table summarizes important performance indicators and parameters that provide insights into the model's training dynamics, namely : Loss, Loss Classifier, Loss Box Reg, Loss Objectness, and Loss RPN Box Reg.

For visual illustration of the cropping operation results, we give in Fig. 3 cropped images obtained from different slices.

Concerning the classification stage of the multifaceted data pipeline, the implementation of Next-ViT requires setting appropriate values for model's parameters. This is specifically done to insure a satisfying

Parameter	Best value	Averaged value
Loss	0.1576	0.3236
Loss Classifier	0.0490	0.1163
Loss Box Reg	0.0900	0.1130
Loss Objectness	0.0088	0.0790
Loss RPN Box Reg	0.0040	0.0153

Table 2: Training Loss results.

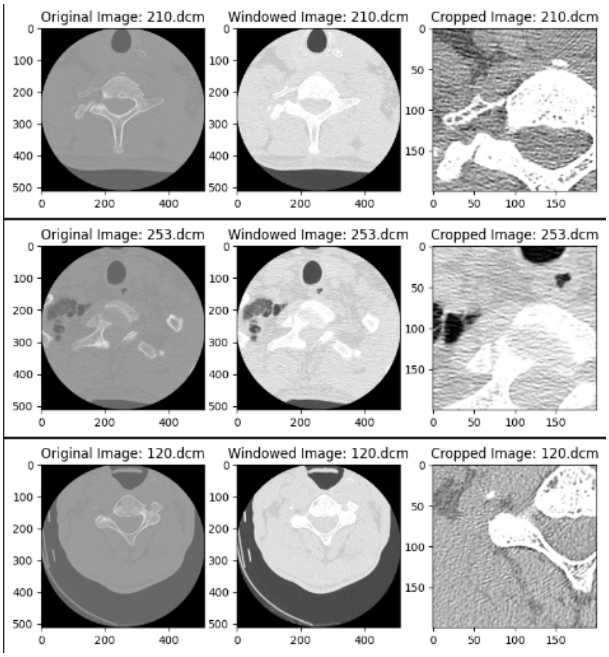


Figure 3: Illustrations of cropped vertebra.

balance between computational efficiency and detail resolution to make the model highly applicable in clinical settings for which timely and accurate diagnosis is paramount. In the context of this work, we have considered the parameter tuning exhibited in Table 3.

Parameter	Value
Patch size	16×16
Latent space dimension	192
Number of encoder blocks	12
Number of MLP heads	3
Total parameters	$\approx$5.5M

Table 3: Next-ViT model parameter tuning.

Also, it is worth to note that we have applied other adaptations to Next-ViT to meet our specific needs. For instance, the output shape is printed and should be [16, 2] of shape. This is to say that for each input image, we get 2 values as output, corresponding to *fracture* and *no fracture* results respectively. In addition, to optimize the neural network, we have employed RAdam optimizer and used a learning rate of 0.001. Specifically, the value of 0.001 is considered a moderate choice, which is neither too high to cause instability nor too low to slow down the learning process. This value is often recommended for Adam and its variants like RAdam due to its effectiveness in a wide range of scenarios.

To validate the robustness and effectiveness of Next-ViT model, we have used two metrics: *accuracy* and *loss*. Hence, the obtained validation results of the model on the RSNA 2022 Cervical Spine Fracture Detection dataset are shown in the graphs presented in Fig. 4. From the latter figure, it is easy to notice that the performance of the Next-ViT model improved through the epochs for both accuracy and loss validation metrics, until achieving a validation accuracy of 95.5% and a validation loss of 2%. This is particularly promising because it suggests that the Next-ViT could be used to develop a fast and accurate

AI-based system for cervical spine fracture detection. Such a system could be used to help radiologists identify fractures more quickly and reliably, and it could also be used to screen patients for suspected fractures in emergency settings.

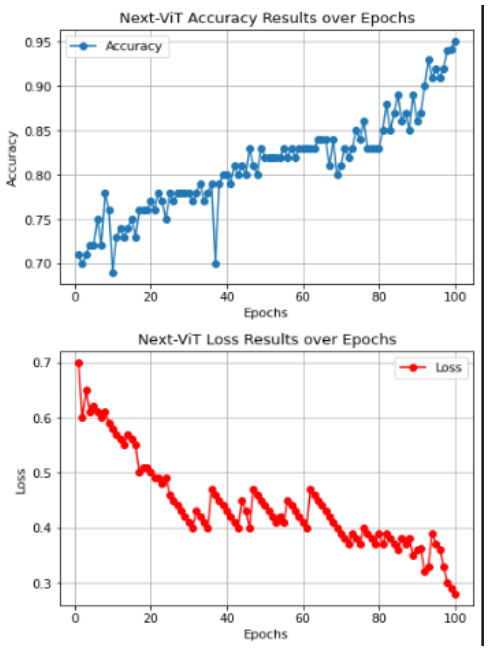


Figure 4: Validation results of Next-ViT.

Moreover, a comparison of the work presented herein with the concurrent work of Showmick Guha et al. [1] that is recently exhibited in the literature is reported in Table 4. From the latter, it is easy to notice that the model of Showmick Guha et al. [1] presents the best accuracy currently. However, the proposed model is more subtitle to offer a superior diagnostic accuracy in the future. In fact, the exhibited architecture foresees continuous model refinement training by taking into consideration the capacity of accepting new unseen data as well as correcting feedback from experts who use the system. So, the two features guarantee a continuously improving diagnostic accuracy. Furthermore, despite the fact that the two works ensure a real time response, nevertheless, the proposed system is clearly more adapted to clinical routines considering the fact that it is deployed on the cloud. Hence, it offers a better scalability and adaptability in terms of resources, brocaded access to advanced tools, and future-ready integration compared to its concurrent work which is designed for miniaturized systems essentially made for a personal use.

Feature	**Proposed work**	**Showmick Guha et al. [1]**
Best accuracy	95,5 %	99.75 %
Deployment	Cloud	Android application
Real-time response	Considered	Considered
Model refinement possibility	Considered	Not considered
Data integrity	Considered	Not considered
Storage and processing capacity	High	Very low

Table 4: Comparison between the proposed work and a concurrent work according to few features.

6 Conclusion

The confluence of AI, medical imaging, and cloud computing represents a promising avenue for revolutionizing the healthcare domain. In this setting, we have made a couple of contributions which are exhibited in this document. Mainly, we have introduced a new comprehensive computational data pipeline tailored for the detection of cervical spine fractures. Specifically, the proposed data pipeline is composed of four

stages, each of which fulfils a unique role to achieve high diagnostic precision and reliability. Furthermore, motivated by several key goals such as improving diagnostic accuracy, increasing scalability, and enhancing data security, we have exhibited, as our second main contribution, a new cloud-based system to extend the capabilities of our computational data pipeline. The new cloud-based architecture represents a paradigm shift in how cervical spine fractures can be detected and managed. The proposed cloud-based system not only streamlines the workflow but also allows for continuous improvement through real-time feedback mechanisms.

Furthermore, the experimental study comprising the implementation, training, and validation of the presented comprehensive computational data pipeline over the RSNA 2022 Cervical Spine Fracture Detection dataset has shown an encouraging performance with regard to a concurrent work in the literature.

While the findings of this paper are compelling, they raise several salient questions that could form the basis of future scholarly inquiry. These include prototyping the proposed cloud-based diagnostic system and its convenience to resource-constrained devices, as well as further refinements and improvements of the proposed multifaceted data pipeline by the integration visualization mechanisms or with the adoption of emerging artificial intelligence paradigms such as deep reinforcement learning and federated learning.

Acknowledgment

The authors are thankful to the anonymous reviewer for his valuable comments that helped to improve the paper.

References

[1] P. Showmick Guha, S. Arpa, and A. Md. A real-time deep learning approach for classifying cervical spine fractures. *Healthcare Analytics*, 4:100265, 2023.

[2] H.M Lin, E. Colak, T. Richards, F.C. Kitamura, L.M. Prevedello, J. Talbott, R.L. Balland E. Gumeler, K.W. Yeom, M. Hamghalam, et al. The RSNA cervical spine fracture CT dataset. *Radiology: Artificial Intelligence*, 5:e230034, 2023.

[3] Z. Merali, J. Wang, J.H. Badhiwala, C.D. Witiw, J.R. Wilson, and M.G. Fehlings. A deep learning model for detection of cervical spinal cord compression in mri scans. *Scientific Reports*, 11(1):14620, 2021.

[4] H. Salehinejad, E. Ho, H.M. Lin, P. Crivellaro, and O. Samorodova. Deep sequential learning for cervical spine fracture detection in computed tomography imaging. *IEEE Transactions on Medical Imaging*, 40(6):1642–1652, 2021.

[5] J.W. Savage, G.D. Schroeder, and P.A. Anderson. Vertebroplasty and kyphoplasty for the treatment of osteoporotic vertebral compression fractures. *J Am Acad Orthop Surg*, 22(10):653–664, Oct 2014.

[6] M. Shaolong, H. Yang, C. Xiangjiu, and G. Rui. Faster rcnn-based detection of cervical spinal cord injury and disc degeneration. *Medical Physics*, 48(2):801–813, 2020.

[7] J.E. Small, P. Osler, A.B. Paul, and M. Kunst. Ct cervical spine fracture detection using a convolutional neural network. *Journal of Computer Assisted Tomography*, 45(4):578–585, 2021.

[8] D.T. Tuan, Q.H. Le, and T.H. Nguyen. *Cervical Spine Fracture Detection via Computed Tomography scan*. PhD thesis, FPT University, 2022.

[9] A.F. Voter, M.E. Larson, J.W. Garrett, and J.P.J. Yu. Diagnostic accuracy and failure mode analysis of a deep learning algorithm for the detection of cervical spine fractures. *AJNR Am J Neuroradiol*, 42(8):1550–1556, Aug 2021.

Advanced Ensemble Learning Framework for Reliable Smart Grid Stability detection

Saliha Mezzoudj[1], Yasmina Saadna[2], and Meriem Khelifa[3]

[1]*Department of Computer Science, University of Algiers, Algiers, Algeria ,* s.mezzoudj@univ-alger.dz

[2]*Labstic laboratory, Batna 2 University, Batna, Algeria ,* y.saadna@univ-batna2.dz

[3]*Artificial Intelligence of Information Technologies, Department of Computer Science and Information Technologies, University of Kasdi Merbah Ouargla, Algeria ,* khelifa.meriem@univ-ouargla.dz

Abstract

The increasing complexity of smart grid systems necessitates advanced methodologies to ensure reliable stability classification and seamless power delivery across consumer domains. This study introduces an innovative ensemble learning framework designed to classify smart grid stability using the Smart Grid Stability Augmented dataset. The proposed framework integrates multiple ensemble techniques, including Bagging, AdaBoost, Stacking, and Voting Classifiers, to improve robustness, accuracy, and reliability. A 5-fold cross-validation strategy is implemented to minimize overfitting and validate model performance. The dataset undergoes preprocessing with feature standardization and binary encoding of the target variable to ensure uniform contributions from all features. Experimental results indicate that the soft Voting Classifier, which is a combination of single machine learning models logistic regression, support vector machine, and random forest (LR+SVC+RF), outperforms other models by achieving a peak accuracy of 97.3%, demonstrating exceptional stability classification performance. Compared to individual machine learning models and existing state-of-the-art approaches, the proposed ensemble framework exhibits superior performance across multiple evaluation metrics. These results underscore the potential of ensemble learning in enhancing smart grid stability, contributing to more reliable and efficient power grid management systems.

Keywords: Grid Stability, Ensemble Learning, Bagging, AdaBoost, Soft Voting, stacking, Cross-Validation

1 Introduction

The smart grid is an advanced concept aimed at transforming the future electricity network by enhancing its flexibility,

adaptability, and autonomous management [20], [16]. This complex system incorporates various interconnected subsystems [16], integrating diverse disciplines and enabling the autonomous operation and control of its parts. It is geographically spread out and consists of a wide range of components. Additionally, the smart grid demonstrates emerging behaviors and continuous development. As a key element in a global network of linked systems, it encourages collaboration to promote the development of innovative services across different sectors. The primary factors propelling advancements in this field are energy efficiency and optimized resource management at both local and global levels, requiring comprehensive monitoring and control [12]. As electricity demand rises with population growth, the dependence on natural resources for power generation increases. Nevertheless, this process remains intricate and expensive. Significant research has been directed towards enhancing grid networks to improve power distribution efficiency. The smart grid offers a promising solution by leveraging Information and Communication Technology (ICT) to gather data on consumer behavior, thereby enabling the creation of context-aware systems that optimize power distribution efficiency [10]. Traditional stability analysis and control methods have proven insufficient for managing the complexities of modern smart grids. In response, recent advances in artificial intelligence (AI) provide effective tools to meet the high demands of security and stability in these systems [14]. The development of an intelligent grid that can accurately predict power demand is essential. This can be achieved through the application of Machine Learning (ML) algorithms [3], [6] to analyze the large amounts of data generated by the grid. These advancements in smart grid technology are crucial for reducing environmental pollution and lowering electricity costs, promoting a more cost-effective and sustainable energy system.

Recent developments in artificial intelligence (AI) and machine learning (ML) have significantly enhanced

the prediction and management of smart grid stability and energy systems. Oqaibi and Bedi (2024) introduced a hybrid forecasting system that integrates data deconstruction and attention mechanisms, achieving a prediction accuracy of 90.45% on the Kaggle dataset. Their work emphasizes the need for optimizing hybrid models to reduce computational complexity and improve prediction efficiency [15]. Xu et al. (2024) proposed a time-series depthwise separable convolutional neural network (CNN) with an attention mechanism, reaching 88.9% accuracy using the UCI dataset. Their study underscores the importance of large datasets for effectively training deep learning models [8]. Further contributions include Mohsen (2023), who developed an efficient artificial neural network (ANN) model for Decentralized Smart Grid Control (DSGC) systems, achieving a testing accuracy of 97.36% and a perfect AUC score of 100% through hyperparameter tuning [18]. Similarly, Alsirhani (2023) combined Multi-Layer Perceptron and Extreme Learning Machine (MLP-ELM) with Principal Component Analysis (PCA), attaining 95.8% accuracy, which highlights its potential for improving grid reliability amid fluctuating energy demands and growing renewable integration [1]. Javaid (2022) proposed a novel stacking ensemble model, MLBCSM, which combines multiple boosting classifiers (AdaBoost, XGBoost, HistBoost, CatBoost, LGBoost) with an Adaptive Synthetic Sampling Technique (ADASYN) to address data imbalance. The model, which includes data preprocessing, balancing, and classification, outperformed traditional methods, achieving 92.39% accuracy and 93.22% recall. These results demonstrate its effectiveness in detecting the stability in smart grids [14].
In this work, a novel ensemble-based machine learning approach is proposed to predict the stability of smart grids by classifying the Smart Grid Stability Augmented Dataset. The experimental results are compared with recent machine learning algorithms, including individual classifiers such as KNN, NB, Support Vector Classifiers, as well as ensemble methods like Bagging, AdaBoost, Stacking, and soft Voting Classifiers. The main steps involved in our contribution:
1. The Smart Grid Stability Augmented Dataset is loaded, and stability labels are mapped to binary values. The dataset is shuffled, and features are standardized to facilitate model convergence.
2. Four ensemble learning models Bagging, AdaBoost, Stacking, and Voting Classifiers are defined. These models utilise a variety of base learners, including Decision Trees, Logistic Regression, Random Forest, and Support Vector Classifier, to enhance predictive accuracy.
3. A 5-fold cross-validation strategy is employed to evaluate model performance, ensuring robust estimates of model effectiveness and mitigating overfitting risks.
4. Quantitative comparison of the models' performance is provided. The voting achieves the highest accuracy and AUC, while Stacking Classifier excels in integrating multiple models. AdaBoost and Bagging show strong performance in balancing precision and recall, and the Voting Classifier provides competitive results across all metrics.
with voting method reaching an accuracy of 97.3%, demonstrating superior predictive capability compared to individual models and other state-of-the-art models.

The rest of the paper is organized as follows. Section II discusses recent state-of-the-art literature related to the application of deep learning algorithms on smart grids. In Section III,the proposed model is discussed in detail. Experimental results are discussed in Section IV, which is followed by a conclusion and future work in Section V.

2 Proposed approach

A variety of machine learning algorithms can be applied to the problem of stability detection, with their effectiveness typically evaluated using metrics such as accuracy and false positive rates. To improve prediction performance and reduce false positives, researchers have proposed numerous ensemble learning methods. Ensemble learning techniques combines multiple machine learning algorithms to achieve enhanced predictive performance compared to standalone models [13]. Broadly, ensemble learning is categorized into two types: parallel and sequential [17] Parallel methods, such as bagging and random forests, train independent base classifiers to promote diversity, whereas sequential methods, including boosting, iteratively refine weak learners to improve accuracy. Ensemble methods are particularly robust and adaptable, excelling in scenarios involving noisy or complex data. This study introduces an ensemble learning framework to classify the stability of smart grid systems using the Smart Grid Stability Augmented dataset. The methodology incorporates various ensemble techniques to enhance the robustness, accuracy, and reliability of stability predictions. In this section, we describe the architecture

of our system as shown in Figure 1.

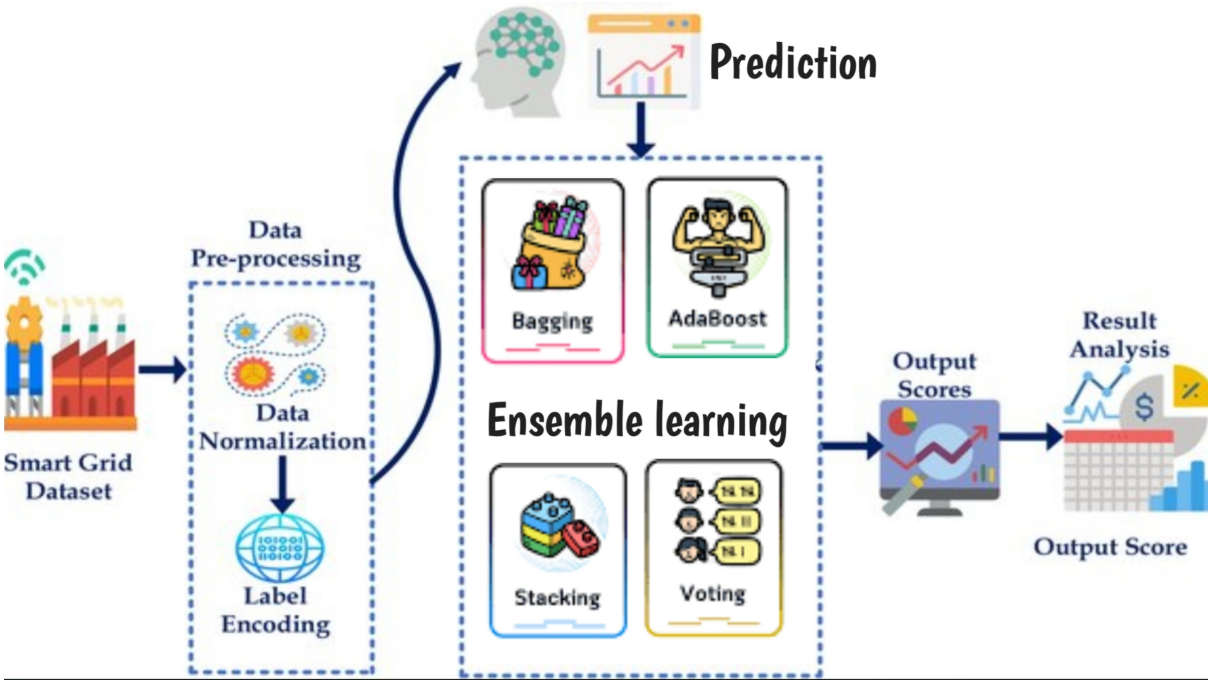


Figure 1: Architecture of the proposed system

The approach is structured as follows:

2.1 DATA PREPROCESSING

Pre-processing is a critical step in improving data quality and enhancing the performance of machine learning (ML) models. The Smart Grid Stability Augmented dataset includes features related to grid stability and a target label (stabf) that indicates stability (stable or unstable). The variability in feature ranges within the dataset can lead to biases, as features with higher magnitudes may dominate during model training. To address this, StandardScaler is employed for data normalization, ensuring all features contribute equally. This technique transforms the data into a common scale, thereby enhancing classifier performance. The categorical target variable is mapped to binary values, where: 0: : represents an unstable state and 1: indicates stability. Non-numeric values in the dataset are converted to numeric format using encoding techniques to make the data suitable for ML algorithms. Additionally, the dataset is shuffled to mitigate any ordering bias, further ensuring the robustness and reliability of the training process.

2.2 CROSS VALIDATION

To enhance model reliability and generalization, a 5-fold cross-validation strategy is employed. The dataset is divided into five equal parts, where each part is used as a validation set once, while the remaining four are used for training. This approach minimizes overfitting and ensures.

2.3 ENSEMBLE LEARNING METHODS

In this part, we explore the ensemble learning paradigm, focusing on its fundamental components, combination techniques for base learners, and methods for selecting ensembles.

2.3.1 Bagging (Bootstrap Aggregating) classifier

Bagging, or Bootstrap Aggregating, is an ensemble learning technique designed to reduce model variance and improve predictive accuracy by combining multiple base models [4], [21]. In this approach, each base model is trained on a distinct bootstrapped sample of the dataset, created by random sampling

with replacement. For this study, the **Decision Tree Classifier** is utilized as the base learner, with a total of 50 estimators. Each tree is independently trained on a bootstrapped sample, enabling the model to capture diverse patterns within the data. After training, predictions for unseen data are obtained by aggregating the outputs of all 50 models:

- For regression tasks, the final prediction is the *average* of all individual predictions.
- For classification tasks, the final prediction is determined by *majority voting* among the models.

This ensemble strategy significantly reduces the variance of the model compared to a single decision tree, leading to more stable and accurate predictions. The Bagging approach is particularly effective for noisy or complex datasets, such as those encountered in smart grid stability detection. By leveraging multiple models and aggregating their outputs, Bagging enhances the robustness and generalization ability of the framework, making it a reliable choice for high-stakes applications in smart grid systems.

2.3.2 AdaBoost (Adaptive Boosting) Classifier

In this study, the AdaBoost algorithm was chosen as the boosting method. Developed by Freund and Schapire [7], AdaBoost is one of the most widely used boosting techniques, offering a strong theoretical foundation and proven efficacy in generating accurate predictions. AdaBoost constructs a strong classifier by combining the weighted outputs of weak classifiers, addressing earlier boosting methods limitations. In our implementation of AdaBoost begins by initializing equal weights for all training samples. In each boosting round, a weak learner (in this case, a Decision Tree Classifier) is trained on the weighted dataset. The classifier's weighted error is calculated, and its performance is quantified using a weight α_t. This weight determines the importance of the weak learner in the final ensemble.

Misclassified samples are assigned higher weights, making them more influential in subsequent iterations. The process is repeated for T boosting rounds, where $T = 50$ in this implementation. The final prediction is made by combining the outputs of all weak classifiers, weighted by their respective importance values.

2.3.3 Stacking

Stacking is an ensemble learning technique that combines predictions from multiple base models (level-0 models) and refines them using a meta-model (level-1 model) [3]. The primary goal is to leverage the strengths of individual models and optimize the final prediction by training an additional layer. In our implementation:

- **Base Models:** Random Forest and Support Vector Classifier are trained independently on the training dataset. Each model generates predictions, capturing unique patterns within the data.
- **Meta-Model:** Logistic Regression is used as a second-level model, which takes the predictions of the base models as input. It learns to combine these predictions optimally, mitigating individual weaknesses.
- **Workflow:** The training process involves generating predictions for the validation set using the base models, constructing a new dataset comprising these predictions, and training the meta-model on this dataset. During inference, the base models generate predictions for unseen data, which are then aggregated by the meta-model to produce the final output.

2.3.4 Voting Classifier Algorithm (Soft Voting)

The voting Classifier is an ensemble learning method that combines predictions from multiple base models to improve predictive accuracy and robustness [9]. In the context of this study, Soft Voting is used, which involves averaging the predicted probabilities from each of the base models. The base models utilized in this framework include:

- Random Forest
- Support Vector Classifier (SVC)
- Logistic Regression

Algorithm 1 presents the implementation details of the Soft Voting Classifier. This algorithm integrates multiple base models by averaging their predicted class probabilities, ensuring a more robust final prediction. The base models utilized in this study include Random Forest, Support Vector Classifier (SVC), and Logistic Regression. Each model generates probability estimates for each class, and the final prediction is determined by selecting the class with the highest average probability.

Algorithm 1 Voting Classifier Algorithm (Soft Voting)

Input:
Training dataset $D = \{X, y\}$, where X denotes feature vectors and y represents class labels.
Base Models: Random Forest, Support Vector Classifier, and Logistic Regression.
Soft Voting aggregation scheme.
Output: Final predicted class labels $\hat{y}_{\text{final}}$.
Step 1: Train each base model on the training dataset D.
$model_1 \leftarrow$ Train Random Forest on D.
$model_2 \leftarrow$ Train Support Vector Classifier on D.
$model_3 \leftarrow$ Train Logistic Regression on D.
Step 2: For each test instance x_{test}, obtain probability estimates from each model:
$p_1 \leftarrow$ Probability prediction from $model_1$ for x_{test}.
$p_2 \leftarrow$ Probability prediction from $model_2$ for x_{test}.
$p_3 \leftarrow$ Probability prediction from $model_3$ for x_{test}.
Step 3: Compute the average probability for each class:
$P_{\text{avg}}(c_k) = \frac{1}{3}(p_1(c_k) + p_2(c_k) + p_3(c_k))$ for each class c_k.
Step 4: Assign the class with the highest averaged probability as the final prediction:
$\hat{y}_{\text{final}} = \arg\max_k P_{\text{avg}}(c_k)$, where k denotes the class label index. =0

3 Results and Discussion

This section presents the results from the experiments conducted to assess the performance of the proposed ensemble learning approach for smart grid stability detection, using the Smart Grid Stability Augmented dataset comprising 60,000 samples.
The experiments were executed on Google Colab, utilizing an online GPU service, with additional processing on a personal computer running Linux OS and an Intel Core i5 processor. Python 3.7 and libraries such as `scikit-learn` and `pandas` were employed for model implementation and evaluation.
The dataset, sourced from the UCI Machine Learning Repository [5], consists of 60,000 instances and 14 attributes related to factors influencing smart grid stability. The target variable indicates system stability with binary labels: 0 for unstable and 1 for stable. The performance of the model was evaluated using several metrics. Accuracy was calculated as the ratio of correct predictions (True Positives and True Negatives) to total instances. Precision measured the accuracy of predicted stable instances, while Recall assessed the proportion of actual stable instances correctly identified. The F1 Score, as the harmonic mean of Precision and Recall, provided a balanced measure, and Specificity evaluated the proportion of correctly predicted unstable instances. These metrics collectively evaluate the overall effectiveness of the ensemble learning model for smart grid stability detection.

The results presented in Table 1 demonstrate the performance of various ensemble learning models in detecting smart grid stability. The models evaluated include Bagging, AdaBoost, Stacking, and soft Voting classifiers, each exhibiting different strengths in terms of accuracy, precision, recall, F1 score, cross-validation accuracy, and ROC AUC. The Bagging classifier performed well with an accuracy of 91.6%, but lagged behind in precision and recall, suggesting challenges in accurately identifying stable and unstable grid states. AdaBoost showed improved precision and F1 score over Bagging, but still had a lower recall, meaning it missed some stable instances.
The Stacking classifier, which combines multiple models, achieved the highest performance with an accuracy of 95.5%, precision of 94.3%, and ROC AUC of 99.3%, demonstrating strong generalization and the ability to discriminate grid stability effectively. The Voting classifier performed the best overall, with an accuracy of 97.3%, precision of 96.6%, and an impressive ROC AUC of 99.7%. This model's performance was bolstered by combining multiple classifiers in a soft-voting scheme, making it highly robust for smart grid applications.

As shown in Figure 2, the Voting and Stacking classifiers emerged as the top performers, offering the best accuracy and ROC AUC scores. These models are particularly suited for real-time smart grid stability detection, where high precision, recall, and robustness are critical. The 5-cross-validation accuracy values for all methods indicate stability in model performance across different subsets of the data, further validating the models' reliability and generalization.

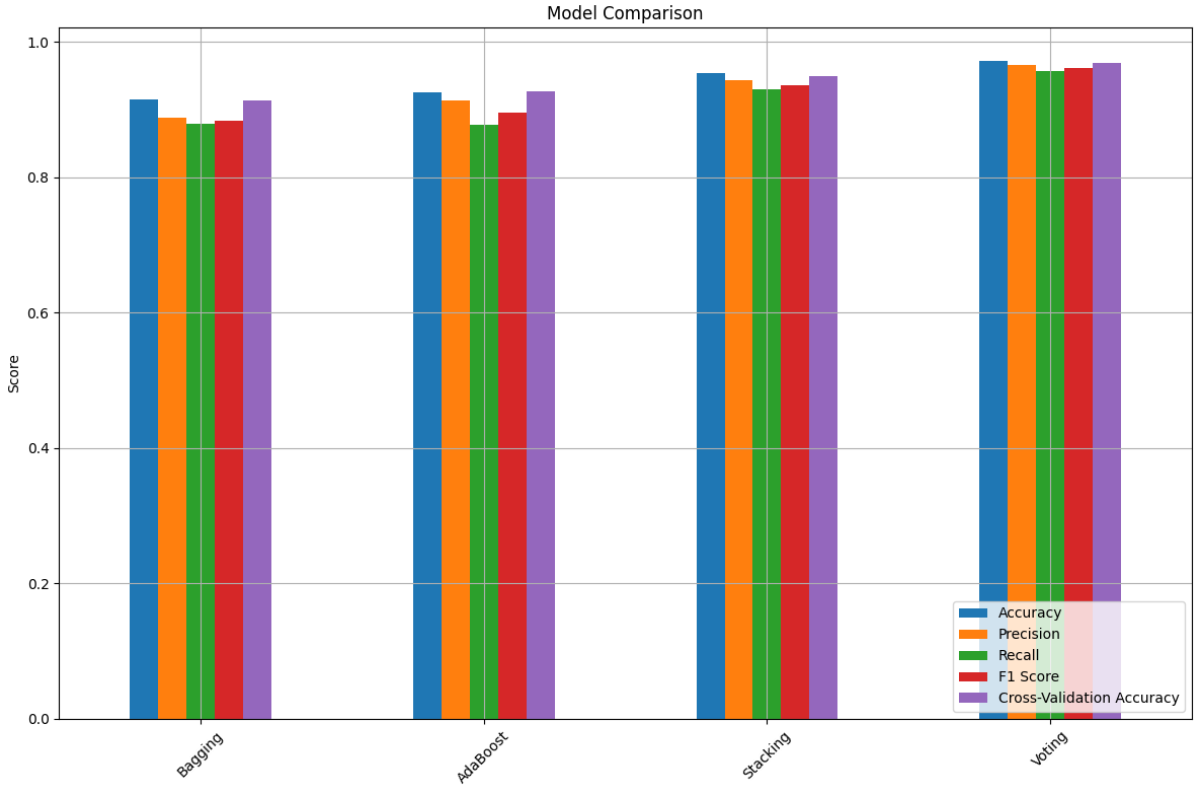


Figure 2: Comparison between ensemble learning methods using 5-cross validation

From Figure 3, the analysis of the learning curves reveals a consistent trend of convergence toward high performance across all ensemble methods, showcasing their strong generalization capabilities. The performance for all models indicate that ensemble techniques effectively handle the complexity of the classification task, leading to optimal predictions. Notably, Stacking and Voting models outperform other methods in most metrics, demonstrating their robust ability to combine diverse features for superior results. Bagging and AdaBoost, while slightly behind in some metrics, still deliver highly competitive performances, further reinforcing the effectiveness of ensemble learning in enhancing classification tasks.

Table 1: Model Comparison Results

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
Bagging	0.916	0.888	0.879	0.883	0.975
AdaBoost	0.926	0.914	0.878	0.895	0.983
Stacking	0.955	0.943	0.930	0.937	0.993
Voting	0.973	0.966	0.958	0.962	0.997

4 Comparison with Existing Approaches

In this study, we evaluated the performance of our proposed ensemble models for smart grid stability detection, comparing them with existing methods based on key metrics such as accuracy, precision, recall, and F1 score as shown in Table 2 . Previous techniques like CART (80.0%), XGBoost (97.82%), and stacking ensemble models (92.395%) showed competitive results, with Mohsen et al. (2023) achieving the highest accuracy of 97.36% using an ANN-based MLP model. Our ensemble models demonstrated

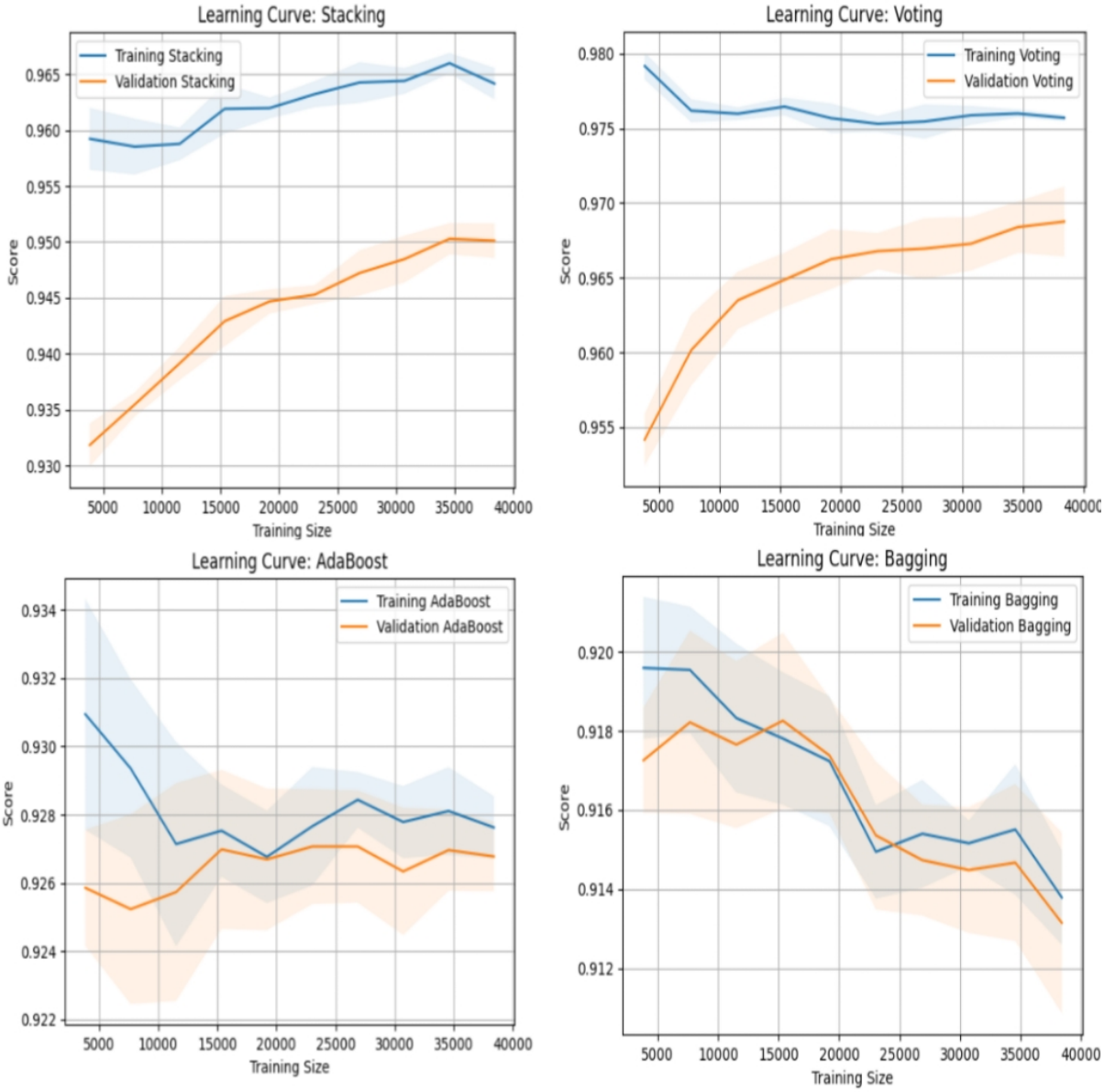


Figure 3: Learning curves (training and testing) of each model

Table 2: Comparison of Our Proposed Approach with Existing Approaches

Year	Reference	Prediction Technique	Accuracy (%)	Precision	Recall	F1 Score
2018	Arzamasov et al. [19]	CART	80.0	–	–	–
2019	Chen et al. [11]	XGBoost	97.82	–	–	–
2022	Javaid et al. [14]	Stacking ensemble model, MLBCSM	92.395	–	93.222	–
2023	Alsirhani et al. [1]	MLP-ELM	95.8	–	–	–
2023	Mohsen et al. [18]	ANN based on MLP	97.36	98.02	98.03	98.02
2024	Alessandro et al. [8]	GAN-GRID	90.45	–	–	–
2024	**Single Model (SVM)** [2]	SVM	81.0	0.869	0.810	0.843
2024	**Single Model (KNN)** [2]	KNN	82.3	0.881	0.823	0.855
2024	**Single Model (DT)** [2]	Decision Tree (DT)	83.4	0.891	0.834	0.866
2024	**Single Model (MLP)** [2]	MLP	84.3	0.897	0.843	0.875
2024	**Single Model (RF)** [2]	Random Forest (RF)	87.4	0.917	0.874	0.900
2025	**Our Model (Bagging)**	Bagging	**91.6**	0.888	0.879	0.883
2025	**Our Model (AdaBoost)**	AdaBoost	**92.6**	0.914	0.878	0.895
2025	**Our Model (Stacking)**	Stacking	**95.5**	0.943	0.930	0.937
2025	**Our Model (Soft Voting)**	Soft Voting	**97.3**	0.966	0.958	0.962

significant improvements. The Bagging model achieved 91.6% accuracy, AdaBoost improved to 92.6%, and Stacking reached 95.5%. The Voting classifier outperformed all, with 97.3% accuracy and impressive precision (0.966), recall (0.958), and F1 score (0.962). These results highlight the effectiveness of ensemble learning in enhancing smart grid stability detection, with advanced methods like Stacking and Voting delivering superior accuracy and balanced performance, showcasing the power of combining diverse models for optimal stability detection in smart grid applications.

5 Conclusion

In this study, we have proposed an ensemble learning framework for smart grid stability detection, leveraging techniques such as Bagging, AdaBoost, Stacking, and soft Voting Classifiers. These methods were evaluated on the Smart Grid Stability Augmented dataset, demonstrating their robustness, accuracy, and reliability in predicting grid stability. Experimental results showed that while simpler models such as Bagging and AdaBoost provide competitive results, more advanced ensemble methods, including Stacking and soft Voting, significantly outperform them, offering higher accuracy, precision, recall, and F1 scores. In particular, the Voting classifier achieved the best overall performance, showcasing the benefits of combining multiple strong classifiers to improve prediction reliability. The results indicate that ensemble learning techniques are well-suited for smart grid stability detection, providing a reliable and scalable solution for ensuring grid stability in real-world applications. Future research could focus on enhancing the proposed framework by integrating it with real-time data from smart grid systems to assess its adaptability in dynamic and evolving environments. Furthermore, exploring the use of deep learning models and hybrid ensemble techniques may offer additional performance improvements. Investigating the incorporation of feature selection or dimensionality reduction methods could also enhance model efficiency and computational scalability.

References

[1] A. Abukwaik A. I. Taloba R. M. Abd El-Aziz A. Alsirhani, M. M. Alshahrani and M. Salem. A novel approach to predicting the stability of the smart grid utilizing mlp-elm technique. *Alexandria Engineering Journal*, 74:495–508, 2023.

[2] B. Prabadevi N. Deepa W. S. Alnumay T. R. Gadekallu A. K. Bashir, S. Khan and P. K. R. Maddikunta. Comparative analysis of machine learning algorithms for prediction of smart grid stability. *International Transactions on Electrical Energy Systems*, 31(9):e12706, 2021.

[3] B. K. Bose. Artificial intelligence techniques in smart grid and renewable energy systems—some example applications. In *Proceedings of the IEEE*, volume 105, pages 2262–2273, Nov. 2017.

[4] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[5] D. Dua and C. Graff. Uci machine learning repository: Smart grid stability augmented dataset. Online, 2019.

[6] F. Un-Noor S. S. Sikander E. Hossain, I. Khan and M. S. H. Sunny. Application of big data and machine learning in smart grid, and associated security concerns: A review. *IEEE Access*, 7:13960–13988, Jan. 2019.

[7] Y. Freund and R. E. Schapire. A decision-theoretic generalization of online learning and an application to boosting. 1995.

[8] X. Liang-et al. H. Xu, F. Hu. A framework for electricity load forecasting based on attention mechanism time series depthwise separable convolutional neural network. *Energy*, 299:131258, 2024.

[9] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Comput. Eng. Inf. Sci.*, 108(2):212–261, Feb. 1994.

[10] S. S. R. Krishnan Q. V. Pham M. P. K. Reddy M. Alazab, S. Khan and T. R. Gadekallu. A multidirectional lstm model for predicting the stability of a smart grid. *IEEE Access*, 8:85454–85463, 2020.

[11] S. Chen Y. Liu C.-H. Zhang M. Chen, Q. Liu and R. Liu. Xgboost based algorithm interpretation and application on post-fault transient stability status prediction of power system. *IEEE Access*, 7:13149–13158, 2019.

[12] M. W. Maier. Architecting principles for systems-of-systems. *Systems Engineering*, 1(4):267–284, 1998.

[13] I. D. Mienye and Y. Sun. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access*, 10:99129–99149, 2022.

[14] A. Aldegheishem N. Alrajeh N. Javaid, M. Akbar and E. A. Mohammed. Employing a machine learning boosting classifiers based stacking ensemble model for detecting non-technical losses in smart grids. *IEEE Access*, 10:121886–121899, 2022.

[15] H. Oqaibi and J. Bedi. A data decomposition and attention mechanism-based hybrid approach for electricity load forecasting. *Complex Intelligent Systems*, 24:1–16, 2024.

[16] T. Erdem P. Breviglieri and S. Eken. Predicting smart grid stability with optimized deep models. *SN Computer Science*, 2:1–12, 2021.

[17] L. Rokach. Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Comput. Statist. Data Anal.*, 53(12):4046–4072, 2009.

[18] H. Kotb M. Pushkarna S. Alphonse S. Mohsen, M. Bajaj and S. S. Ghoneim. Efficient artificial neural network for smart grid stability prediction. *International Transactions on Electrical Energy Systems*, 2023(1):9974409, 2023.

[19] K. Bohm V. Arzamasov and P. Jochem. Towards concise models of grid stability. In *Proc. Int. Conf. Commun. Control Comput. Technol. Smart Grids*, pages 1–6, Aalborg, Denmark, Oct. 2018.

[20] Z. Li L. Zeng Y. Zhao R. Zhang et al. Z. Shi, W. Yao. Artificial intelligence techniques for stability analysis and control in smart grids: Methodologies, applications, challenges and future directions. *Applied Energy*, 278:115733, 2020.

[21] Z.-H. Zhou. Ensemble learning. In *Encyclopedia of Biometrics, Volume 1*, pages 270–273. Springer, Berlin, 2009.

Unmasking Deepfakes: CNNs and Vision Transformers for Cutting-Edge Detection

Abdelhalim Saadi[1], Yacine Slimani[2], Ridha Louze[3], and Roufaida Hammadou[4]

[1] ***Faculty of technology Setif 1 University – Ferhat Abbas Setif, Algeria ,*** *halim.saadi@gmail.com*
[2] ***Faculty of technology Setif 1 University – Ferhat Abbas Setif, Algeria,*** *slimany09@gmail.com*
[3] ***Faculty of NTIC University of Abdelhamid Mehri – Constantine 2, Algeria,*** *ridha.louze@univ-constantine2.dz*
[4] ***Faculty of NTIC University of Abdelhamid Mehri – Constantine 2, Algeria,*** *roufaida.hammadou@univ-constantine2.dz*

Abstract

Deepfake technology, driven by Generative Adversarial Networks (GANs), poses challenges in digital security by enabling highly realistic synthetic media. This study proposes a detection framework combining Convolutional Neural Networks (CNNs) for local feature extraction and Vision Transformers (ViTs) for global analysis. Evaluated on two datasets, CNNs achieved 97.1 % accuracy on a 140K-image dataset, outperforming ViTs at 90.06%, though ViTs showed better generalization. Despite these advances, deepfake detection faces challenges like adversarial attacks and dataset biases. Future work will enhance real-time processing, robustness, and multi-modal approaches integrating audio and behavioral cues.

Keywords: Deep Learning (DL), Deepfake Detection, Convolutional Neural Networks (CNN), Vision Transformers (ViT), Generative Adversarial Networks (GAN).

1 Introduction

The rise of deepfake technology has brought significant challenges to digital media security, enabling the creation of highly realistic synthetic images and videos. While deepfakes have applications in entertainment and creative industries, they also pose serious threats, including misinformation, identity fraud, and political manipulation. The increasing sophistication of Generative Adversarial Networks (GANs) has made detecting manipulated media more complex, necessitating advanced detection techniques. This study proposes a deepfake detection framework leveraging Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) to enhance classification accuracy. Using two datasets—a 140K Real and Fake Faces dataset and a smaller Real and Fake Face Detection dataset—the models are trained and evaluated based on accuracy, loss, and robustness. The paper is structured as follows: Section 2 reviews deepfake generation and detection techniques, Section 3 presents the methodology, Section 4 discusses experiments and results, and Section 5 concludes with future perspectives. This research contributes to the ongoing efforts to strengthen digital media security and combat deepfake threats. [**?**].

2 Review methodology and literature

Deepfake technology, powered by deep learning and artificial intelligence, has rapidly evolved in recent years, leading to highly realistic synthetic media that are often indistinguishable from authentic content. While this techno-logy has numerous positive applications in entertainment and creative industries, it also poses significant risks, including misinformation, identity fraud, and political manipulation. This section provides a comprehensive overview of deepfake generation techniques, detection methodologies, and the current challenges in combating manipulated media.

2.1 Deepfake Generation Techniques

Deepfake generation has rapidly evolved with advancements in deep learning, particularly through the use of Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). These techniques allow the synthesis of highly realistic images, videos, and audio that are increasingly difficult to

distinguish from authentic media. This section provides an overview of the most prominent deepfake generation methods and their impact on digital media.

- **Generative Adversarial Networks (GANs)** Generative Adversarial Networks (GANs) are among the most widely used architectures for deepfake generation. A GAN consists of two competing neural networks: a generator that produces synthetic media and a discriminator that attempts to distinguish between real and generated content. Through iterative training, the generator improves its ability to create highly realistic outputs. Advanced variations, such as StyleGAN and StyleGAN2, have significantly enhanced the quality of generated images by enabling fine-grained control over facial features, expressions, and lighting conditions [1]. Recent studies have also explored the potential of Latent Flow Diffusion (LFD), which incorporates optical flow sequences in the latent space to enhance temporal coherence in deepfake videos [11]. Compared to conventional GANs, LFD provides better preservation of spatial and motion consistency, making generated videos appear more authentic.

- **Variational Autoencoders (VAEs) and Hybrid Models** Variational Autoencoders (VAEs) are another class of generative models used in deepfake generation. Unlike GANs, which rely on adversarial training, VAEs learn a probabilistic representation of data to generate realistic samples. They have been particularly effective in face-swapping applications, where they enable smooth blending of facial features while maintaining structural consistency [15]. Hybrid models combining GANs and VAEs have also gained traction. These models leverage the structured latent space of VAEs with the adversarial refinement of GANs to generate higher-quality deepfakes. The integration of attention mechanisms within these architectures has further improved the realism of generated media by focusing on fine details such as skin texture and micro-expressions [2].

- **Face manipulation techniques** Face manipulation techniques in deepfake generation can be categorized into three main types: Face Swapping: This technique replaces the face of a person in a video with another person's face while maintaining the original facial expressions and movements. It is commonly implemented using autoencoders and GANs. The DF-Platter dataset [12] demonstrates that face-swapping deepfakes can be generated at both high and low resolutions, highlighting the challenges in detection. Facial Attribute Manipulation: This method alters specific facial features such as age, gender, and expressions. It is achieved using models like StarGAN and AttGAN, which modify targeted attributes while preserving the overall identity of the subject [2]. Such manipulations are widely used in applications ranging from entertainment to identity anonymization. Lip-Sync Manipulation: This technique synchronizes lip movements with an audio track, making it appear as though a person is speaking words they never actually said. Models like Wav2Lip and SyncGAN have demonstrated impressive results in creating realistic lip-sync deepfakes, posing significant challenges in forensic detection [15].

- **Text-to-Image and Text-to-Video Synthesis** With the advent of large-scale generative models, deepfake generation has extended beyond face manipulation to full-body synthesis. Text-to-image and text-to-video synthesis models, such as DALL•E and Stable Diffusion, enable the creation of highly realistic synthetic content based on textual descriptions. These models use diffusion processes to iteratively refine images, resulting in high-fidelity outputs that can be used for both benign and [14]. Furthermore, recent research has explored deepfake phylogeny, which examines how iterative manipulations can evolve deepfakes over multiple generations, leading to increasingly deceptive synthetic media [13]. The DeePhy dataset was developed to study the progression of deepfakes and their impact on detection algorithms.

- **Challenges in Deepfake Generation** While deepfake generation techniques have significantly improved, they present substantial ethical and security concerns. The ability to create highly realistic synthetic media has raised issues related to misinformation, identity fraud, and political propaganda. The development of novel detection techniques must keep pace with advancements in generation methods to mitigate potential risks [20]. Moreover, existing deepfake generation models often suffer from limitations such as excessive computational requirements, data dependency, and difficulty in generating highly dynamic scenes. Researchers are exploring ways to enhance the efficiency and realism of these models while addressing concerns related to misuse and ethical responsibility [4]. Deepfake generation techniques have advanced rapidly with the integration of GANs, VAEs, and hybrid models. Face manipulation methods such as face swapping, attribute manipulation, and lip-syncing have reached new levels of realism, making detection increasingly

challenging. The emergence of text-to-image and text-to-video synthesis models has further expanded the capabilities of deepfake technology. However, as generation methods evolve, the need for robust and adaptive detection frameworks becomes more critical. Future research must focus on improving the interpretability of generative models, developing counter-measures against adversarial attacks, and ensuring the ethical use of deepfake technology.

2.2 Deepfake Detection Techniques

As deepfake generation techniques continue to evolve, detecting these synthetic manipulations has become an essential challenge in digital media security. Various deep learning-based approaches, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformer-based models, have been developed to distinguish real content from manipulated media. This section provides an overview of state-of-the-art deepfake detection techniques and their effectiveness in different application domains.

- **Convolutional Neural Networks (CNNs) for Image and Video Detection** Convolutional Neural Networks (CNNs) have been widely adopted for deepfake detection due to their ability to extract spatial features from images and videos. CNN-based models analyze inconsistencies in pixel distributions, texture artifacts, and facial asymmetries that may not be perceptible to the human eye. Studies have shown that CNNs, particularly Xception and MobileNet architectures, achieve high accuracy in detecting face-swapping deepfakes, with results ranging between 91% and 98% depending on the dataset used [7]. Despite their effectiveness, CNN-based models face challenges when applied to real-world deepfakes. These models often struggle with generalization across different datasets due to biases introduced during training. Additionally, CNNs primarily focus on spatial features, making them less effective in detecting temporal inconsistencies in deepfake videos [17].

- **Recurrent Neural Networks (RNNs) and Temporal Analysis** To address the limitations of CNNs in video deepfake detection, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have been utilized for temporal analysis. These models analyze sequential frames in a video to detect unnatural facial movements, such as inconsistent blinking patterns or unnatural lip-syncing [9]. The use of spatiotemporal convolutional networks has further enhanced the capability of RNN-based approaches. For example, the Celeb-DF dataset benchmark demonstrated that incorporating temporal features significantly improves detection accuracy, outperforming frame-based detection models [10]. However, these approaches remain computationally expensive and require substantial processing power, limiting their feasibility for real-time applications.

- **Transformer-Based Models for Deepfake Detection** Recent advancements in deep learning have led to the adoption of Transformer-based models for deepfake detection. Vision Transformers (ViTs) leverage self-attention mechanisms to capture both local and global dependencies in an image, making them highly effective in detecting subtle deepfake artifacts. Multi-modal Transformer architectures, such as M2TR, integrate RGB and frequency-domain features to improve detection accuracy [10]. Compared to CNNs and RNNs, Transformer-based models demonstrate superior generalization capabilities across different datasets. They are particularly effective in detecting complex deepfakes that incorporate high-quality synthesis techniques. However, their high computational cost remains a challenge, necessitating further research into optimization techniques for practical deployment [6].

- **Multi-Modal Deepfake Detection Approaches** Multi-modal deepfake detection approaches integrate information from multiple sources, such as visual and auditory cues, to enhance detection robustness. Joint audio-visual deepfake detection has been proposed as an effective strategy, leveraging synchronization inconsistencies between speech and facial expressions [22]. These methods have shown promising results in identifying lip-sync deepfakes and voice-cloning manipulations. In addition to audio-visual synchronization, PRNU (Photo-Response Non-Uniformity)-based methods have been explored for deepfake detection. PRNU, commonly used in digital forensics, identifies unique device fingerprints left during the image capture process. Recent studies indicate that PRNU-based approaches can complement deep learning models in hybrid detection frameworks [8].

- **Challenges in Deepfake Detection** Despite advancements in deepfake detection, several challenges remain: Generalization Across Different Datasets: Most deepfake detection models struggle with dataset-specific biases. Methods trained on one dataset often fail to generalize well to unseen deepfakes generated by different techniques [3]. Adversarial Robustness: Adversarial attacks can be used to fool deepfake detection models by introducing imperceptible perturbations. This highlights the need for more robust adversarial training strategies [16]. Real-Time Processing Efficiency: Many state-of-the-art detection models are computationally intensive, making real-time deepfake detection a significant challenge [18].
- **Future Directions in Deepfake Detection** To improve deepfake detection, future research should focus on: Hybrid Detection Models: Combining CNNs, RNNs, and Transformer-based models to leverage their respective strengths. Few-Shot and Zero-Shot Learning: Reducing reliance on large labeled datasets to enhance detection generalization [21]. Blockchain and Forensic Watermarking: Implementing digital watermarking techniques to verify content authenticity and track manipulations [5].

3 Contribution

This section presents the methodology adopted for deepfake detection. It begins with a description of the proposed project, followed by the system architecture and the development process of the models used. The chapter also includes details on the dataset, implementation, and performance evaluation of the deep learning models.

3.1 Project Description

Our proposed project consists of two main phases: the generation phase using GANs and the detection phase, where we evaluate two efficient deep learning models—CNN and ViT—to differentiate between real and fake images.

- **Generation Phase (Using GANs) :** In the generation phase, fake images are created using a GAN architecture, which comprises two adversarial neural networks: a generator and a discriminator. The generator takes a random latent vector as input and produces synthetic images, which are then passed to the discriminator. The discriminator, which has access to both real and generated images, is trained to distinguish between them, thereby forcing the generator to improve its ability to create realistic images. This generation process is crucial because deepfake detection models rely on deep learning, which requires large datasets for accurate predictions. However, many existing datasets suffer from low image resolution and are too small to effectively train advanced models like ViT.
- **Detection Phase (Using CNN & ViT Models)** For the detection phase, both the CNN and ViT models receive an image as input. Before processing, the images go through a data preprocessing step to ensure optimal training and testing conditions. The dataset is then split into training and testing sets and fed into either the CNN or ViT model. Once training is complete, we evaluate the model's performance and save the trained model for real-world predictions. The trained models can then analyze new images and determine whether they are real or fake. The proposed system follows a structured pipeline, as illustrated in the system architecture diagram, which includes both the generation and detection phases, ensuring a robust and efficient deepfake detection approach (See Figure 1).

Figure 1.

3.2 Deep Convolutional GAN (DCGAN) Development

The Deep Convolutional Generative Adversarial Network (DCGAN) is used for generating fake images. It consists of two main components:

- **The Discriminator Model** The first step is to define the discriminator model. The model must take a sample image from our dataset as input and output a classification prediction as to whether the sample is real or fake. This is a binary classification problem:

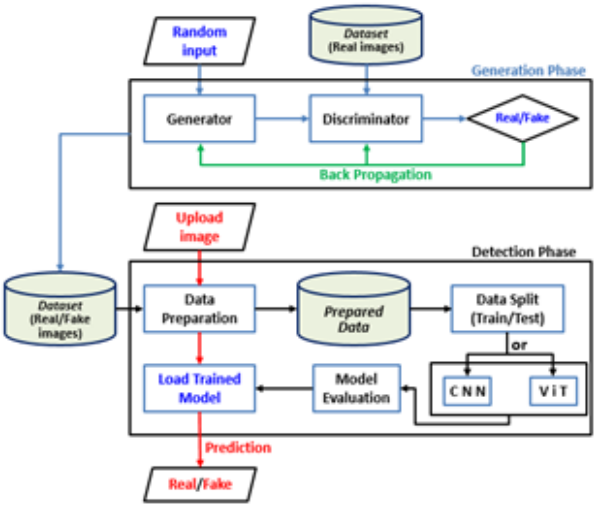


Figure 1: Workflow of the proposed system

1. Inputs: An image with one channel and a resolution of 256 × 256 pixels.
2. Outputs: A binary classification, where the model predicts the likelihood that the input image is real or fake. The discriminator architecture consists of:
3. Five convolutional layers (Conv2D), each followed by:
   (a) LeakyReLU activation (instead of ReLU) to allow better gradient flow.
   (b) Batch normalization to stabilize training.
   (c) Dropout layers to prevent overfitting.
4. A final dense layer with a sigmoid activation function, which outputs a probability score.

A final dense layer with a sigmoid activation function, which outputs a probability score. The model is trained using the binary cross-entropy loss function, with the Adam optimizer (learning rate = 0.00015, momentum = 0.5) to ensure stability.

- **The Generator Model** The generator is responsible for creating fake images. It takes a latent vector (random noise) as input and transforms it into a realistic image through a series of upsampling layers.

  1. Inputs: A 100-dimensional latent space vector sampled from a Gaussian distribution.
  2. Outputs: A three-channel (RGB) image of 256 × 256 pixels with values normalized between [0,1]. The generator architecture consists of:
  3. A Dense layer that expands the latent vector into a lower-resolution feature map.
  4. Reshaping and upsampling layers to progressively increase the spatial resolution.
  5. Several transposed convolutional layers (Conv2DTranspose), each followed by:
     (a) Batch normalization to improve stability.
     (b) LeakyReLU activation for non-linearity.
  6. A final Conv2D layer with a sigmoid activation function, ensuring the output image values remain within the valid range.

- **GAN Model (Combining Generator & Discriminator)** Once both the generator and discriminator are defined, they are combined to form a complete GAN model. The training process follows these steps:

  1. The generator creates a batch of fake images from random latent vectors.
  2. These fake images are passed to the discriminator, along with real images from the dataset.
  3. The discriminator predicts whether each image is real or fake.

4. Backpropagation is applied, updating both the generator and discriminator weights to improve their respective performances.
5. This adversarial training continues until the generator produces highly realistic images that can fool the discriminator.

By iteratively refining the generator and discriminator, the GAN model learns to generate increasingly convincing fake images, which are later used to train the deepfake detection models. A plot of the model is also created and we can see that the model expects a 100-element point in latent space as input and will predict a single output classification label.

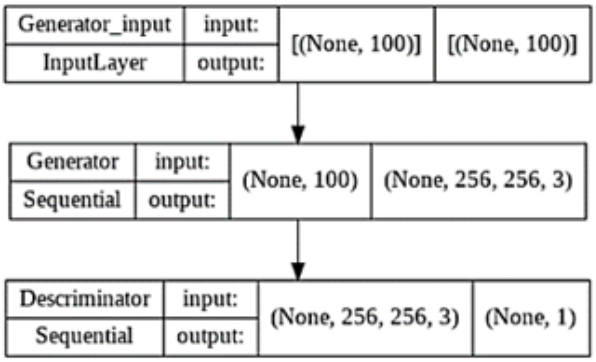


Figure 2: Plot of the Composite Generator and Discriminator model in the GAN

3.3 Process Development of DeiT (Data-efficient Image Transformer)

The DeiT (Data-efficient Image Transformer) model is an optimized version of the Vision Transformer (ViT), designed for efficient training on smaller datasets. Unlike Convolutional Neural Networks (CNNs), which rely on convolutional layers to extract local features, DeiT leverages the self-attention mechanism to capture both local and global dependencies within an image. This characteristic enables it to recognize complex patterns and structural inconsistencies that may indicate deepfake manipulations. The development process of DeiT follows steps:

- Linear Embedding Layer:
  - The input image is split into fixed-size patches (e.g., 16×16 pixels).
  - Each patch is flattened and mapped into a high-dimensional feature space through a learned embedding matrix.
  - A learnable classification token is added to the sequence, and positional encodings are introduced to preserve spatial relationships.
- Transformer Encoder:
  - The sequence of image patches passes through L identical layers, each containing:
  - A Multi-Head Self-Attention (MSA) mechanism, which enables the model to analyze relationships between patches.
  - A Feed-Forward Network (MLP) that applies non-linear transformations for feature enhancement.
  - Layer Normalization (LN) and skip connections to stabilize training and improve information retention.
- Multi-Head Self-Attention (MSA) Mechanism:
  - Computes attention between all patches, allowing the model to focus on important regions of the image.
  - Uses Query (Q), Key (K), and Value (V) matrices to determine the weight of each patch in the final representation. Classification and Output:
  - After passing through multiple transformer layers, the classification token is extracted.

- A fully connected layer is applied to classify the image as real or fake.

DeiT offers a powerful alternative to CNNs for deepfake detection, particularly when dealing with large datasets. However, due to its reliance on large-scale training data, its performance can be impacted when applied to smaller datasets. In this study, CNN demonstrated higher accuracy on limited data, while DeiT showed better scalability and generalization potential for future deepfake detection improvements [19].

4 Experiments and Results

This section describes the implementation process and the experiments conducted to evaluate the proposed deepfake detection model. The implementation consists of dataset preparation, model training, performance evaluation, and final deployment.

1. **Dataset** The experimentation of the proposed technique is implemented by using the two datasets : The first one is the "140k real and fake faces" dataset contains 70k real faces from the Flickr dataset collected by Nvidia, as well as 70k fake faces sampled from 1 million fake faces (generated by style GAN) [1] . The second is "Real and fake face detection" datasets contain two subfolders training real and training fake. Training real contains 1081 images and training fake contains 960 images, the total dataset is 2041 images [2]

2. **Parameter Settings** The models were trained using the following hyperparameters:

Table 1: PARAMETER SETTINGS

Model	Epochs	Batch size	Activation	Optimizer
CNN	20	64	Sigmoid	Adam
DeiT-Tiny	20	32	ReLU	Adam

3. **Performance Evaluation and Discution** The performance evaluation of the CNN and DeiT-Tiny models was conducted using key metrics such as training accuracy, validation accuracy, training loss, and validation loss, as summarized in Table II. The results highlight that CNN outperformed DeiT-Tiny, especially on the smaller dataset, achieving 94.15% validation accuracy on the 140K dataset and 81.88% on the Real and Fake Face Detection dataset. In contrast, DeiT-Tiny reached 90.31% and 61.27%, respectively, indicating its difficulty in learning from limited data and reliance on larger training sets for optimal performance.

Table 2: PERFORMANCE EVALUATION TABLE

Model	Dataset	Train accuracy	Validation accuracy	Train loss	Validation loss
CNN	140K Real and Fake Faces	97.11 %	94.15%	7.47%	14.50%
CNN	Real and Fake Face Detection	94.58%	81.88%	22.72%	43.95%
DeiT-Tiny	140K Real and Fake Faces	90.06%	90.31%	37.20%	37.01%
DeiT-Tiny	Real and Fake Face Detection	85.05%	61.27%	43.89%	68.44%

Figure 4 and 5 illustrate the CNN model's stability, with smooth loss curves and consistently high accuracy, confirming its reliability in deepfake detection. Figure 6 and Figure 7 depict the DeiT-Tiny model's slower convergence and higher validation loss, suggesting greater training data requirements for stable results. Despite its generalization potential, DeiT-Tiny struggled with small datasets, whereas CNN demonstrated robust and reliable performance across both datasets. Overall, these findings confirm

[1] https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection
[2] https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces

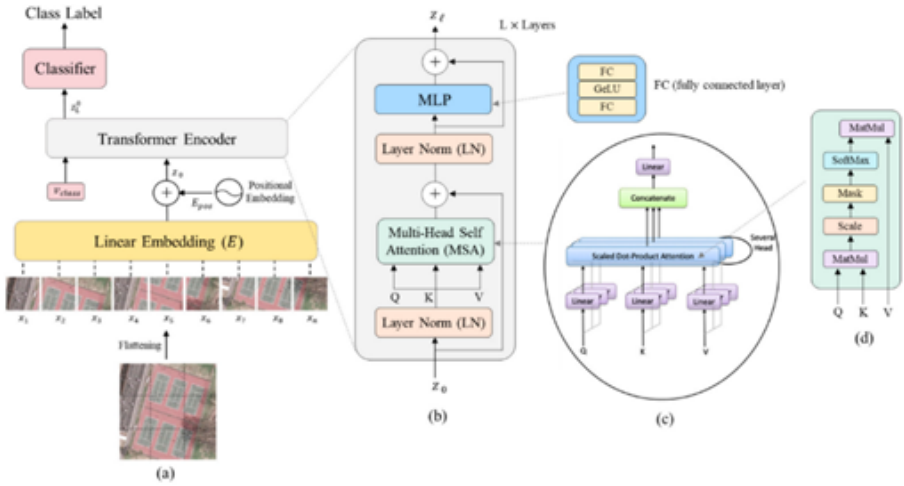


Figure 3: Vision transformer architecture

that CNN is better suited for real-world deepfake detection applications, particularly when dataset availability is limited. While DeiT-Tiny offers strong generalization, it requires extensive data and longer training times to match CNN's performance, emphasizing the need for model selection based on dataset size and computational constraints.

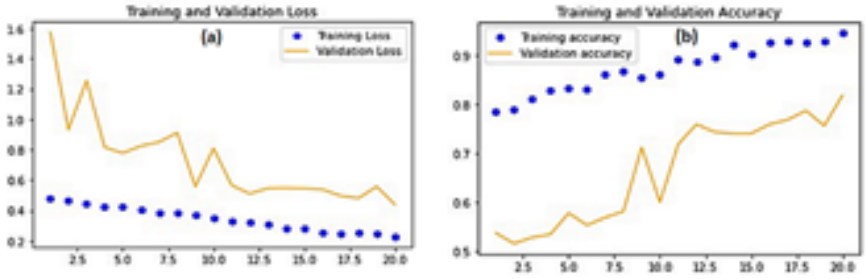


Figure 4: CNN Model Performance on Real and Fake Face Detection Dataset : (a) Loss function, (b) Accuracy

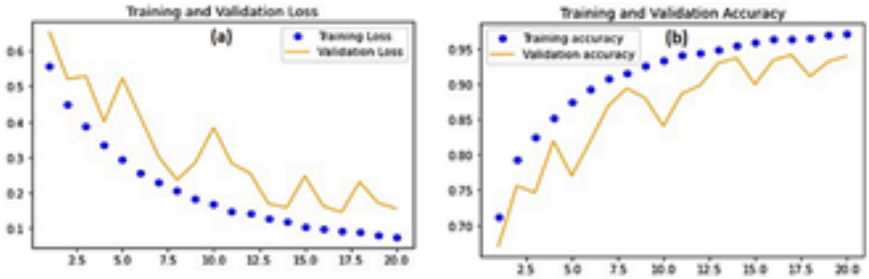


Figure 5: CNN Model Performance on 140K Real and Fake Faces Dataset : (a) Loss function, (b) Accuracy

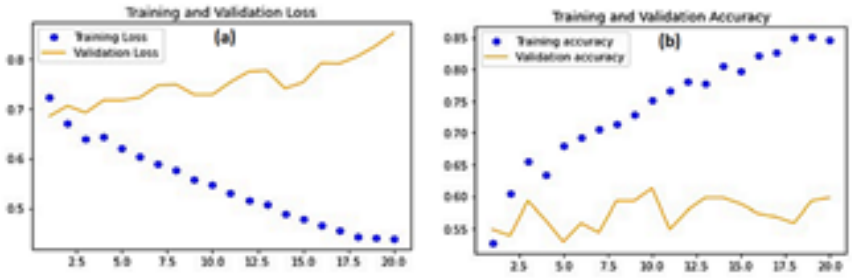


Figure 6: DeiT-Tiny Model Performance on Real and Fake Face Detection Dataset : (a) Loss function, (b) Accuracy

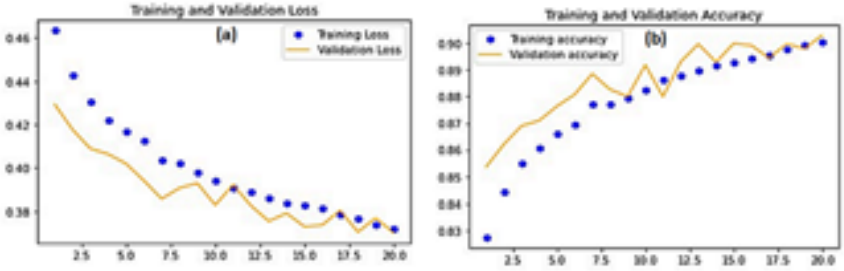


Figure 7: DeiT-Tiny Model Performance on 140K Real and Fake Faces Dataset : (a) Loss function, (b) Accuracy

5 Conclusion

Deepfake technology presents significant challenges in digital security and misinformation prevention, requiring robust detection mechanisms. This study explored CNN and DeiT-Tiny models for deepfake detection, demonstrating that CNN achieved higher accuracy and stability, especially on smaller datasets, while DeiT-Tiny required larger datasets for optimal performance. Despite advancements, deepfake detection remains complex due to adversarial attacks, dataset biases, and computational constraints. Future research should focus on real-time detection, improving model robustness, and integrating multi-modal approaches such as audio and behavioral analysis. This study contributes to enhancing digital media security, emphasizing the need for continuous advancements in AI-driven detection frameworks to combat deepfake threats effectively.

References

[1] F. Abbas and A. Taeihagh. Unmasking deepfakes: A systematic review of deepfake detection and generation techniques using artificial intelligence. *Expert Systems With Applications*, 124260, 2024.

[2] Z. Akhtar. Deepfakes generation and detection: a short survey. *Journal of Imaging*, 9(1):18, 2023.

[3] A. Heidari et al. Deepfake detection using deep learning methods: A systematic and comprehensive review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 14(2):e1520, 2024.

[4] A. Kaushal et al. A review on deepfake generation and detection: bibliometric analysis. *Multimedia Tools and Applications*, pages 1–41, 2024.

[5] B. Dolhansky et al. The deepfake detection challenge (dfdc) dataset. *arXiv preprint*, 2020.

[6] C. Li et al. A continual deepfake detection benchmark: Dataset, methods, and essentials. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1339–1349, 2023.

[7] D. Pan et al. Deepfake detection through deep learning. In *2020 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*, pages 134–143, 2020.

[8] F. Lugstein et al. Prnu-based deepfake detection. In *Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security*, pages 7–12, 2021.

[9] H. Zhao et al. Multi-attentional deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2185–2194, 2021.

[10] J. Wang et al. M2tr: Multi-modal multi-scale transformers for deepfake detection. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*, pages 615–623, 2022.

[11] K. Aashish et al. Latent flow diffusion for deepfake video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3781–3790, 2024.

[12] K. Narayan et al. Deephy: On deepfake phylogeny. In *2022 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2022.

[13] K. Narayan et al. Df-platter: Multi-face heterogeneous deepfake dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9739–9748, 2023.

[14] M. Masood et al. Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward. *Applied Intelligence*, 53(4):3974–4026, 2023.

[15] M. Rehaan et al. Face manipulated deepfake generation and recognition approaches: A survey. *Smart Science*, 12(1):53–73, 2024.

[16] M. S. Rana et al. Deepfake detection: A systematic literature review. *IEEE Access*, 10:25494–25513, 2022.

[17] O. De Lima et al. Deepfake detection using spatiotemporal convolutional networks. *arXiv preprint*, 2020.

[18] S. R. Ahmed et al. Analysis survey on deepfake detection and recognition with convolutional neural networks. In *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–7, 2022.

[19] Y. Bazi et al. Vision transformers for remote sensing image classification. *Remote Sensing*, 13(3):516, 2021.

[20] Z. Yan et al. Df40: Toward next-generation deepfake detection. *arXiv preprint*, 2024.

[21] T. Zhang. Deepfake generation and detection, a survey. *Multimedia Tools and Applications*, 81(5):6259–6276, 2022.

[22] Y. Zhou and S. N. Lim. Joint audio-visual deepfake detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Review on deep learning optimization using knowledge and dataset distillation in medical imaging diagnostics

Nor-Elhouda Laribi[1], Djamel Gaceb[2], Abdellah Rezoug[3], and Fayçal Touazi[4]

[1]***LIMOSE Laboratory University M'Hamed Bougara Boumerdes, Algeria,***
n.laribi@univ-boumerdes.dz
[2]***LIMOSE Laboratory University M'Hamed Bougara Boumerdes, Algeria,***
d.gaceb@univ-boumerdes.dz
[3]***LIMOSE Laboratory University M'Hamed Bougara Boumerdes, Algeria,***
a.rezoug@univ-boumerdes.dz
[4]***LIMOSE Laboratory University M'Hamed Bougara Boumerdes, Algeria,***
f.touazi@univ-boumerdes.dz

Abstract

The integration of deep learning-based artificial intelligence solutions in hospital environments introduces significant challenges, including data privacy restrictions, limited computational resources, and constraints related to the quality and simplicity of the models used. In this review, we highlight the recent advancements in knowledge distillation and dataset distillation as emerging solutions to these challenges in the field of medical imaging. These techniques offer practical benefits in clinical settings by enabling faster training, reduced model size, improved inference speed, and enhanced accuracy, while supporting privacy-preserving learning across decentralized systems and edge devices. Knowledge distillation transfers knowledge from a complex to a simple model, enabling efficient deployment without high loss in diagnostic performance. Dataset distillation, by contrast, focuses on synthesizing datasets that match the pretrained model on real data, reducing data storage requirements. Together, these methods improve learning efficiency, model accuracy, and resource optimization in hospital workflows. However, their integration into medical environments also presents limitations. Challenges such as pipeline complexity, scalability issues, and performance inconsistency across architectures or high-resolution tasks still persist. Overall, this review provides a comprehensive overview of potential and limitations of these two types of distillations in healthcare, offering insights into how these methods can support more scalable, accurate, and privacy-aware AI solutions for medical imaging.

Keywords: Healthcare, medical imaging, deep learning, knowledge distillation, dataset distillation, data privacy.

1 Introduction

Artificial intelligence (AI), and deep learning in particular, has become increasingly crucial in medical imaging, offering significant improvements in diagnostic accuracy, efficiency, and decision support systems. From radiology to pathology, deep learning models have demonstrated capabilities that rival or exceed human experts in specific tasks. However, deploying these powerful models in real-world hospital settings presents significant challenges. The clinical environment presents challenges, including stringent data privacy, limited computational resources in edge settings, and the practical need for real-time or near-real-time inference. These constraints pose significant challenges to the adoption of conventional, large-scale deep learning models, which are often data-intensive, resource-requirement. To address these limitations, two emerging strategies have gained traction in the research community: Knowledge Distillation (KD) and Dataset Distillation (DD). These techniques aim to retain the performance benefits of deep learning while reducing the computational and data demands that often hinder clinical deployment. Knowledge distillation works by transferring knowledge from a large, complex model (the "teacher") to a smaller, more efficient one (the "student"), preserving accuracy while improving speed and reducing resource usage. Dataset distillation, on the other hand, generates compact synthetic datasets that can replicate the behavior of real data, reducing storage needs and enabling faster training cycles and deployment settings. In this review, we provide a comprehensive overview of both KD and DD techniques in the context of medical imaging, with a particular focus on their application to brain MRI-based disease

diagnosis. We explore the methodological foundations of these approaches, analyze recent advancements, and examine their limitations in clinical settings. Through comparative studies, we highlight how various KD strategies, such as soft label supervision, intermediate feature transfer, dual-stream learning, and attention-based mechanisms, enable the compression of large teacher models into lightweight student models without significant loss in performance. In parallel, DD methods have demonstrated the ability to achieve competitive results across a range of tasks, including COVID-19 detection from chest X-rays, Alzheimer's classification from MRI scans, and skin lesion analysis, all while significantly reducing dataset size and training requirements. By analyzing the performance, benefits, and trade-offs of both KD and DD, this review offers insights into how these emerging techniques can support the development of more scalable, efficient, and privacy-aware AI systems in healthcare.

2 Knowledge distillation

Pour améliorer le diagnostic des anomalies sur les IRM cérébrales, plusieurs études exploitent la distillation de connaissances (Knowledge Distillation, KD) en transférant les connaissances d'un modèle enseignant complexe vers un modèle étudiant plus léger, afin de maintenir une haute précision diagnostique tout en réduisant la consommation de mémoire et le temps d'inférence. Par exemple, l'approche proposée dans [1], utilisant un ensemble de 357 images IRM, a atteint une précision impressionnante de 98,10 %. FM-LiteLearn, présenté dans [23], intègre les images afin d'améliorer la représentation des caractéristiques tumorales ; une stratégie de distillation multi-enseignants (MT-KD) y est appliquée pour optimiser les performances. Évalué sur le jeu de données BT_NAGMN5, le modèle proposé T-ResNet18 a obtenu une amélioration de 9,4 % de la précision de classification. Une autre étude, présentée dans [16], utilise l'auto-distillation (Class Activation Self-Distillation, stratégie CASD) pour améliorer la classification multi-modale du Gliome en affinant l'extraction des caractéristiques au sein d'un réseau à flux unique. Figure 1 shows Knowledge Distillation Framework: Soft Label Supervision from Teacher to Student.

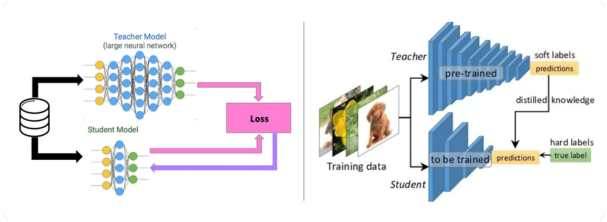


Figure 1: Knowledge Distillation Framework: Soft Label Supervision from Teacher to Student

Addressing data insufficiency in 3D brain imaging, recent studies [17, 26, 8] have demonstrated the effectiveness of knowledge distillation (KD) in enhancing model performance with limited data. For instance, in [17], KD improves performance by transferring knowledge from a powerful teacher model to a lightweight student model that combines a convolutional neural network (CNN) for feature extraction with a long short-term memory (LSTM) network to capture inter-slice correlations. This approach achieved an accuracy of 85.96%, representing a 3.83% improvement in Alzheimer's disease detection using 3D MRI scans.

To improve AI transparency in medical image analysis, another study [8] introduces Knowledge Distillation and Feature Map Visualization (KD-FMV), where a DenseNet121 teacher model is trained and transfers its knowledge to a lightweight student model. The method balances hard and soft losses for brain tumor classification: the teacher model achieved 98.77% accuracy, while the best student model reached 97.48% with a lower loss of 0.0944. In Alzheimer's classification, the teacher model attained 99.38% accuracy, with the best student model achieving 99.46% and a lower loss of 0.0194.

On the other hand, the Confidence Regularized Knowledge Distillation (CReg-KD) framework [26] achieved the highest accuracy across multiple architectures: ResNet-18 (∼94%), ResNet-50 (∼93%), DenseNet-121 (∼93%), and InceptionV3 (∼94%), consistently outperforming other distillation methods as the sample size decreased.

To address privacy concerns in brain tumor MRI analysis, FedBrain-Distill [7] and FedSPD [24] adopt a federated learning approach combined with knowledge distillation (KD). Results from the Figshare brain tumor dataset (see Figure 2) show that, with two teacher models, FedBrain-Distill achieves over 93% accuracy within just 10 communication rounds, whereas traditional federated learning (FL) methods

require 100 rounds to reach similar performance. When using five teachers, the model reaches nearly 94% accuracy, matching state-of-the-art results.

FedSPD, on the other hand, employs similarity-preserving knowledge distillation to align feature representations across clients. It outperforms traditional FL methods by 78.41% and personalized FL (PFL) methods by 10.55% in non-IID settings. Moreover, it enhances efficiency by reducing training time by 67.25% and model size by 49.34%.

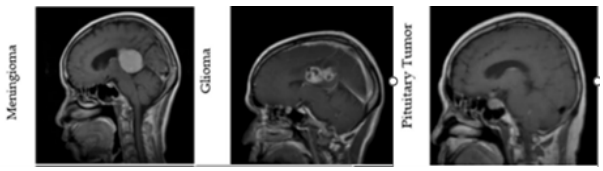


Figure 2: Brain Tumor MRI images from FiGSHARE dataset

Other studies, such as [20], utilize multiple teacher models and a lightweight student model incorporating feature aggregation, attention mechanisms, and a custom distillation loss function to improve learning efficiency while maintaining high accuracy. In [20], teacher models are trained on different datasets, while the student learns from both labeled and unlabeled data. Experimental results on the ACDC dataset and various source datasets demonstrate the method's effectiveness. For instance, on ACDC, the model achieved 89.40% accuracy with only 32 labeled samples, which improved to 95.85% with 64 labeled samples. Likewise, the F1-score increased from 89.42% to 95.74% with more labeled data [19].

Recently, vision transformers (ViTs) have been increasingly applied in conjunction with distillation techniques to transfer both intermediate features and soft labels from ViTs to smaller student models, mitigating the data inefficiency and computational complexity of ViTs in brain tumor MRI classification. LCDEiT (Linear-Complexity Data-Efficient Image Transformer) [6] introduces a custom gated-pooled CNN teacher and an external attention mechanism to improve training efficiency and reduce dependency on large datasets. This strategy achieved high classification performance: 98.11% accuracy and 97.86% F1-score on the Figshare dataset, and 93.69% accuracy and 93.68% F1-score on the BraTS-21 dataset.

Hybrid models [5, 3] further enhance feature representation while significantly reducing model complexity in the distillation process. The Data-Efficient Knowledge Distillation (HDKD) approach [5] employs a CNN-based teacher model that distills both logit- and feature-level knowledge into a hybrid student architecture combining CNN and ViT components with a lightweight convolutional block (MBCSA). This method achieved 92.9% accuracy using only 200 images.

Finally, Quantum ViTs (QViTs) [3] leverage KD to pretrain quantum vision transformer models from high-quality teachers, thereby reinforcing the strengths of QViTs. On the OASIS dataset for Alzheimer's disease classification, QViT_28 (AUC: 0.812, ACC: 0.693) closely rivals ViT_28 (AUC: 0.822, ACC: 0.701) while maintaining quantum efficiency. Furthermore, QViT_224 (AUC: 0.785, ACC: 0.678) outperforms ViT_224 (AUC: 0.603, ACC: 0.400).

On Alzheimer's disease classification tasks, the study [22] employed a Res-Transformer as the teacher model and a ResU-Net as the student model to enhance training stability and optimize skip connections for improved image reconstruction. Through knowledge distillation, this approach achieved an accuracy of 96.9% and a gain of 7.2% in performance.

In another study [4], a Residual Temporal Attention Block (RTAB) was introduced to distill temporal dependencies in a dual-stream vision transformer (DS-ViT). The method transfers knowledge from a segmentation model to a classification model by computing residuals between MRI scans over time, guiding the model to focus on subtle cues related to disease progression. This approach achieved 89.9% accuracy and 91.7% recall on the MIRIAD Alzheimer's dataset.

Additionally, the study in [10] explored the feasibility of using vision transformers (ViTs) under low-data constraints for Alzheimer's diagnosis. The proposed method reached 79.7% accuracy on the ADNI1 dataset and 82.0% on the ADNI2 dataset, significantly outperforming training without distillation, which achieved only 67.7% and 69.8% accuracy respectively.

3 Dataset distillation

Dataset distillation (DD) is a technique that compresses the knowledge of a large dataset into a small set of synthetic images, enabling models to learn effectively while significantly reducing data size and

Reference	Teacher Model (T)	Student Model (S)	Distillation Strategy	Performance
[8] **KD-FMV**	DenseNet121 (27.37 MB)	Custom CNN NB params: 37M	Feature matching + soft targets	98.77% (T), 97.48% (S)
[7] **FedBrain-Distill**	2 or 5 decentralized VGG16 teachers NB params: ∼138M × (2/5)	ConvNet 94,986 params	Federated soft distillation	**5 teachers**: 94.38% (IID)(S) 93.34% (non-IID) (S) **2 teachers:** 93.60% (IID)(S) 92.36% (non-IID) (S)
[6]**LCDEiT**	Gated pooled CNN (Figshare - 3 classes) NB params: 90795 BraTS-21 (4 classes): 90828	Transformer D1: ∼338,502 D2: ∼33,632	DEiT (Student) + Gated-Pooled CNN (Teacher) + External Attention	**Figshare** Acc 98.11 (T), Acc :94.33 (S) **BraTS** Acc 93.69 (T), Acc :87.96 (S) **LCDEiT** Figshare: Acc:98.11 BraTS: Acc :93.69
[23] **MT-KD**	Multiple large models NB params: N.A	T-ResNet18 NB params: 11.7M	Multi-teachers (ensemble distillation)	+9.4% acc improvement
[26] **CReg-KD**	ResNet-18 (NB: ∼11.7M) ResNet-50 (NB: ∼25.6M) DenseNet-121 (NB: ∼7.98M) InceptionV3 (NB: ∼23.9M)	ResNet-18, ResNet-50, DenseNet-121, InceptionV3	Self-distillation (same model used)	Baseline: Acc 87.05 → KD: 92.35±2.10 Baseline: Acc 88.45 → KD: 92.02±2.48 Baseline: Acc 86.01 → KD: 92.36±2.27 Baseline: Acc 90.21 → KD: 94.05±1.20

Table 1: Performance Comparison of Knowledge Distillation Methods on Brain MRI Images

preserving privacy (see Figure 4). The survey presented in [11] categorizes DD methods into two main frameworks: the *Meta-Learning Framework*, which optimizes synthetic data using methods such as Backpropagation Through Time (e.g., *DD*, *LD*, *GTN*) and Kernel Ridge Regression (e.g., *KIP*, *FRePo*); and the *Data Matching Framework*, which aligns properties between real and synthetic data through Gradient Match (e.g., *DC*, *IDC*), Trajectory Match (e.g., *MTT*, *TESLA*), and Distribution Match (e.g., *DM*, *KFS*, *IDM*). The survey concludes that Data Matching methods, particularly recent factorized approaches such as *IDC*, *RTP*, and *KFS*, generally outperform Meta-Learning methods on complex datasets such as *CIFAR-100* and *Tiny ImageNet*. Alternatively, a more recent survey [27] proposes a different taxonomy based on four dimensions: optimization objective, network update fashion, synthetic data parameterization, and label learning strategy. It classifies DD methods into Performance Matching (e.g., *DD*, *FRePo*), Distribution Matching (e.g., *DM*, *IDC*), and Parameter Matching (e.g., *MTT*, *HaBa*). This taxonomy highlights key differences in update strategies, label types, and parameterization styles, showing that modern methods such as *FRePo* and *DSA* offer superior scalability and accuracy across both simple and complex datasets. Taken together, both surveys suggest that modern, factorized, and optimization-efficient DD methods—particularly those based on Data Matching—are more effective and scalable, marking a clear evolution in the field.

In dataset distillation (DD) for natural images, the structure and patterns of the original data are often preserved. Typically, the distilled dataset is initialized using real images or slightly modified versions, allowing the synthetic data to retain key visual features from the original dataset. This helps maintain important textures, shapes, and object structures, enabling models to learn from a smaller subset without significant loss of information. Recent advances in DD have introduced a variety of innovative strategies to improve efficiency, robustness, privacy, and scalability. The Importance-Aware Adaptive Dataset Distillation (IADD) [15] enhances performance by assigning importance-aware weights to network parameters during training, leading to state-of-the-art results on CIFAR-10, CIFAR-100, and

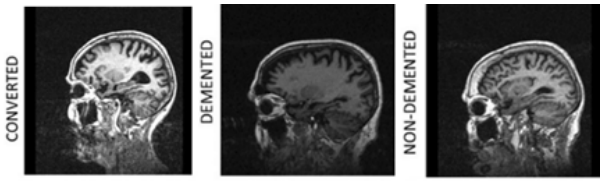


Figure 3: Brain MRI Samples of Alzheimer's Lesions from the OASIS-2 Dataset

Ref	Teacher Model (T)	Student Model (S)	Distillation Strategy	Dataset	Performance
[8]	DenseNet121 params: 27.37 MB	Custom CNN params: 37M	Logit matching + spatial feature map analysis	3D MRI	ACC: 99.46% (S) ACC: 99.38% (T)
[3]	TinyViT params: ∼5M	Quantum ViT (QVLT)	Soft label KD with quantum-classical hybrid learning	OASIS	ACC:0.965 AUC: 0.983 (T) ACC: 0.656 AUC: 0.763(S)
[22]	ResTransformer (U-Net based) NB params: ∼40M–60M	ResU-Net params: ∼7M–15M	Intermediate features + soft labels	Private	ACC: N.A (T) ACC: 96.9% (S) +7.2% acc
[4]	3D_Unet-Model (EastSurfer) NB params: ∼24M	DS-VLT params: ∼10–14M	Dual-stream distillation (segmentation to classification)	MIRIAD	ACC: N.A (T) ADAPT_ACC: 0.903 (baseline) DS-ViT_(S) ACC: 0.941
[10]	3D ResNet-152 params: ∼256M	Lightweight Transformer params: N.A	Distillation token with self-attention	ADNI1 ADNI2	ADNI1: ACC : 84.63% (T) ACC : 79.7%(S) ADNI2: ACC 82.51% (T) ACC: 82.0%(S)

Table 2: Performance Comparison of KD Methods on Alzheimer MRI Images

Tiny ImageNet. In the domain of robustness, TrustDD [18] introduces Pseudo-Outlier Exposure (POE) to distill datasets that are more resistant to out-of-distribution (OOD) inputs, achieving top AUROC and AUPR-OUT scores on CIFAR-10. The method proposed in [29] improves training efficiency through early-stage model snapshots and parameter perturbation, enabling up to 20× speedups without compromising accuracy. Privacy concerns are addressed in SFDD [2], which applies a local differential privacy mechanism (LDPO-RLD) in a decentralized setting, protecting gradient updates while improving model performance, with an 8.94% accuracy gain on the GTSRB dataset. Study [30] proposes a diffusion-based patch selection strategy for synthetic data generation, introducing a novel DD method that uses a frozen diffusion model as a teacher to select informative image patches from real data, rather than generating synthetic images. By leveraging the model's learned data distribution and text-guided semantics, it ranks and clusters patches to build a compact yet effective distilled dataset. A student model trained on this curated set learns efficiently, preserving the original data's feature distribution and improving semantic alignment compared to prior approaches. This strategy achieved 70.0% top-1 accuracy on ImageNet-1K. Finally, Distributional Dataset Distillation (DDD) is introduced in [21], a novel approach addressing inefficiencies in prototype-based DD methods, particularly the hidden storage costs of explicit label encoding. Rather than distilling into individual samples, DDD represents each class using compact per-class statistical distributions, coupled with a decoder to reconstruct representative data. This formulation allows significantly more memory-efficient distillation. To enhance scalability, the authors propose a federated distillation strategy by splitting the dataset into subsets, distilling them in parallel

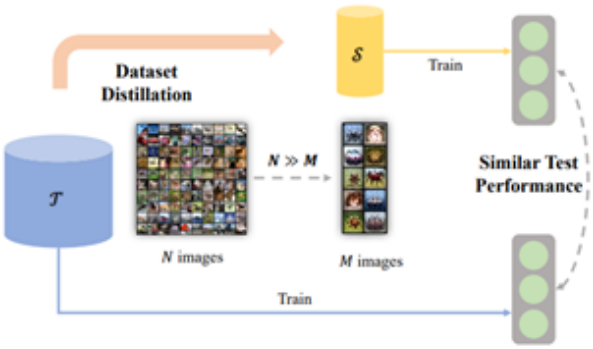


Figure 4: Dataset Distillation Framework: performance matching

via specialized sub-task models, and merging the results. Extensive experiments demonstrate that DDD achieves state-of-the-art performance, including a +6.9% accuracy improvement on ImageNet-1K under tight storage constraints (equivalent to just two images per class), highlighting both its efficiency and scalability. Collectively, these contributions mark significant progress, as illustrated in Table 3, making dataset distillation more accurate, scalable, privacy-preserving, and efficient across diverse domains and challenges.

Ref	Dataset	Classes	Size	IPC	Model(s)	Distilled Performance	Original Performance
[15]	CIFAR-10	10	60000	50	ConvNetD3	ACC = 72.6%	ACC = 84.8 ± 0.1%
[15]	CIFAR-100	100	60000	50	ConvNetD3	ACC = 49.0%	ACC = 56.2 ± 0.3%
[18]	CIFAR-10	10	60000	10	ConvNet	AUROC= 78.64%	AUROC= 65.79%
		100	60000	10	ConvNet	AUROC: 82.04%	AUROC= 49.94%
		10	60000	50	ConvNet	ACC = 60.22%	ACC = 60.55%
					AlexNet	ACC = 58.36%	ACC = 56.23%
					VGG	ACC = 56.29%	ACC = 55.02%
[18]	CIFAR-100	100	60000	10	ResNet	ACC = 51.25%	ACC = 50.00%
[29]	ImageNet-10	10	~13,000	10	ResNetAP-10	ACC = 74.6%	SpeedUp ×4.57, Acc gain ×1.01
[29]	ImageNet-100	100	~133,000	10	ResNetAP-10	ACC = 48.4%	SpeedUp ×4.76, Acc gain ×1.04
[30]	ImageNet-1K	1000	~1.28 M	50/100	VLT-B to ResNet-18	IPC 50: ACC=65.4 ± 0.7% IPC 100:ACC=70.0 ± 0.3%	NA
[21]	ImageNet-1K	1000	~1.28 M	10	ConvNet	ACC=30.5%	+6.9% gain over baseline
[2]	GTSRB	43	39,27	1	ConvNet	ACC=32.13%	+0.19% over centralized DD (31.94%)
		43	39,27	10	ConvNet	ACC=65.38%	ACC=66.55%

Table 3: Performance Comparison of Dataset Distillation Methods on Natural Images (IPC: Image Number Per Class)

However, in medical imaging, privacy concerns require a different approach. Instead of initializing distilled data from real medical images, the process begins with completely random patterns, often generated using Gaussian noise or other noise-based strategies. These synthetic samples are then optimized through dataset distillation techniques to replicate the training behavior of real medical images while ensuring that no identifiable structures or sensitive patient information are retained. This strategy enables privacy-preserving model training while maintaining the effectiveness of the distilled datasets for downstream tasks.

In the medical imaging domain, several recent DD methods have been proposed to enable secure, efficient, and privacy-preserving model training. For instance, UniCompress [25] applies dataset distillation to the domain of medical image compression, using feature alignment and cross-attention in a knowledge distillation (KD) pipeline to transfer information from a teacher to a lightweight student model, achieving 4–5× faster compression with top-tier PSNR and SSIM scores. The Anonymous Gastric Image Distillation method [12] uses a gradient-based approach that achieves a harmonic mean (HM) of 0.877, outperforming ResNet-18 trained on up to 3,000 real images. Similarly, soft-label dataset distillation (SLDD) [13] also achieves an HM of 0.877, surpassing both traditional hard-label distillation methods and large-scale ResNet-18 models. The study concludes that the minimum number of compressed images required is correlated with the number of model parameters.

For cross-hospital sharing of COVID-19 chest X-ray data, another study [14] achieves 82.7% accuracy using only 20 images per class, closely approaching the 88.9% accuracy obtained when training on the full dataset. MedSynth [9] introduces a condensation framework that uses an attention-based generator fine-tuned with a Vision Transformer (ViT) to align synthetic and real data logits, achieving up to 97.11% accuracy and 96.27% AUC on Alzheimer's and ISIC 2019 datasets.

Further advancing medical dataset distillation, a progressive trajectory matching method [28] applies multi-stage alignment of model parameters trained on synthetic versus real data, combined with dynamic overlap mitigation and scheduled retraining. This approach achieves state-of-the-art accuracy on high-resolution datasets, including 66.18% on COVID19-CXR, 65.37% on BREAST-ULS, and 51.19% on SKIN-HAM, using only two images per class, outperforming all previously reported distillation techniques.

Ref	Dataset	Size	D.Size / IPC	Model	Distilled Performance	Baseline Performance
[25]	CT Scans	201 3D patients	∼512× compression ratio	ResNet-50	ACC = 0.9811 (Liver) ACC = 0.9812 (Colon) ACC = 0.9758 (Spleen)	Teacher models (no DD): slower by 4–5×
[12]	Gastric X-ray	815 patients	1	ResNet-18	Sensitivity = 0.886 Specificity = 0.869	+5% HM
[13]	Gastric X-ray	815 patients	1	GoogLeNet ResNet-18 AlexNet VGG16	HM = 0.882 HM = 0.869 HM = 0.836 HM = 0.916	Cross-model comparison
[14]	COVID-19 Chest X-ray	21 165	20	ConvNet	Accuracy = 82.7%	88.9% (full dataset)
[9]	Alzheimer's, ISIC 2019	5 121 samples	50	DCGAN	ACC = 97.11%	∼20× smaller dataset
[28]	COVID19-CXR	21 165	2/10	ConvNet	IPC 2: ACC = 66.18% ± 0.02 IPC 10: ACC = 69.65% ± 0.01	90.22% ± 0.01
[28]	BREAST-ULS	780	2/10	ConvNet	IPC 2: ACC = 65.37% ± 0.02	74.00% ± 0.07
	SKIN-HAM	10 015	2/10	ConvNet	IPC 10: ACC =68.90% ± 0.01 IPC 2: ACC = 51.19% ± 0.02	70.17% ± 0.02

Table 4: Performance Comparison of DD Methods on Medical Images (IPC: Image Number Per Class)

Future research in dataset distillation should focus on advanced techniques like GANs, VAEs, and diffusion models to enhance data fidelity and privacy-preserving data sharing which is crucial in sensitive domains like medical imaging. Advancements in methods like MedSynth and DDD should emphasize scalability, generalization, and robustness.

4 Discussion

Knowledge distillation (KD) in brain MRI imaging has emerged as a powerful technique not only for compressing large models into smaller, more efficient ones but also for addressing challenges such as privacy, diversity, and adaptability to specific target tasks. When integrated with approaches like federated learning (e.g., FedBrain-Distill [5]), it enables model training without directly sharing sensitive data, thereby enhancing privacy. FedBrain-Distill focuses on privacy and communication efficiency in federated settings, where multiple decentralized teacher models produce soft labels (via temperature-scaled softmax) for a central lightweight student, resulting in strong generalization performance without exchanging data or model weights (94.38% IID, 93.34% non-IID). In contrast, deep models such as DenseNet121, used in [6], serve as rich teachers to guide a custom CNN student via both soft targets and feature representation matching, achieving high performance (98.77% teacher, 97.48% student) despite the student having more parameters—highlighting computational efficiency and inference speed as more critical than model size.

Furthermore, recent studies have expanded the scope of KD by exploring the use of more complex or "heavy" teacher models to optimize performance, efficiency, and model architectures, adapting the methods for specific applications and deployment environments. For instance, the study in [10] replaces heavy teacher models with a lightweight gated CNN and uses attention-guided and self-supervised distillation to train a compact transformer-based LCDEiT student (approximately 338K parameters), achieving competitive accuracy (up to 98.11%) with far lower complexity. Collectively, these studies demonstrate how KD can optimize performance, efficiency, privacy, and architectural design depending on the specific use case.

In the context of Alzheimer's disease datasets, several studies have shown the power of KD in reducing model complexity without compromising—and in some cases, even enhancing—student model

performance. For example, the work in [6] uses KD through logit matching and feature map analysis, allowing a custom 5-layer CNN to approach the accuracy of a DenseNet121 teacher (99.38% vs. 99.46%) while requiring nearly ten times fewer operations, demonstrating how lightweight models can achieve comparable results. In [15], knowledge transfer from a Res-Transformer to a lightweight ResU-Net student results in a 7.2% accuracy increase and an estimated 5–10× reduction in computational complexity, emphasizing the benefits of distilling both soft labels and intermediate features.

Taking a novel direction, study [27] explores the integration of KD with quantum neural networks, where a TinyViT teacher distills knowledge into a QViT student model. This approach achieves up to 80× compression while maintaining strong classification performance (AUC up to 0.812), highlighting KD's potential in hybrid quantum-classical learning systems. Similarly, KD can go beyond model-to-model transfer for the same task. In [18], a dual-stream KD strategy is employed to bridge two distinct tasks—segmentation and classification—by distilling structural knowledge from a segmentation model into a DS-ViT classifier. This cross-task distillation leads to notable improvements in classification accuracy (0.899) and recall (0.917), while also reducing model size by approximately 5×, showcasing KD's ability to enable transfer learning across functionally different but related domains. Another approach, proposed in [29], introduces a distillation token mechanism that transfers knowledge from a large 3D ResNet-152 to a significantly smaller transformer-based student. Despite achieving 9.7× compression, the performance gain is modest (0.1%) and computational costs remain high due to the heavy teacher model. These studies illustrate the broad utility of KD—not only as a model compression method, but also as a bridge across architectures, learning paradigms, and deployment constraints—especially within the sensitive and resource-limited field of medical imaging.

Inspired by the principles of KD, a related and increasingly powerful technique known as dataset distillation (DD) has emerged. Unlike KD, which transfers knowledge from one model to another, DD transfers knowledge from real data—or a model trained on it—into a much smaller, synthetic dataset. In this paradigm, the teacher is not a model but the original data distribution itself, and the student (often with the same architecture as the teacher) is trained exclusively on the distilled dataset. This method aims to match the performance of models trained on full datasets using only a synthesized subset derived from the original data. Recent studies have further developed this concept by evaluating multiple model architectures during the distillation process, thereby improving the generalizability of the distilled data across tasks and learners.

Among its key advantages, DD significantly accelerates training times and reduces storage requirements, making it especially effective for simpler datasets such as CIFAR-10 and GTSRB. For instance, studies show that with just 10 to 50 images per class (IPC), models trained on distilled CIFAR-10 can achieve up to 72.6% accuracy compared to 84.8% on the full dataset, while GTSRB achieves 65.38% using IPC 10 versus 66.55% with full data. This makes DD ideal for edge deployment or federated learning setups with strict data transmission and storage limits. Additionally, DD can deliver notable speedups (e.g., 4.5× on ImageNet subsets) and even occasional performance improvements over traditional distributed training.

However, these advantages come with significant limitations. As dataset complexity increases (e.g., CIFAR-100, ImageNet-100, ImageNet-1K), the performance gap between models trained on distilled versus full datasets becomes substantial. For example, on ImageNet-1K with 10 IPC, accuracy drops to 30.5%, although increasing to 100 IPC improves it to 70.0%. These datasets contain high inter-class variability and complex patterns that are difficult to capture with limited synthetic data. Moreover, the success of DD is highly architecture-dependent: while simple networks like AlexNet and VGG retain relatively high AUC and accuracy on distilled data, more complex models like ResNet and ViT are more sensitive to the quality and diversity of synthetic samples. To mitigate this, some methods introduce teacher-student mechanisms during DD, improving results at the cost of greater pipeline complexity.

DD has shown great promise in medical imaging, achieving significant data reduction while preserving performance, particularly with ConvNet and ResNet architectures. Studies indicate that even under extremely low IPC settings, models can maintain strong performance on tasks such as CT scan segmentation and chest X-ray classification. However, a major challenge to generalizability remains: distilled data often overfit to the model architecture they were generated for, reducing reusability across other architectures. Furthermore, on more diverse or clinically detailed datasets, performance tends to decline, and important but rare features may be lost. Most research has focused on convolutional architectures, limiting exploration of modern transformer-based models. Additionally, the interpretability and clinical reliability of synthetic data remain open concerns in real-world medical applications.

5 Conclusion

Knowledge and dataset distillation are emerging as impactful techniques in medical imaging, particularly in hospital settings where data privacy, sharing restrictions, and limited computational resources are key concerns. KD enables the compression of large models into efficient, high-performing versions, making it ideal for decentralized systems, edge devices, and federated learning environments where raw data cannot be shared. DD complements this by creating compact synthetic datasets that capture the essential patterns of full datasets, enabling training without exposing sensitive medical data. These approaches support privacy-preserving and resource-efficient AI deployment but face notable challenges. KD often involves complex training pipelines with teacher-student models, while DD may struggle to retain diagnostic fidelity in high-resolution or fine-grained tasks. Both methods also show performance variability across different architectures, highlighting the need for careful tuning and optimization. Looking forward, integrating generative adversarial networks (GANs) into dataset distillation offers a promising direction. GANs can improve the realism and diversity of synthetic data, helping to close the performance gap between distilled and full datasets. With continued research, KD and DD have strong potential to support scalable, accurate, and privacy-aware AI systems in clinical environments.

References

[1] R. Anantathanavit, F. H. Raswa, T. Thaipisutikul, and J. C. Wang. Lightweight brain tumor diagnosis via knowledge distillation. In *2024 International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, pages 1–6. IEEE, August 2024.

[2] M. Arazzi, M. Cihangiroglu, S. Nicolazzo, and A. Nocera. Secure federated data distillation. *arXiv preprint*, 2025.

[3] T. Boucher and E. B. Mazomenos. Distilling knowledge into quantum vision transformers for biomedical image classification. *arXiv preprint*, 2025.

[4] K. Chen, Y. Wang, Y. Zhou, and H. Wang. Ds-vit: Dual-stream vision transformer for cross-task distillation in alzheimer's early diagnosis. *arXiv preprint*, 2024.

[5] O. S. EL-Assiouti, G. Hamed, D. Khattab, and H. M. Ebied. Hdkd: Hybrid data-efficient knowledge distillation network for medical image classification. *Engineering Applications of Artificial Intelligence*, 138:109430, 2024.

[6] G. J. Ferdous, K. A. Sathi, M. A. Hossain, M. M. Hoque, and M. A. A. Dewan. Lcdeit: A linear complexity data-efficient image transformer for mri brain tumor classification. *IEEE Access*, 11:20337–20350, 2023.

[7] R. J. Gohari, L. Aliahmadipour, and E. Valipour. Fedbrain-distill: Communication-efficient federated brain tumor classification using ensemble knowledge distillation on non-iid data. *arXiv preprint*, 2024.

[8] Y. Jiang, X. Zhao, Y. Wu, and A. Chaddad. A knowledge distillation-based approach to enhance transparency of classifier models. *arXiv preprint*, 2025.

[9] R. Kanagavelu, M. Walia, Y. Wang, H. Fu, Q. Wei, Y. Liu, and R. S. M. Goh. Medsynth: Leveraging generative model for healthcare data sharing. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 654–664, Cham, October 2024. Springer Nature Switzerland.

[10] K. Kunanbayev, V. Shen, and D. S. Kim. Training vit with limited data for alzheimer's disease classification: An empirical study. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 334–343, Cham, October 2024. Springer Nature Switzerland.

[11] S. Lei and D. Tao. A comprehensive survey of dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):17–32, 2023.

[12] G. Li, R. Togo, T. Ogawa, and M. Haseyama. Soft-label anonymous gastric x-ray image distillation. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 305–309. IEEE, October 2020.

[13] G. Li, R. Togo, T. Ogawa, and M. Haseyama. Compressed gastric image generation based on soft-label dataset distillation for medical data sharing. *Computer Methods and Programs in Biomedicine*, 227:107189, 2022.

[14] G. Li, R. Togo, T. Ogawa, and M. Haseyama. Dataset distillation for medical dataset sharing. *arXiv preprint*, 2022.

[15] G. Li, R. Togo, T. Ogawa, and M. Haseyama. Importance-aware adaptive dataset distillation. *Neural Networks*, 172:106154, 2024.

[16] J. Li, L. Zhang, K. Zhong, and G. Qian. A discrepancy-aware self-distillation method for multi-modal glioma grading. *Knowledge-Based Systems*, 295:111858, 2024.

[17] Y. Li, J. Luo, and J. Zhang. Classification of alzheimer's disease in mri images using knowledge distillation framework: an investigation. *International Journal of Computer Assisted Radiology and Surgery*, 17(7):1235–1243, 2022.

[18] S. Ma, F. Zhu, Z. Cheng, and X. Y. Zhang. Towards trustworthy dataset distillation. *Pattern Recognition*, 157:110875, 2025.

[19] N. Matcha, S. Ramanarayanan, M. Al Fahim, R. GS, K. Ram, and M. Sivaprakasam. Sft-kd-recon: Learning a student-friendly teacher for knowledge distillation in magnetic resonance image reconstruction. In *Medical Imaging with Deep Learning*, pages 1423–1440. PMLR, January 2024.

[20] S. Nabavi, K. A. Hamedani, M. E. Moghaddam, A. A. Abin, and A. F. Frangi. Multiple teachers-meticulous student: A domain adaptive meta-knowledge distillation model for medical image classification. *arXiv preprint*, 2024.

[21] T. Qin, Z. Deng, and D. Alvarez-Melis. Distributional dataset distillation with subtask decomposition. *arXiv preprint*, 2024.

[22] Y. Song, J. Wang, Y. Ge, L. Li, J. Guo, Q. Dong, and Z. Liao. Medical image classification: knowledge transfer via residual u-net and vision transformer-based teacher-student model with knowledge distillation. *Journal of Visual Communication and Image Representation*, 102:104212, 2024.

[23] S. Tan, Y. Cai, Y. Zhao, J. Hu, Y. Chen, and C. He. FM-LiteLearn: A lightweight brain tumor classification framework integrating image fusion and multi-teacher distillation strategies. In *International Conference on AI in Healthcare*, pages 89–103, Cham, August 2024. Springer Nature Switzerland.

[24] B. Wu, D. Shi, and J. Aguilar. Brain tumors classification in mris based on personalized federated distillation learning with similarity-preserving. *International Journal of Imaging Systems and Technology*, 35(2):e70046, 2025.

[25] R. Yang, Y. Chen, Z. Zhang, X. Liu, Z. Li, K. He, and Q. Dai. Unicompress: Enhancing multi-data medical image compression with knowledge distillation. *arXiv preprint*, 2024.

[26] Y. Yang, X. Guo, C. Ye, Y. Xiang, and T. Ma. Creg-kd: Model refinement via confidence regularized knowledge distillation for brain imaging. *Medical Image Analysis*, 89:102916, 2023.

[27] R. Yu, S. Liu, and X. Wang. Dataset distillation: A comprehensive review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):150–170, 2023.

[28] Z. Yu, Y. Liu, and Q. Chen. Progressive trajectory matching for medical dataset distillation. *arXiv preprint*, 2024.

[29] L. Zhang, J. Zhang, B. Lei, S. Mukherjee, X. Pan, B. Zhao, and D. Xu. Accelerating dataset distillation via model augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11950–11959, 2023.

[30] X. Zhong, S. Sun, X. Gu, Z. Xu, Y. Wang, M. Zhang, and B. Chen. Efficient dataset distillation via diffusion-driven patch selection for improved generalization. *arXiv preprint*, 2024.

Part II

Deep Learning and Data Processing Applications

RL-Guided Pruning of CNNs Using Graph Embeddings

Karima Amrouche[11], Yacine Ait Ali Yahia[21], and Ilhem Kherroubi[31]

[1]*Laboratoire de la Communication dans les Systèmes Informatiques (LCSI), École Nationale Supérieure d'Informatique, BP 68M, 16309, Alger, Algérie*

Abstract

This paper presents a novel method for compressing Convolutional Neural Networks (CNNs) to enable efficient deployment on low-capacity devices. The proposed approach combines neural network pruning with reinforcement learning (RL) and graph embedding. Each network is represented as a computational graph, and Graph Convolutional Networks (GCNs) are utilized to learn graph-level embeddings that inform pruning decisions. By applying Proximal Policy Optimization (PPO), we automate the selection of layer-wise pruning ratios, eliminating the need for manual tuning. Experiments on ResNet-34 and VGG-19, trained on the CIFAR-10 dataset, demonstrate that our method achieves up to 80% compression while maintaining or improving model accuracy through post-pruning rewinding. We evaluated both structured and unstructured pruning strategies, analyzing the trade-offs in accuracy, FLOPs, parameter count, and inference time.

Keywords: Model Compression, Deep Neural Networks, Graph Embedding, Reinforcement Learning, Neural Networks, Pruning, Convolutional Neural Networks, CNN Acceleration.

1 Introduction

The deployment of convolutional neural networks (CNN) in low-capacity devices, such as smartphones and IoT devices [1], presents significant challenges due to their high computational demands and large memory requirements. These constraints limit the use of CNNs in real-time applications, such as facial recognition or object detection, especially in environments where cloud resources are unavailable or introduce unacceptable latency.

Model compression techniques, particularly neural network pruning, have effectively overcome these challenges. By reducing the complexity of CNNs, pruning reduces computational cost and memory usage, enabling deployment on resource-constrained devices without significant performance loss. However, determining the optimal pruning strategy for each layer remains challenging, as it often requires manual tuning and is sensitive to the network structure.

We introduce a method that combines neural network pruning with reinforcement learning (RL) and graph embeddings to automate the compression process. This approach reduces the need for manual tuning while preserving accuracy and efficiency, key requirements for deploying CNNs on low-capacity devices.

2 Related Work

Model compression techniques can be categorized into several approaches, including knowledge distillation, quantization, factorization, and pruning [2]. Among these, pruning offers a direct and effective way to reduce model complexity by eliminating redundant parameters, often with minimal loss in accuracy. It is particularly attractive because it can be applied post-training, preserves the original architecture, and complements other techniques such as quantization.

The theoretical basis for pruning is strengthened by the Lottery Ticket Hypothesis (LTH) [3], which proposes that overparameterized networks contain smaller, trainable subnetworks—referred to as "winning tickets"—capable of reaching comparable accuracy to the full model when trained in isolation. As Frankle and Carbin describe:

> "A randomly-initialized, dense neural network contains a subnetwork that is initialized such that—when trained in isolation—it can match the test accuracy of the original network after training for at most the same number of iterations." [3]

This insight highlights the inherent redundancy in large neural networks and motivates pruning as a principled strategy for model compression. However, identifying such subnetworks remains non-trivial, especially in deeper architectures, where brute-force or heuristic pruning methods become computationally prohibitive. This challenge underscores the need for more scalable and intelligent pruning approaches—such as those guided by reinforcement learning and graph-based representations.

In practice, pruning techniques are generally classified into two categories: unstructured and structured. Unstructured pruning removes individual weights, often resulting in sparse models with high compression rates. However, these irregular patterns typically require specialized hardware for efficient execution. Structured pruning, in contrast, removes entire filters, channels, or blocks, producing smaller, dense models that are more compatible with conventional hardware and easier to deploy [4, 5]. Despite their practical advantages, traditional pruning methods often rely on fixed heuristics to determine layer-wise pruning ratios, which may not generalize well across architectures or datasets.

To overcome these limitations, recent work has turned to reinforcement learning (RL) to automate the pruning process. Notable examples include ABCPruner [6] and CCPruner [7], which use RL agents to learn pruning policies that balance model efficiency and accuracy. While these methods have shown promise, they often treat layers independently and fail to account for the structural dependencies across the network.

Our work addresses this gap by modeling the neural network as a computational graph, enabling a more holistic view of the architecture. By leveraging graph embeddings, we capture global structural information that informs pruning decisions, leading to more coherent and scalable model compression.

3 Building a Computational Graph from a Neural Network

As illustrated in Figure 1, our compression pipeline consists of the following stages:

1. **Initialization**: The process begins with the initialization of a deep neural network, either from scratch or using a pretrained model. At this stage, the initial weight values are preserved to enable potential rewinding after pruning, as part of the iterative compression strategy.

2. **Training**: The initialized model is trained on the target task until it achieves satisfactory performance. This yields a fully trained network that serves as the baseline for subsequent pruning.

3. **Pruning**: Reinforcement learning is employed to determine the optimal pruning rates for each layer. A Graph Convolutional Network (GCN) is used to encode the computational graph of the trained model into a latent state representation. This representation is then processed by a policy network, which generates pruning decisions. These decisions are evaluated based on a reward signal reflecting the trade-off between compression and model accuracy.

4. **Rewinding**: After pruning, the model is reverted to its initial weights saved during the initialization phase. This step, known as rewinding, facilitates the identification of subnetworks—often referred to as "lottery tickets"—that can be retrained from the original initialization to achieve strong performance.

5. **Retraining**: The selected subnetwork is retrained from its original initialization. The objective is to recover the model's performance to a level comparable to the original unpruned network, thereby achieving an efficient and accurate compressed model.

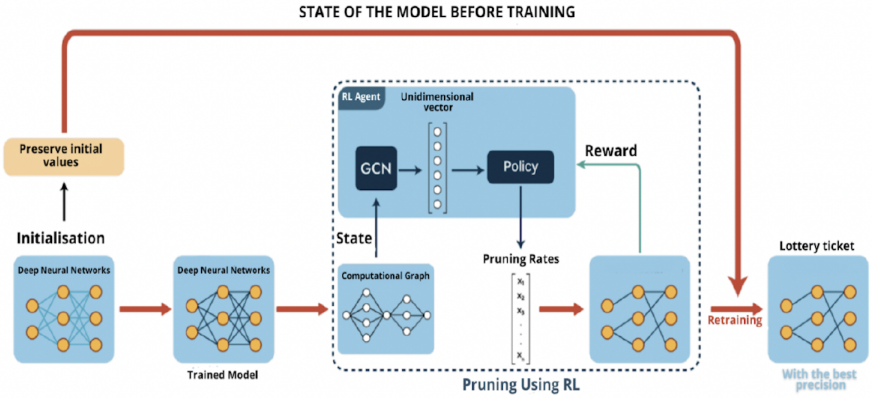


Figure 1: Pipeline illustrating CNN pruning guided by reinforcement learning, using graph embeddings to encode architectural information.

The model automates the pruning phase, removing the need for human input. A DNN's computational graph maps operations like addition and multiplication during this phase to produce outputs. Nodes perform computations, while edges guide the data flow through the network's layers [8].

Algorithm 1 outlines the procedure for constructing a subgraph for a single layer, which involves initializing input and output nodes and creating edges to represent the connections between them.

Algorithm 1: Subgraph Construction for a CNN Layer

Data: n: Number of input channels
Input: N: Number of output channels
E: List of edges (empty for the first layer)
Result: E: Updated list of edges
$Output$: Input node of the next layer

$Output \leftarrow Input + N + 1$;
for $i \leftarrow 1$ **to** N **do**
  $E \leftarrow \text{Insert}(E, (Input, Input + i))$; // Insert edge $e^{ik}_{[input]}$
  $E \leftarrow \text{Insert}(E, (Input + i, Output))$; // Insert edge $e^{ik}_{[output]}$

4 Graph Embedding

Graph embedding transforms structured data into low-dimensional vector representations, facilitating efficient learning and decision-making in downstream tasks. In this work, we represent convolutional neural networks (CNNs) as computational graphs, where nodes correspond to layers or operations and edges capture the flow of information. This graph-based formulation enables the use of Graph Convolutional Networks (GCNs) to encode the topological and functional properties of the CNN architecture.

By leveraging GCNs, we obtain a fixed-size embedding of the network that preserves both structural dependencies and feature hierarchies, which are essential for informing pruning decisions [9]. This compact representation is then passed to the reinforcement learning (RL) agent, which uses it to select optimal pruning strategies. The use of GCNs ensures that similar CNN architectures yield similar embeddings, improving the generalization of the pruning policy across different models.

As input to the GCN, we define the node features using the number of nodes ($|V|$) and their attributes (F_{in}), along with edge indices ($2, |\mathcal{E}|$), and output node features ($|V|, F_{\text{out}}$). Instead of focusing solely on individual node embeddings, we aim to compute a holistic representation of the entire graph. To achieve this, we first apply a GCN encoder that maps the graph G to a set of node embeddings $H \in R^{N \times d}$, as shown in Equation 1:

$$H = \text{GCN}_{\text{encoder}}(G) \in R^{N \times d} \tag{1}$$

The resulting node embeddings are then aggregated using a global pooling operation. Specifically, we use a `GlobalMeanPool` that computes the average over all node embeddings, producing the final graph-level embedding g, as defined in Equation 2:

$$g = \frac{1}{N} \sum_{n=1}^{N} h_i \tag{2}$$

In Equation 2, h_i denotes the embedding of the i-th node, N is the total number of nodes in the graph, and d is the dimensionality of the embedding space. This aggregated representation g captures the structural and semantic properties of the CNN architecture, and is used as input to the reinforcement learning agent.

5 Criteria for Choosing the Compression Ratio

Our goal is to optimize convolutional neural networks (CNNs) for deployment on low-capacity devices by applying structured pruning techniques. This reduces inference time, memory usage, and model size. To guide pruning, we focus on two key efficiency criteria:

- **Model Parameters**: Reducing the number of parameters directly decreases the computational and memory overhead, enhancing the efficiency of the model [10].
- **FLOPs (Floating Point Operations)**: FLOPs quantify the computational effort required per inference and are especially relevant in CNNs due to weight sharing [11]. Minimizing FLOPs has a direct impact on inference speed and energy consumption.

FLOPs are calculated per layer type as follows:

- **Convolutional Layers:**

$$\text{FLOPs} = 2 \times C_O \times C_I \times K \times O \tag{3}$$

  where C_O is the number of output channels, C_I is the number of input channels, K is the kernel size, and O is the number of output elements.

- **Fully Connected Layers:**

$$\text{FLOPs} = 2 \times I \times O \tag{4}$$

  where I and O are the number of input and output units, respectively.

6 Pruning Methods Selection

To enable the deployment of convolutional neural networks (CNNs) on low-capacity devices, it is essential to reduce their inference time, memory footprint, and overall model size. Pruning—i.e., removing less important components of the network—is a widely used approach to achieve such compression. In this work, we evaluate both unstructured and structured pruning techniques for their effectiveness in compressing CNN architectures.

- **Unstructured Pruning:** This method eliminates individual weights from convolutional kernels based on their magnitude or contribution. While it can result in highly sparse models with minimal impact on accuracy, it often fails to yield practical improvements in inference time or memory usage. This is largely due to the lack of hardware-level support for irregular sparsity, which limits the efficiency gains on general-purpose devices.
- **Structured Pruning:** In contrast, structured pruning removes entire filters, channels, or even layers. This produces a more compact and regular architecture, leading to measurable reductions in computation and memory requirements. However, structured pruning carries a higher risk of accuracy degradation if critical components are pruned without adequate guidance.

In addition to the pruning strategy itself, the choice of how to reinitialize and retrain the pruned network plays a crucial role in recovering or maintaining performance [12]. We evaluate the following post-pruning training approaches:

- **Rewinding (100%):** After pruning, the remaining weights are reset to their initial values recorded at the start of training. This approach is motivated by the Lottery Ticket Hypothesis, which suggests that certain subnetworks can achieve competitive performance when trained from their original initialization.
- **Random Initialization:** As a baseline, we reinitialize the surviving weights with new random values after pruning. This serves to evaluate whether rewinding provides a significant advantage over fresh initialization.
- **Fine-Tuning:** This method retains the final values of the remaining weights and continues training with a reduced learning rate. The goal is to refine the pruned model without substantially altering its learned representations. Fine-tuning is commonly used in practice due to its simplicity and effectiveness.

7 Implementation of Reinforcement Learning

To automate the pruning process, we employ reinforcement learning (RL) with the Proximal Policy Optimization (PPO) algorithm. The RL agent is trained to predict optimal pruning ratios for each layer of a convolutional neural network, with the objective of maximizing compression while preserving classification accuracy.

Experiments were conducted on the VGG-19 and ResNet-34 architectures using the CIFAR-10 dataset, with a global compression target of 80%. The agent receives as input a graph-level embedding of the CNN architecture, obtained via a Graph Convolutional Network (GCN) encoder. Based on this representation, the agent generates pruning decisions across layers. Following pruning, we apply several retraining strategies—including random initialization, weight rewinding, and fine-tuning—to restore or improve performance. This framework enables a systematic exploration of the trade-off between model compactness and accuracy, leading to efficient CNNs suitable for deployment on resource-constrained devices.

7.1 Proximal Policy Optimization (PPO)

Proximal Policy Optimization (PPO) is a widely used reinforcement learning algorithm designed to improve the policy—the strategy for selecting actions—while maintaining stability and efficiency. It achieves this by carefully balancing exploration (trying new actions) and exploitation (choosing the best-known actions) [13]. PPO optimizes a clipped surrogate objective that limits abrupt changes to the policy and incorporates additional terms to enhance learning. The total objective function consists of three main components:

- **Policy Surrogate Loss:** This term encourages beneficial updates to the policy based on the advantage of actions taken. It uses a ratio of the new and old policy probabilities:

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta\text{old}}(a_t|s_t)} \tag{5}$$

  To prevent large, destabilizing policy updates, PPO applies a clipping mechanism:

$$L_1(\theta) = \min\left(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right) \tag{6}$$

- **Value Function Loss:** This component minimizes the error between the predicted value of a state and its target return, computed using the mean squared error:

$$L_2(\theta) = \left(V_\theta(s_t) - V_{\text{targ},t}\right)^2 \tag{7}$$

- **Entropy Bonus:** To promote exploration and prevent premature convergence to deterministic policies, PPO adds an entropy regularization term:

$$L_3(\theta) = S[\pi_\theta](s_t) \tag{8}$$

The overall loss function used to update the policy parameters is a weighted sum of the three components:

$$L_{\text{Total}}(\theta) = \hat{E}_t \left[L_1 + c_1 L_2 + c_2 L_3 \right] \tag{9}$$

where c_1 and c_2 are coefficients that balance the contributions of the value loss and entropy bonus, and $\hat{E}_t[\cdot]$ denotes the empirical average over a finite batch of experiences.

Exploration Noise: During exploration, the agent samples actions from a Gaussian policy. The probability density function of a Gaussian distribution is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{x - \mu}{\sigma} \right)^2 \right) \tag{10}$$

Initially, a fixed standard deviation σ controls the amount of randomness in action selection. Over time, this noise is gradually reduced to encourage exploitation of learned policies as training progresses.

7.2 Memory and Experience Replay

To enhance learning efficiency and generalization, the PPO agent maintains a memory buffer that stores past interactions with the environment, including states, actions, action probabilities, and rewards. Rather than updating the policy network using only the most recent data, the agent samples mini-batches of past experiences. This experience replay strategy reduces overfitting and provides a more diverse set of training samples, enabling the agent to learn from a broader distribution of environment interactions.

7.3 Reinforcement Learning Environment

The reinforcement learning (RL) environment models the deep neural network as a computational graph in a simulated setting that reflects structural changes during pruning. The environment provides the RL agent with the graph representation based on the CNN's topology and dynamically tracks key metrics, including the number of parameters and FLOPs. The pruning process proceeds step by step until the agent satisfies the compression constraints, at which point the search is terminated. This mimics the RL paradigm, where the environment evolves with each action and ends an episode once a terminal condition—such as reaching a pruning goal—is met.

7.4 Timestamps and Agent Interaction

The RL agent interacts with the environment incrementally to prune the network towards a target compression ratio. At each time step, relevant model attributes are updated, including pruning ratios and input/output channel sizes. Upon reaching the compression goal, the pruned model is evaluated in terms of classification accuracy, and a reward is assigned. If the agent fails to meet the target FLOPs within the episode, it is penalized accordingly. To maintain meaningful compression while avoiding excessive information loss, pruning ratios for each layer are constrained within the range [0.02, 0.9].

8 Experiments and Results

Following the formulation of our approach, we conducted experiments on the VGG-19 and ResNet-34 architectures using the CIFAR-10 dataset. The implementation was carried out in Python with support from various libraries: NumPy for numerical computations, Matplotlib for visualizations, and PyTorch and PyTorch Geometric for deep learning and graph-based modeling, respectively. Torchvision was used for computer vision tasks, while Weights Biases (W&B) provided experiment tracking. Execution and collaboration were facilitated via Google Colab, Amazon EC2, and Google Drive.

The experimental pipeline focused on evaluating pruning capabilities through four stages: initial model training, reinforcement learning-based pruning (targeting 80% compression), application of post-pruning retraining methods (random initialization, rewinding, fine-tuning), and final evaluation of model performance across multiple metrics including accuracy, parameter count, FLOPs, and model size.

8.1 VGG-19 Evaluation

8.1.1 Unstructured Pruning

Table 1 presents the results of unstructured pruning experiments on the VGG-19 architecture using different post-pruning strategies.

Table 1: VGG-19 performance after unstructured pruning under various post-pruning strategies.

Method	Accuracy (%)	Error (%)	FLOPs (%)	Params (%)	Size (MB)
Without pruning	92.98	-7.02	100.0	100.0	548
Rewinding	93.17	-6.83	19.9	19.0	500
Random initialization	92.92	-7.08	19.9	19.0	500
Fine-tuning	92.85	-7.15	19.9	19.0	500

The application of unstructured pruning to the VGG-19 model achieved over 80% compression in terms of FLOPs and parameters. Among the retraining strategies, the rewinding method yielded the highest accuracy, slightly outperforming both random initialization and fine-tuning, and even exceeding the baseline model's original accuracy. Moreover, it demonstrated slightly faster convergence. However, due to the nature of unstructured pruning—where individual weights are removed rather than entire structures—the overall model size remained largely unchanged. This is because the weight matrices retain their original dimensions, limiting the benefits in memory reduction.

8.1.2 Structured Pruning

Table 2 summarizes the performance of the VGG-19 model following structured pruning, evaluated with the same three post-pruning strategies.

Table 2: VGG-19 performance after structured pruning under various post-pruning strategies.

Method	Accuracy (%)	Error (%)	FLOPs (%)	Params (%)	Size (MB)
Without pruning	92.98	-7.02	100.0	100.0	548
Rewinding	91.59	-8.41	19.82	19.82	24.1
Random initialization	90.94	-9.06	19.82	19.82	24.1
Fine-tuning	89.82	-10.18	19.82	19.82	24.1

As shown in Table 2, structured pruning achieves a compression factor of over 22 times in model size compared to the unpruned baseline. Among the retraining techniques, rewinding consistently outperforms random initialization and fine-tuning in terms of accuracy, though it does not fully recover the original model's performance. Rewinding also converges faster than the other methods. This performance gap is attributed to the nature of structured pruning, which removes entire filters rather than individual weights. While this simplifies network structure and significantly reduces memory footprint, it can result in a slight drop in classification accuracy. The overall trend is further illustrated in Figure 2.

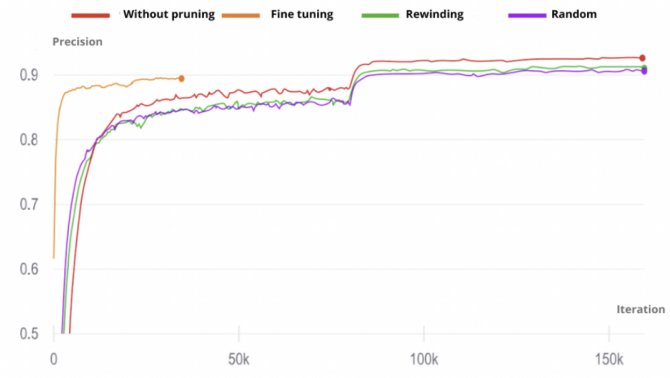


Figure 2: Accuracy comparison of post-pruning strategies for structured pruning on VGG-19.

8.1.3 Inference Time Analysis

Figure 3 presents a comparison of inference times across varying batch sizes for the original network, unstructured pruning, and structured pruning on the VGG-19 model. Structured pruning exhibits the lowest inference latency, followed by unstructured pruning, and finally the baseline unpruned model. The improved efficiency of structured pruning is attributed to the reduction in matrix size, leading to fewer operations. In contrast, unstructured pruning results in moderate gains due to the presence of zeroed weights, which marginally reduce computation.

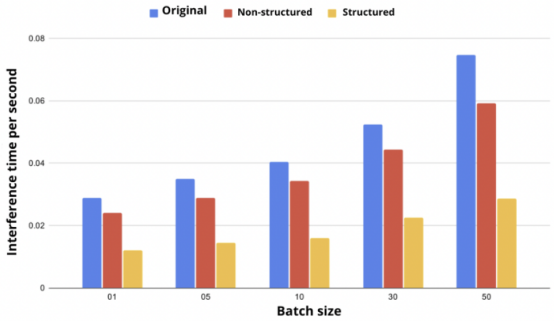


Figure 3: Inference time comparison of VGG-19 under different pruning strategies across batch sizes.

8.2 ResNet-34 Evaluation

8.2.1 Unstructured Pruning

Table 3 reports the performance of ResNet-34 following unstructured pruning using three retraining strategies.

Table 3: ResNet-34 performance after unstructured pruning with various post-pruning strategies.

Method	Accuracy (%)	Error (%)	FLOPs (%)	Params (%)	Size (MB)
Without pruning	87.20	-12.80	100.0	100.0	83.1
Rewinding	87.24	-12.76	17.46	17.75	83.1
Random Initialization	86.30	-13.70	17.46	17.75	83.1
Fine-tuning	86.61	-13.39	17.46	17.75	83.1

The unstructured pruning results for ResNet-34 closely resemble those observed for VGG-19. Among the post-pruning strategies, rewinding consistently delivers the highest accuracy, outperforming both random initialization and fine-tuning, as shown in Table 3 and Figure 4. However, it should be noted that rewinding experienced convergence challenges during the initial 60,000 training iterations.

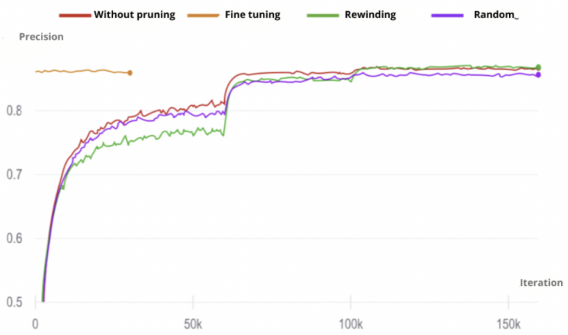


Figure 4: Accuracy trends of ResNet-34 under unstructured pruning using different retraining strategies.

8.2.2 Structured Pruning

Table 4 outlines the results of structured pruning on ResNet-34, which follow trends similar to those observed in the VGG-19 experiments.

Table 4: ResNet-34 performance after structured pruning with various post-pruning strategies.

Method	Accuracy (%)	Error (%)	FLOPs (%)	Params (%)	Size (MB)
Without pruning	87.20	-12.80	100.0	100.0	83.1
Rewinding	85.94	-14.06	18.95	18.95	6.2
Random Initialization	85.12	-14.88	18.95	18.95	6.2
Fine-tuning	85.35	-14.65	18.95	18.95	6.2

As summarized in Table 4, structured pruning on ResNet-34 results in a significant reduction in model size—over 13 times smaller than the original—while maintaining competitive accuracy. Rewinding again proves to be the most effective retraining strategy, yielding the highest accuracy and fastest convergence, as illustrated in Figure 5. The observed accuracy drop is attributed to the aggressive nature of structured pruning, where entire filters are removed, simplifying the model structure without considering individual weights.

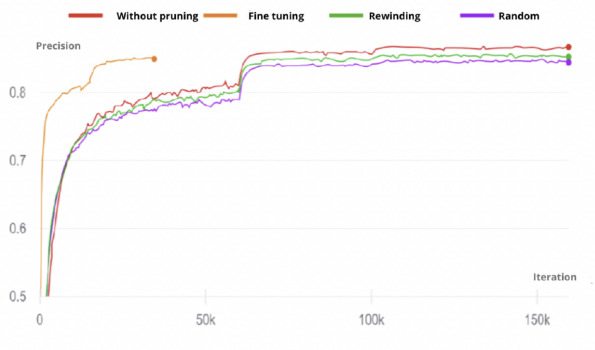


Figure 5: Accuracy trends of ResNet-34 under structured pruning using different retraining strategies.

8.2.3 Inference Time Analysis

To further evaluate the efficiency of pruning strategies, we compare inference times for ResNet-34 across different batch sizes. As shown in Figure 6, GPU usage did not significantly impact inference time, so CPUs were used to better highlight contrast. Structured pruning once again yielded the fastest inference, followed by unstructured pruning, and lastly the original model. This trend aligns with the observations for VGG-19.

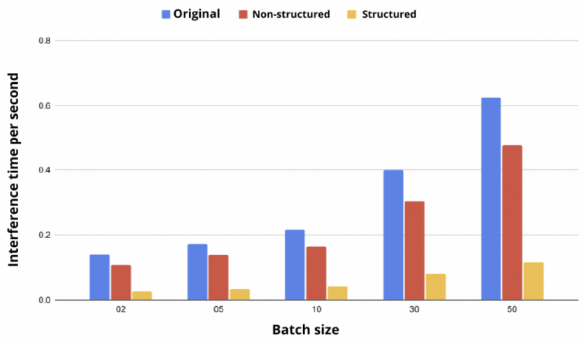


Figure 6: Inference time comparison of ResNet-34 under different pruning strategies across batch sizes.

9 Conclusion

This study highlights the importance of neural network compression for deployment on resource-constrained devices. Beginning with the Lottery Ticket Hypothesis, we explored pruning as a means of reducing the computational cost and memory footprint of deep neural networks while maintaining competitive performance. We evaluated multiple pruning techniques, including structured and unstructured methods, and investigated the role of graph theory through the integration of graph embeddings with reinforcement learning.

Our proposed approach enables efficient pruning by leveraging graph-level representations of CNN architectures and training an RL agent to make informed pruning decisions. The agent achieves up to 80% model compression while preserving accuracy.

Comparative experiments on VGG-19 and ResNet-34 architectures demonstrated the effectiveness of both pruning strategies. Structured pruning was particularly impactful, achieving up to a 13-fold reduction in model size with only a minor reduction in accuracy. Across all experiments, the rewinding strategy consistently outperformed random initialization and fine-tuning, confirming its robustness in post-pruning recovery.

References

[1] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[2] Rupesh Kumar Mishra, Hirdesh Kumar Patel Gupta, and Tapobrata Dutta. A survey on deep neural network compression. *arXiv preprint arXiv:2010.03954*, 2020.

[3] Jonathan Frankle. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

[4] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks. In *Advances in Neural Information Processing Systems 28 (NeurIPS)*, 2015.

[5] Hao Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach. *arXiv preprint arXiv:1607.03250*, 2016.

[6] Mingbao Lin, Rongrong Ji, Yan Wang, Baochang Zhang, Yongjian Wu, Shunyu Yao, and Qi Tian. Channel pruning via automatic structure search. *arXiv preprint arXiv:2001.08565*, 2020.

[7] Yuntao Chen, Xiaoyu Wen, Yujie Zhang, and Weisong Shi. Ccprune: Collaborative channel pruning for learning compact neural networks. *Neurocomputing*, 451:35–45, 2021.

[8] Moshe Looks, Marcello Herreshoff, DeLesley Hutchins, and Peter Norvig. Deep learning with dynamic computation graphs. *arXiv preprint arXiv:1702.02181*, 2017.

[9] Hongyun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding. *IEEE Transactions on Knowledge and Data Engineering*, 30:1616–1637, 2018.

[10] Shih-Chia Hsia, Shih-Hsuan Wang, and Chih-Yuan Chang. Convolutional neural network with low operation flops and high efficiency. *Journal of Real-Time Image Processing*, 18:1309–1319, 2021.

[11] Rajarshi Guha, David T. Stanton, and Peter C. Jurs. History of quantitative structure–activity relationships. *Journal of Chemical Information and Modeling*, 45(5):1109–1121, 2005.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Image fusion using a new evolution equation

Sabira Benalia[1] **and Mohammed Hachama**[2]

[1] ***USTHB, Laboratoire AMNEDP, Faculté de mathématiques, B.P. 32, El Alia, Bab Ezzouar, 16111 Alger, Algeria***

[2] ***National Higher School of Mathematics, Scientific and Technology Hub of Sidi Abdellah, P.O. Box 75, Algiers 16093,*** ALGERIA.

Abstract

In this paper, we solve the image fusion problem using a new mathematical model. We reformulate a recent osmosis model using nonlocal differential operators. Experimental results show that the nonlocal model obtained very good qualitative results compared with state-of-the- art models, including modern deep learning techniques.

Keywords: Image restoration, image fusion, Nonlocal differential operators, Energy minimization.

1 Introduction

Our objective is to fuse two grayscale images f (foreground) and b (background), which are real-valued functions defined on the same closed bounded regular domain $\Omega \subset \mathbb{R}^2$ (f and b are supposed in $L^\infty(\Omega,(0,+\infty))$). Note that the extension to color images can be easily accomplished by processing each color channel independently. The domain Ω is decomposed into three distinct regions: Ω_f which represents the image region copied from the foreground, Ω_b which represents the image region copied from the background, and Ω_{sb} is a transition region in which f and b are mixed (refer to Figure 1).

Figure 1: Image fusion results. From left to right: background, selected region, foreground, Seamless Poisson Editing [4], our nonlocal osmosis model.

2 Proposed model

We propose a new nonlocal model for image fusion which consists in minimizing the energy:

$$\mathcal{E}(u) := \mathcal{S}(u) + \lambda \mathcal{F}(u), \tag{1}$$

where $\mathcal{S}$ is a fusion term and $\mathcal{F}$ is a fidelity term. These terms are balanced using positive weight λ.

The fusion term: This term is a nonlocal osmosis model:

$$\mathcal{S}(u) = \frac{1}{2}\int_\Omega v(x) \left|\nabla_{NL}\left(\frac{u}{v}\right)\right|^2 (x)\ \mathrm{d}x, \tag{2}$$

where

$$v(x) = f^{\alpha(x)}(x) b^{1-\alpha(x)}(x) \tag{3}$$

and α is defined as follows:

$$\alpha(x) = \begin{cases} 1, & \text{if } x \in \Omega_f, \\ G(x), & \text{if } x \in \Omega_{sb}, \\ 0, & \text{if } x \in \Omega_b. \end{cases}$$

where the function G ensures a smooth transition on Ω_{sb} between 0 and 1[1], while $v = f$ on Ω_f and $v = b$ on Ω_b. The function v can be seen as a "rough solution" of the fusion problem.

Fidelity term: To ensure that the solution u should stay close to f in the foreground region, we minimize the following term

$$\mathcal{F}(u) = \frac{1}{2}\int_{\Omega} \frac{\alpha(x)}{v(x)} \left(f(x) - u(x)\right)^2 \, \mathrm{d}x. \tag{4}$$

In (2), $\nabla_{NL} : \Omega \times \Omega \to \mathbb{R}_+$ stands for a nonlocal gradient which permits to take into account nonlocal interactions between distant pixels. Here, we use the Gilboa operator [3] defined by

$$\nabla_{NL} u(x, y) := (u(y) - u(x)) \sqrt{w(x, y)}, \quad \forall x, y \in \Omega,$$

where $w : \Omega \times \Omega \to \mathbb{R}_+$ is a weight assumed to be symmetric. The inner product of two "vector" functions v, $v_1 : \Omega \times \Omega \longrightarrow \mathbb{R}$ is defined as

$$< v, v_1 >:= \int_{\Omega \times \Omega} v(x, y) v_1(x, y) \, \mathrm{d}x\mathrm{d}y.$$

The magnitude of v is a function $|v| : \Omega \to \mathbb{R}$ defined by

$$|v|(x) := \sqrt{\int_{\Omega} v(x, y)^2 \, \mathrm{d}y}.$$

The nonlocal divergence $\mathrm{div}_{\mathrm{NL}} v : \Omega \to \mathbb{R}$ is defined as the adjoint of the nonlocal gradient:

$$(\mathrm{div}_{NL} v)(x) = \int_{\Omega} \left(v(x, y) - v(y, x)\right) \sqrt{w(x, y)} \, \mathrm{d}y.$$

The nonlocal Laplacian $\triangle_{NL} u$ is a function defined on Ω by

$$\triangle_{NL} u(x) := \frac{1}{2}(\mathrm{div}_{NL} \nabla_{NL} u)(x) = \int_{\Omega} \left(u(y) - u(x)\right) w(x, y) \, \mathrm{d}y.$$

where the factor $\frac{1}{2}$ is introduced to be consistent with the graph Laplacian definition.

2.1 Evolution problem

In this section, we derive the evolution equation associated with (1), which can be seen as a "continuous" gradient descent algorithm. We seek a time-dependent solution $u(t, \cdot)$ that evolves over time toward the minimizer of the energy (1).

Proposition 1 *For $f, b \in L^\infty(\Omega,\ \mathbb{R}_+)$, the evolution process associated to 1 is defined as follows:*

$$\begin{cases} \partial_t u\,(t, x) &= \mathrm{div}_{NL}\left(v(x) \nabla_{NL}\left(\dfrac{u(t, .)}{v}\right)(x, y)\right) + \lambda \alpha(x)(f(x) - u(t, x)) & \text{in } \Omega_t, \\ u(0, x) &= f(x), & \text{in } \Omega, \end{cases} \tag{5}$$

where $\Omega_t = (0, T) \times \Omega$.

Proof 1 *Let $\psi \in C_c^\infty(\Omega)$ be a test function and $t > 0$. For convenience, we may drop the the notation of the variable t. We compute the Gâteaux derivative of $\mathcal{E}(u)$ at u in the direction ψ. The Gâteaux variations of $\mathcal{F}$ is straightforward:*

$$\mathcal{F}'(u) = \frac{\alpha}{v}(u - f).$$

[1] The exact definition of G is given in the Section dedicated to the numerical resolution.

On the other hand, we have

$$\begin{aligned}
\mathcal{S}^{'}(u)(\psi) =& \lim_{t\to 0} \frac{\mathcal{S}(u+t\psi)-\mathcal{S}(u)}{t},\\
=& \iint_{\Omega\times\Omega} v(x)\left(\frac{u(y)}{v(y)}-\frac{u(x)}{v(x)}\right)\left(\frac{\psi(y)}{v(y)}-\frac{\psi(x)}{v(x)}\right)\omega(x,y)\ \mathrm{d}y\mathrm{d}x,\\
=& \iint_{\Omega\times\Omega} v(x)\left(\frac{u(y)}{v(y)}-\frac{u(x)}{v(x)}\right)\frac{\psi(y)}{v(y)}\omega(x,y)\ \mathrm{d}y\mathrm{d}x\\
&-\iint_{\Omega\times\Omega}\left(\frac{u(y)}{v(y)}-\frac{u(x)}{v(x)}\right)\psi(x)\,\omega(x,y)\ \mathrm{d}y\mathrm{d}x,\\
=& \iint_{\Omega\times\Omega} v(y)\left(\frac{u(x)}{v(x)}-\frac{u(y)}{v(y)}\right)\frac{\psi(x)}{v(x)}\omega(y,x)\ \mathrm{d}x\mathrm{d}y\\
&-\iint_{\Omega\times\Omega}\left(\frac{u(y)}{v(y)}-\frac{u(x)}{v(x)}\right)\psi(x)\,\omega(x,y)\ \mathrm{d}y\mathrm{d}x,\\
=& \int_\Omega\left(-\frac{1}{v(x)}\int_\Omega v(x)\left(\frac{u(y)}{v(y)}-\frac{u(x)}{v(x)}\right)w(x,y)\ \mathrm{d}y\right)\psi(x)\ \mathrm{d}x\\
&+\int_\Omega\left(\frac{1}{v(x)}\int_\Omega\left(v(y)\left(\frac{u(x)}{v(x)}-\frac{u(y)}{v(y)}\right)\right)w(x,y)\ \mathrm{d}y\right)\psi(x)\ \mathrm{d}x,\\
=& \int_\Omega\left(-\frac{1}{v(x)}div_{NL}\left(v\nabla_{NL}\left(\frac{u}{v}\right)\right)(x)\right)\psi(x)\ \mathrm{d}x.
\end{aligned}$$

Thus, we obtain:

$$\mathcal{E}^{'}(u) \quad = \quad -\operatorname{div}_{NL}\left(v\nabla_{NL}\left(\frac{u}{v}\right)\right)-\lambda\alpha(f-u).$$

The Euler-Lagrange equation $\mathcal{E}'(u)=0$ is difficult to solve. Consequently, we use the suboptimal gradient descent procedure

$$\partial_t u\,(t,x) = -\mathcal{E}'(u).$$

Thus, we obtain the system (5).

3 Numerical resolution

Let Ω_{d} be a discrete grid associated to the continuous domain Ω:

$$\Omega_{\mathrm{d}} = \{1,\dots,M\}\times\{1,\dots,N\}.$$

We use the following notations:

$$\mathbf{i}=(i_1,i_2)\in\Omega_d,\ \mathbf{x_i}=(x_{i_1},y_{i_2})\in\Omega,\ u_{\mathbf{i}}\approx u(\mathbf{x_i}),\ \alpha^{\mathbf{i}}\approx\alpha(\mathbf{x_i}).$$

The image f is approximated by a discrete matrix f_d.

The weight function ω defines a similarity between two pixels $\mathbf{x_i}$ and $\mathbf{x_j}$ by comparing their neighborhoods:

$$\omega_{\mathbf{i},\mathbf{j}} \quad = \quad \omega\big(\mathbf{x_i},\mathbf{x_j}\big) \quad = \quad \exp\left(-\frac{1}{h^2}\sum_{t_1,t_2=-r}^{r} G_\sigma(t_1,t_2)\big(f(\mathbf{x_{i+t}})-f(\mathbf{x_{j+t}})\big)^2\right), \quad (6)$$

where r defines the neighborhood, $\mathbf{t}=(t_1,t_2)$, and $h>0$ is a scale parameter and G_σ is a gaussian function ($G_\sigma(t_1,t_2)=\dfrac{1}{2\pi\sigma^2}e^{-\frac{t_1^2+t_2^2}{2\sigma^2}}$, with $\sigma>0$).

The approximation of the nonlocal Laplacian is given by:

$$\Delta_{NL}u\,(\mathbf{x_i})\approx(\Delta_{NL}u)_{\mathbf{i}}=\sum_{\mathbf{j}\in\Omega_d}(u_{\mathbf{j}}-u_{\mathbf{i}})\omega_{\mathbf{i},\mathbf{j}}.$$

The nonlocal divergence of $p:\Omega\times\Omega\to\mathbb{R}$ is approximated by

$$\operatorname{div}_{NL}(p)\,(\mathbf{x_i})\approx(\operatorname{div}_{NL}(p))_{\mathbf{i}}=\sum_{\mathbf{j}\in\Omega_d}(p_{\mathbf{i},\mathbf{j}}-p_{\mathbf{j},\mathbf{i}})\sqrt{\omega_{\mathbf{i},\mathbf{j}}}.$$

We approximate space derivative by forward differences and time derivative using explicit Euler method. The discrete iterative scheme of (5) writes:

$$\begin{aligned} u_{\mathbf{i}}^{n+1} &= u_{\mathbf{i}}^{n} + \frac{dt}{dx}\left((\Delta_{NL}u^n)_{\mathbf{i}} + \sum_{\mathbf{j}\in\Omega_d} \frac{v_{\mathbf{i}}}{v_{\mathbf{j}}} u_{\mathbf{j}}^n \omega_{\mathbf{i},\mathbf{j}} - \sum_{\mathbf{j}\in\Omega_d} \frac{v_{\mathbf{j}}}{v_{\mathbf{i}}} u_{\mathbf{i}}^n \omega_{\mathbf{i},\mathbf{j}} \right) \\ &+ dt\ \lambda \alpha^{\mathbf{i}} (f_{\mathbf{i}} - u_{\mathbf{i}}^n) \end{aligned}$$

where dt, dx are the discretization steps, n is the time index.

Then, the discrete problem can be written as algebraic equations of the form

$$U^{n+1} = \mathbf{A}U^n + \mathbf{P}F,$$

where $U \in \mathbb{R}^m$, $m = M \times N$, is the vector obtained by concatenating the columns of the discrete image u, n is the time index, $F = (f_{\mathbf{i}})_{1\leq \mathbf{i} \leq m}$ and the initial condition is set to $U^0 = F$.

4 Experimental results

We conducted extensive experiments to evaluate our model and compare it to previous state-of-the-art techniques: Poisson Image Editing [2], Parisotto et al. [4]. We implemented our algorithm using Matlab 2020a, but the weight function had been implemented as a c-mex file (C-program). Simulations have been conducted on a Core i5-4210 (2.60 GHz) processor with 4GO RAM.

Based on an empirical analysis, we determined the values of the parameter λ and found that $\lambda \in [0, 1]$. For the weight function, we choose, in most cases, a patch size of 5×5, search window of size 7×7, $\sigma = 0.003$ and the filtering parameter $h = \sigma$.

We used the software and test data provided by the authors [2], [3], [4]. In some approaches, the authors presented an automatic method for selecting region. However, in this work, we assume that the subdomain is given to focus on the numerical solution of the nonlocal equation . Let's montion, for the images that have not the same size, we create a new image foreground with the same size of background and with the centroid of selected region in the position with othor coordinates.

The function α allows to indicat from which of the two images the structural information comes from. In the case where $\Omega_{sb} = \emptyset$, α is binar, i.e. $\alpha(x) \in \{0, 1\}$, so α vanishes on the pixels of the background image b and is equal to one otherwise. Alternatively, α can be smoothed by means of a Gaussian convolution so as to favour a smooth transition on $\Omega_{sb} \neq \emptyset$.

Experimental results obtained by the nonlocal model are presented on Figure 2 (qualitative evaluation). As demonstrated visually, our method outperforms previous state-of-the-art techniques in most cases. It produces a better skin tone than the seamless Poisson editing model and Parisotto et al. [4] model as showing in Figure 2(a) and Figure 2(b). It is a powerful tool for manipulating colors, two differently colored versions of those images can be mixed seamlessly. For exemple, in Figure 2(d), our model overcome the problem of the undesirable visible seam of Parisotto et al. [4] model. An example is shown in Figure 2(e), in which our model can facilitate the transfer of partly transparent objects (the rainbow).

4.1 Comparison with deep learning techniques

We also assess the performance of the proposed image fusion method using the Lytro[5] dataset.The fusion results of the proposed method are compared with three recently developed fusion algorithms. The first one is GFDF [5]. The second compared algorithm is DCT_EOL [1]. The third method is CNN [6]. All image fusion results[6] of these algorithms are available online. The results are presented on Figure 3.

[3] https://doi.org/10.5201/ipol.2016.163

[4] http://cs.brown.edu/courses/cs129/results/proj2/damoreno/

[5] https://sandipanweb.wordpress.com/2017/10/03/some-variational-image-processing-possion-image-editing-and-its-applications/

[5] https://www.researchgate.net/publication/291522937_Lytro_Multi-focus_Image_Dataset

[6] https://github.com/xingchenzhang/MFIF

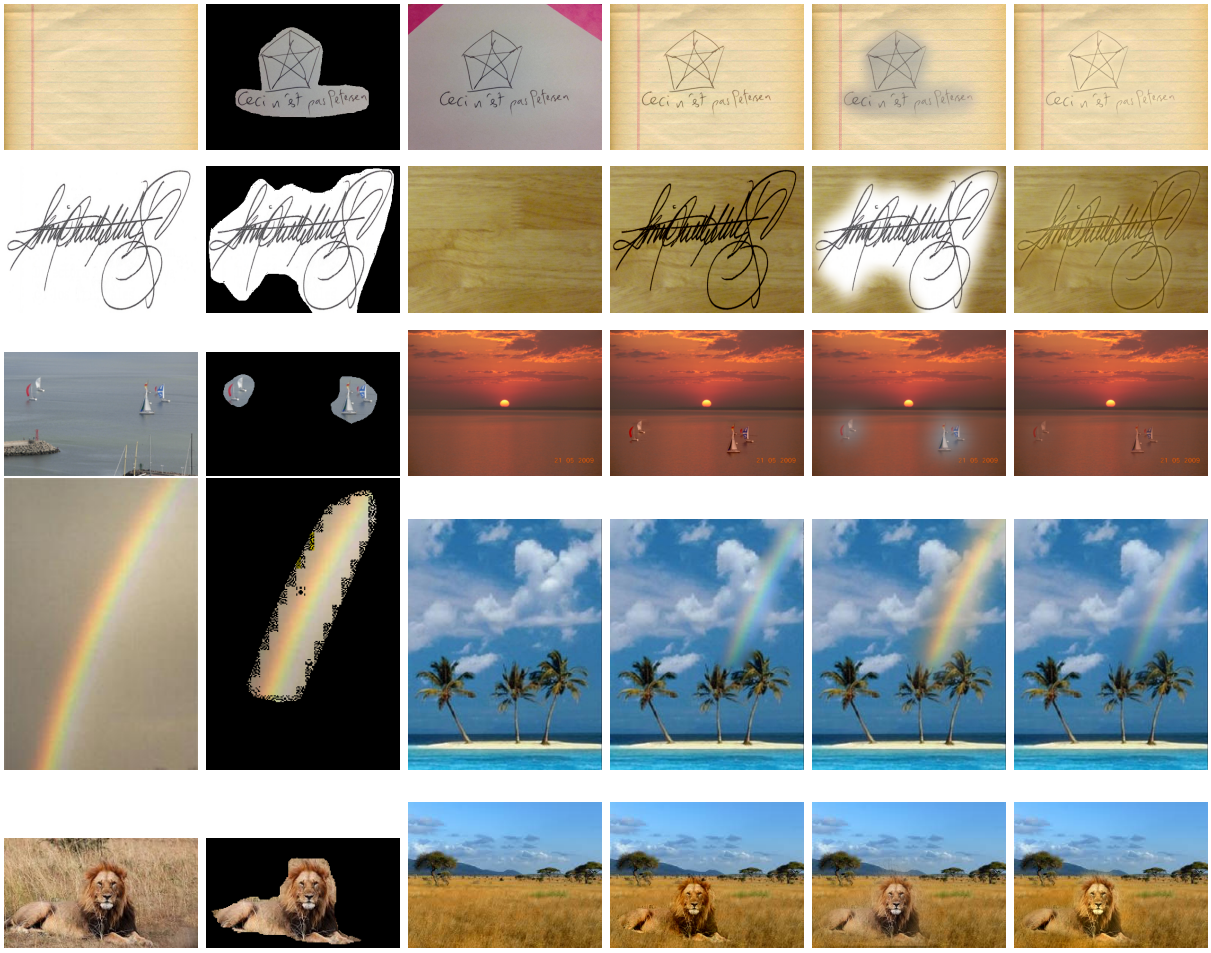


Figure 2: Image fusion results of different models ((1)-(8)). From left to right: background, selected region, foreground, Seamless Poisson Editing [2], Parisotto et al. [4], nonlocal osmosis model (1).

Figure 3: Qualitative comparison of results of Lytro dataset. (1) and (2) are: Foreground and background. From (3) to (6) are fused images obtained by: GFDF [5], DCT_EOL [1], CNN [6], nonlocal osmosis model (1).

5 Conclusion

In this paper, we presented a novel nonlocal model for image fusion that utilizes nonlocal derivatives in the recently developed osmosis model. Experimental analyses have shown that our proposed model demonstrates its effectiveness and superiority over local image fusion models. Our proposed method provides visually plausible image data fusion that is invariant to multiplicative brightnessb changes.

References

[1] M. Amin-Naji and A. Aghagolzadeh. Multi-focus image fusion in dct domain using variance and energy of laplacian and correlation coefficient for visual sensor networks. *Journal of AI and Data Mining,*, pages 33–250, 2018.

[2] J. Matas Di Martino, Gabriele Facciolo, and Enric Meinhardt-Llopis. Poisson Image Editing. *Image Processing On Line*, pages 300–325, 2016.

[3] Guy Gilboa and Stanley Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2009.

[4] Simone Parisotto, Luca Calatroni, Aurélie Bugeau, Nicolas Papadakis, and Carola-Bibiane Schönlieb. Variational osmosis for non-linear image fusion. *IEEE Transactions on Image Processing*, pages 5507–5516, 2020.

[5] L. Zhang X. Qiu, M. Li and X. Yuan. Guided filter-based multi-focus image fusion through focus region detection. *Signal Processing: Image Communication*, pages 35–46, 2019.

[6] H. Peng Y. Liu, X. Chen and Z.Wang. Multi-focus image fusion with a deep convolutional neural network. *Information Fusion,*, pages 191–207, 2017.

MambaCT: Feature Enhancement-Based Low-Dose CT Image Denoising Using Vision Mamba and a Scaling Adapter

Abdelkarim Cherhabil[1], Lahcène Mitiche[1], and Amel Baha Houda Adamou-Mitiche[1]

[1] *Department of Electronics and Telecommunications, Ziane Achour University of Djelfa, Laboratoire de Modélisation, Simulation et Optimisation des Systèmes Complexes Réels*

Abstract

With the rapid development of Mamba models, Vision Mamba is replacing convolutional neural networks (CNNs) and vision transformers (ViTs), emerging as the dominant trend in computer vision tasks. After achieving remarkable success in natural language processing, Mamba models have garnered increasing interest in the medical imaging community for their ability to understand global context. However, there has been limited research on medical image denoising based on the Vision Mamba architecture. In this paper, we propose MambaCT, a model that integrates the key features of UNet and Mamba with a Scaling Adapter in the visual state space (VSS) Block. Additionally, we propose skip connection spatial-channel processing attention (SCSPA) to enhance feature integration and robustness as a pathway in place of traditional skip connections. MambaCT outperforms previous state-of-the-art (SOTA) models across various architectures in both visual quality and quantitative performance, requiring only 0.83G MACs and achieving an SSIM of 0.9104 and RMSE of 9.3423. The model was evaluated on the AAPM-Mayo Clinic low-dose computed tomography (LDCT) Grand Challenge Dataset.

Keywords: Low-dose CT, Vision Mamba, Medical Image Denoising, Adapter, State Space Models, Auto-encoder.

1 Introduction

Computed tomography is a diagnostic imaging method that precisely aligns X-ray, gamma, ultrasound, and ion beams to create cross-sectional images of the human body [1]. Clinical, industrial, and other fields make extensive use of CT [1] [31]. It is particularly effective in reconstructing organ structures at various depths and angles [2] [3]. Several algorithms have been created to improve image quality in low-dose CT (LDCT) scans in order to address this issue. Using physical models and existing data, researchers employ iterative techniques in classical ways to reduce noise and artifacts. For instance, some image priors are expressed as sparse transforms utilizing compressive sensing (CS) to address issues in internal CT, low-dose, few-view, and finite-angle CT [4]. Examples include dictionary learning [5], low-rank [6], non-local means (NLM) [7][8][9], total variation (TV), and its variations [10][11][12][13], among other methods. Li et al. [49] reconstructed feature similarities in large neighborhood images using NLM. Aharon et al. used dictionary learning [50] to denoise LDCT images, drawing inspiration from sparse representation theory, resulting in considerable improvement in denoising quality while reconstructing abdominal images [51]. Block-matching 3D (BM3D) has been shown by Feruglio et al. to be efficient for a range of X-ray imaging applications [52]. However, this method's inaccuracy in determining the noise distribution in the image domain prevents the optimal balance between noise reduction and structure preservation. Due to restrictions on data volume, the accuracy of these conventional approaches is typically still poor [14].

Since the advent of deep learning, CNNs have been the dominant method for denoising low-dose CT (LDCT) images. CNNs extract features through convolutional operations, where the kernel moves across the entire image, resulting in a relatively small parameter volume for the CNN model. This approach effectively captures significant local features, and the network's receptive field is incrementally expanded through layer stacking. Numerous traditional deep learning algorithms have been applied to the field of low-dose CT (LDCT) image denoising for reconstructing high-quality images. These include convolutional neural networks (CNNs) [15] [16] [29] [30], encoder-decoder networks with residual connections [17][18][19], and generative adversarial networks (GANs) [20][21]. The work of Chen et al. can be considered groundbreaking, as they were among the first to utilize convolution, deconvolution, and shortcut

connections to design a prototype of a residual encoder-decoder convolutional neural network, known as RED-CNN [17]. To improve the quality of denoised photos, Yang et al. used a generative adversarial network with Wasserstein distance (WGAN) and a perceptual loss mechanism [20]. Compared to other CT denoising techniques, Fan et al. developed a quadratic neuron-based autoencoder that is more resilient and useful for model efficiency [**?**]. The retrieval of detailed structural details in the denoised images may be adversely affected by CNNs' limits in capturing long-range contextual information within images, notwithstanding their intriguing results for LDCT [23].

The integration of Transformer models into image denoising has significantly improved performance, resulting in higher accuracy and reduced processing times [24][25]. This advancement has brought about a revolution in image processing. Recent studies reveal that Transformer modules can effectively replace traditional convolutions in deep neural networks. They work by processing sequences of image patches, leading to the development of Vision Transformers (ViTs). Dosovitskiy et al. first proposed the vision transformer (ViT) in the CV field by mapping an image into 16×16 sequence words [24]. Wang et al. propose an innovative approach called the Convolution-free Token2Token Dilated Vision Transformer [26]. Luthra et al. introduce a fresh approach named Eformer, which stands for Edge Enhancement-based Transformer. Eformer is a unique architectural framework that constructs an encoder-decoder network using transformer blocks [27]. Jian et al. propose SwinCT, which utilizes a feature enhancement module (FEM) inspired by the Swin Transformer architecture. The FEM in SwinCT is employed to capture and enrich the high-level features within medical images [28].

According to the above analysis, Mamba models offer significant advantages over both CNN and transformer models, including greater visual interpretability due to their intrinsic Visual State Space (VSS) blocks [32]. Beyond their effectiveness, Mamba models are appealing to physicians because their self-explanatory nature allows doctors to understand the model's reasoning. Öztürk et al. [33] pioneered the application of an innovative SSM architecture, named DenoMamba, to enhance LDCT image denoising without increasing model complexity. This novel approach employs an hourglass-shaped structure, featuring encoder-decoder stages built with custom-designed FuseSSM blocks. Li et al. [34] introduced a CACTSR, which integrates VMamba and Transformer technologies with Mixed Attention Blocks and Cross Attention Blocks to enhance feature utilization and facilitate cross-window information interaction. The above studies demonstrate the promising results of Mamba-based deep learning models in visual tasks.

Motivated by the aforementioned study, we introduce MambaCT, a paradigm that combines the key components of Mamba and UNet by integrating a Scaling Adapter within the VSS Block. Our approach incorporates an SCSPA module pathway in place of traditional skip connections, resulting in images that exhibit superior performance both quantitatively and visually. According to experimental results, our model outperforms other state-of-the-art models, achieving the highest SSIM value and the lowest RMSE value.

The main contributions of this paper are as follows:

1. We propose MambaCT, a model that utilizes a U-Net-based Mamba network architecture and incorporates a Scaling Adapter within the VSS Block. The Scaling Adapter enhances the restoration of detailed and structural information in denoised images.

2. We propose the Skip Connection Spatial-Channel Processing Attention (SCSPA) module pathway as an alternative to traditional skip connections.

3. Assess the model's performance by comparing it with previous works using various metrics, including MACs, SSIM, and RMSE.

2 Methods

The architecture of the proposed MambaCT, as shown in Fig. 1, draws inspiration from both U-Net [35] and VMamba [36]. Designed specifically for LDCT denoising, MambaCT comprises four key modules: 1) Patch Extraction, 2) VSS Blocks, 3) SCSPA Module, and 4) Resizing Modules.

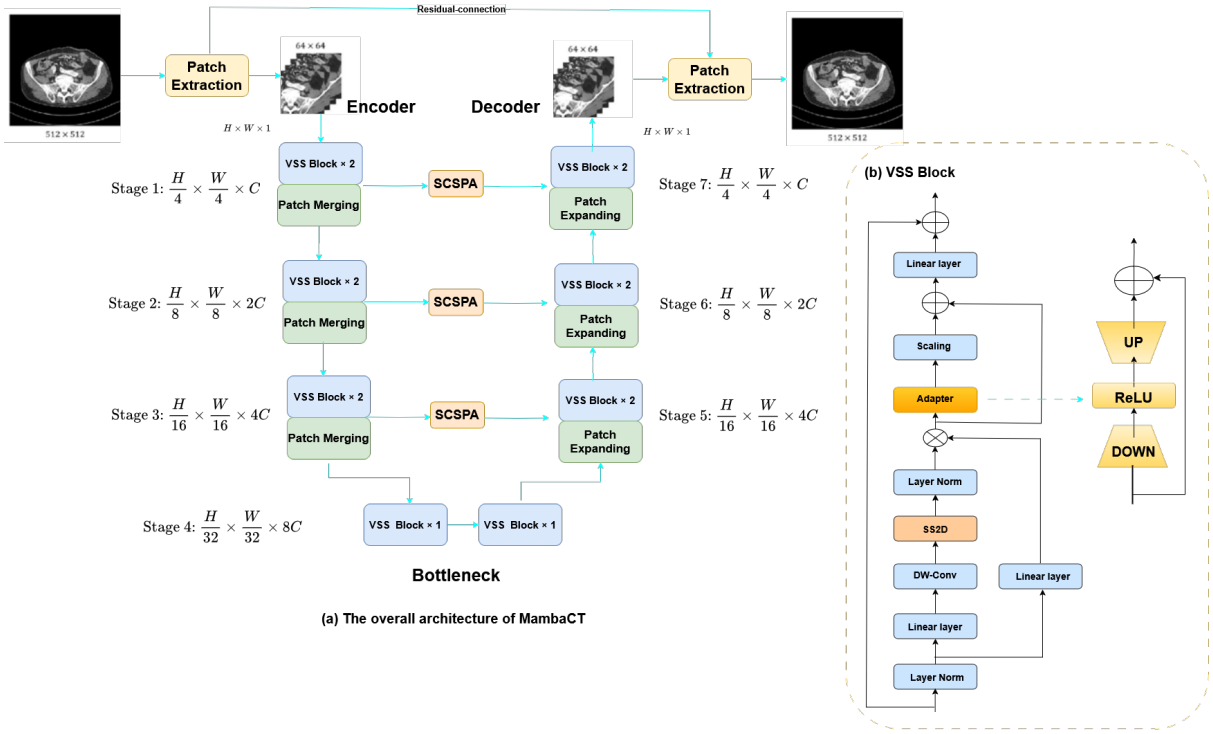


Figure 1: The overall structure of MambaCT (a).The VSS Block serves as the primary building block of MambaCT, with SS2D and the Adapter as its core operations (b).

2.1 Patch extraction

To train deep learning models effectively, a large volume of samples is crucial, which can be particularly challenging in clinical imaging. In our study, we addressed this issue by using CT scans with overlapping slices. This method has proven to be both effective and successful, as it helps capture perceptual differences in local regions and greatly boosts the number of samples available [37][38][39].

2.2 VSS Block

The core of MambaCT is the VSS Block, which serves as the primary building block of the model. The VSS Block incorporates SS2D and the Adapter as its core operations and is derived from [36], as shown in Figure 1(b). Its structure begins with Layer Normalization, followed by a split into two branches. The first branch applies a linear layer and the activation function SiLU [40]. The second branch processes the input through a linear layer, depthwise separable convolution, activation, and the SS2D module. the SS2D module provides contextual information to image patches via a compressed hidden state along scanning paths (Figure 2(b)), reducing computational complexity from quadratic to linear compared to self-attention mechanisms (Figure 2(a)). After SS2D, the features undergo Layer Normalization and are combined with the first branch's output through element-wise multiplication. A Scaling Adapter is then applied, allowing for the learnable adjustment of the adapted features' contribution. Directly after the adapter, we scale the embedding by a scale factor s [41]. Finally, the result passes through a linear mixing layer and is combined with a residual connection to produce the block's output. this architecture efficiently processes spatial information while maintaining linear complexity, making it well-suited for medical image denoising in LDCT.

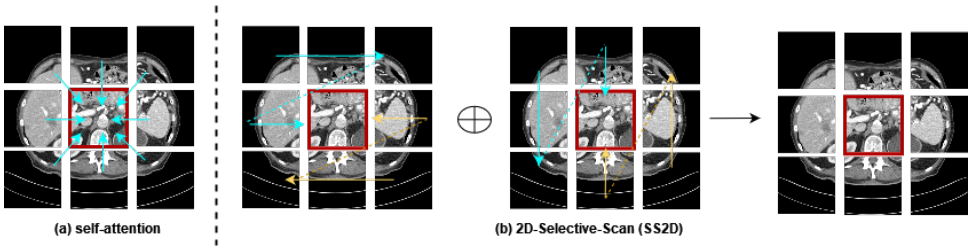


Figure 2: Comparison of correlation establishment between image patches via (a) self-attention and (b) the proposed 2D-Selective-Scan (SS2D). Red boxes indicate the query image patch, with patch opacity representing the degree of information loss.

2.2.1 2D-Selective-Scan for Vision Data

A scan expansion operation, an S6 block, and a scan merging operation are the three primary parts of the SS2D module. According to Figure 3, SS2D first unfolds input patches into sequences along four distinct traversal paths (i.e., scan expanding), processes each patch sequence using a separate S6 block in parallel, and then reshapes and merges the resultant sequences to form the output map (i.e., scan merging). By adopting complementary 1D traversal paths, SS2D enables each pixel in the image to effectively integrate information from all other pixels in different directions, facilitating the establishment of global receptive fields in the 2D space.

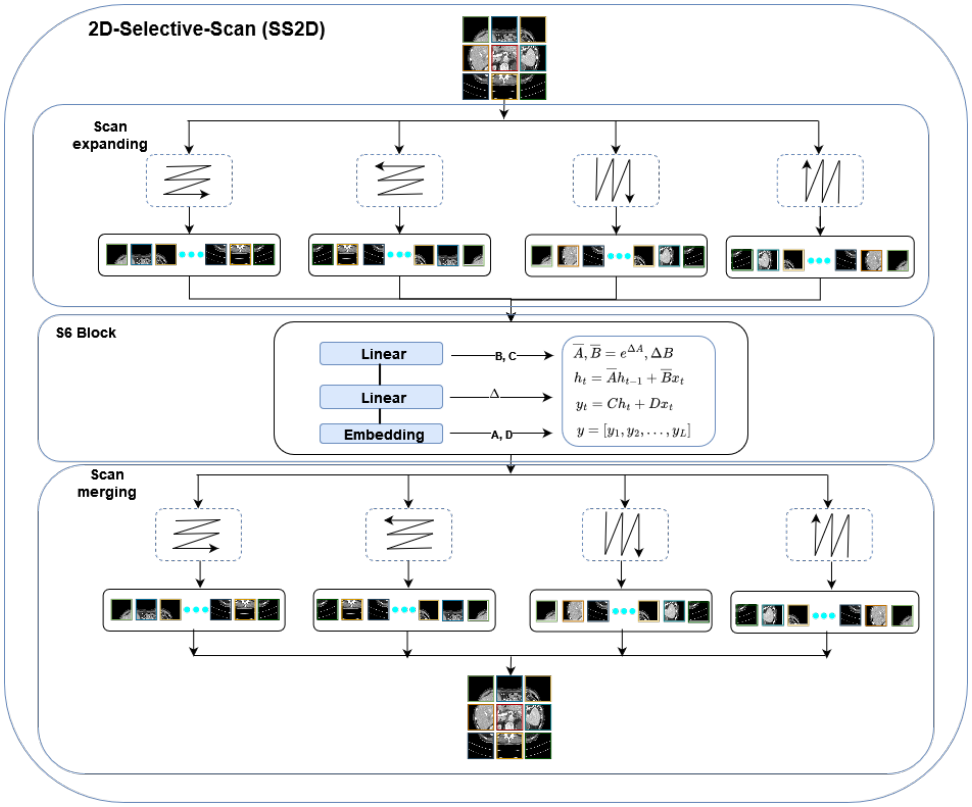


Figure 3: The overall structure of the 2D Selective Scan (SS2D) process.

2.2.2 Scaling Adaptor

The Adapter operates as a bottleneck model. The down-projection layer reduces the dimensionality of the input embedding using a basic MLP layer with parameters $W_{\text{down}} \in \mathbb{R}^{d \times \hat{d}}$, and the up-projection layer restores the compressed embedding to its original dimensionality with an additional MLP layer with parameters $W_{\text{up}} \in \mathbb{R}^{\hat{d} \times d}$, where $\hat{d}$ is the bottleneck middle dimension and satisfies $\hat{d} < d$. Additionally, there is a ReLU layer [42] between these projection layers for non-linear properties. Residual connections remain a crucial aspect, helping in training deeper networks by preventing gradient vanishing problems.

2.3 Skip Connection Spatial-Channel Processing Attention

In contrast to using a single attention mechanism, the combination of channel attention and spatial attention, especially in a sequential manner, significantly enhances the model's ability to capture important feature information [43]. Inspired by [44], we propose a SCSPA mechanism that applies sequential channel-spatial attention to the skip connections. As illustrated in Fig. 4, the SCSPA module consists of two key components: one for spatial attention and one for channel attention. A channel reduction action (C $\rightarrow$ C/rate), a ReLU activation, and a channel expansion operation (C/rate $\rightarrow$ C) comprise the Channel Attention Submodule. The two 7x7 convolutions that make up the Spatial Attention Submodule are followed by batch normalization (BN) and ReLU activation in the first one, and batch normalization and a sigmoid activation in the second. After that, element-wise multiplication is used to merge the outputs of these two routes. The dimensions of the input and the final output are [B, H, W, C].

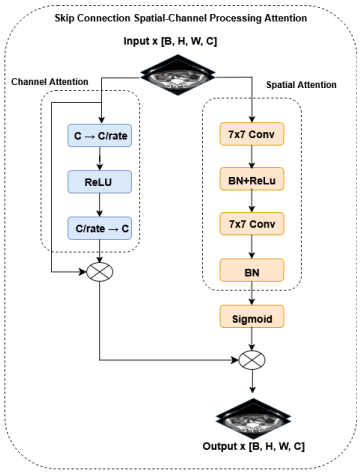


Figure 4: The overall structure of SCSPA.

2.4 Resizing module

The Patch Merging components function as downsampling mechanisms, diminishing the spatial dimensions of feature maps while amplifying the channel count. This approach enables the network to capture hierarchical features across various scales. Conversely, the Patch Expanding modules in the decoder act as counterparts to the Patch Merging modules in the encoder. They reverse the downsampling process, progressively restoring spatial resolution while decreasing the number of channels.

3 Experiment

In this section, we begin by listing the languages and tools utilized: Python, PyTorch, and CUDA.

Dataset: The publicly accessible clinical dataset from the 2016 NIH-AAPM Mayo Clinic LDCT Grand Challenge was used to train and evaluate the model [45]. Ten anonymous individuals' 2,378 low-dose (quarter) and 2,378 normal-dose (full) CT scans with 3.0-mm whole-layer slices are included in this dataset. We chose patient L506's data, which consists of 211 slice images with numbers ranging from 000 to 210, for testing. The model was trained using the data from the remaining nine cases.

Experiment setup: PyTorch 1.11.0 [46] and CUDA 12.4.0 were used in the experiments, which were conducted on an Ubuntu 22.04 LTS system with an Intel(R) Core(TM) i7-12700k CPU @ 2.70 GHz. The four NVIDIA RTX 3070 Ti 8G GPUs were used to train the model. Four blocks were chosen at random from each image's available slices for training. For 4,000 epochs, the batch size was fixed at 16. The ADAM-W optimizer, which has a learning rate of 1.0×10^{-5}, was used to reduce the mean squared error loss. After training, the model's performance was assessed using the standard metrics in the field.

4 Discussion

To evaluate denoising performance in LDCT images, we retrained all models using their officially available code. We propose MambaCT, a model that combines key features from UNet and Mamba, enhanced with a Scaling Adapter in the VSS Block. Additionally, our approach incorporates an SCSPA module pathway in place of traditional skip connections to improve feature integration. As shown in Table 1 for the L506 dataset, MambaCT achieved the highest quantitative metrics, surpassing all other methods.

Table 1: Quantitative comparison of different methods on L506 in terms of learnable parameters (#param.), MACs, SSIM, and RMSE. Bold values represent our method's performance.

Method	#param.	MACs	SSIM↑	RMSE↓
LDCT	–	–	0.8759	14.2416
RED-CNN [17]	1.85M	5.05G	0.8952	11.5926
WGAN-VGG [20]	34.07M	3.61G	0.9008	11.6370
MAP-NN [47]	3.49M	13.79G	0.8941	11.5848
AD-NET [48]	2.07M	9.49G	0.9041	9.7166
MambaCT	**62.08M**	**0.83G**	**0.9104**	**9.3423**

To provide a thorough evaluation of denoising performance, we use both qualitative and quantitative methods. The quantitative analysis focuses on two key metrics: SSIM, and RMSE. Additionally, model complexity is assessed based on the number of trainable parameters (#param.) and multiply-accumulate operations (MACs). Table 1 presents the average SSIM, and RMSE across all slices of L506. Our MambaCT model achieves the highest SSIM of 0.9104, the lowest RMSE of 9.3423.

Figure 5 presents the results of various networks on L506 with Lesion No. 575, while Figure 6 displays the regions of interest (ROIs) from the rectangular area highlighted in Figure 5.

Visual analysis of these figures 5 and 6 demonstrates MambaCT's superior capability in achieving three key objectives: noise and artifact removal, maintenance of high-level spatial smoothness, and preservation of target image details.

While RED-CNN, built on convolutional networks, shows proficiency in noise and artifact elimination while retaining image details, it faces limitations in structural recovery. This constraint stems from its computational architecture, which prioritizes high-frequency information extraction, such as texture details. Furthermore, RED-CNN's effectiveness is hampered by its finite receptive field size, impeding comprehensive global information capture.

Detailed examination of the ROIs in Figure 6 reveals varying performance across methods: 1. WGAN-VGG and MAP-NN introduce unwanted artifacts, manifesting as additional shadows and tissue-like structures. 2. RED-CNN and AD-NET yield improvements in image clarity and smoothness compared to WGAN-VGG and MAP-NN, though residual blotchy noise persists around lesion areas.

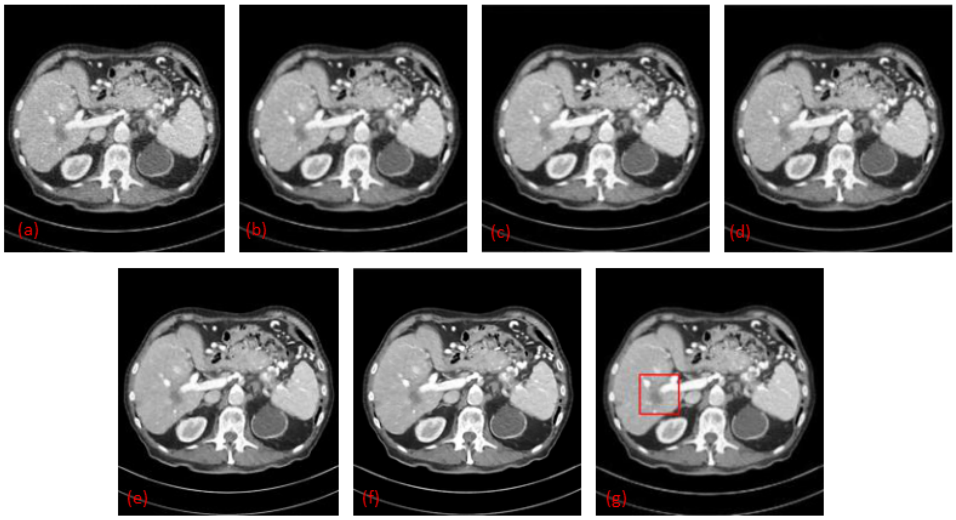


Figure 5: various networks' denoised findings on L506 with Lesion No. 575. These include LDCT (a), RED-CNN (b), WGAN-VGG (c), MAP-NN (d), AD-NET (e), MambaCT (f), and NDCT (g). The window for display is [-160, 240] HU.

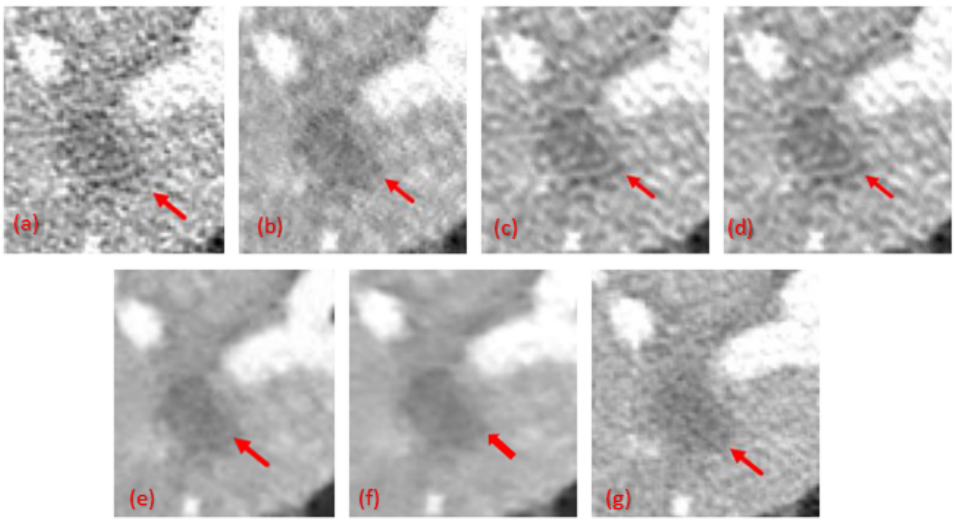


Figure 6: Fig. 5 shows the ROIs of the rectangle. These include LDCT (a), RED-CNN (b), WGAN-VGG (c), MAP-NN (d), AD-NET (e), MambaCT (f), and NDCT (g).

Comparatively, MambaCT performs best on all metrics: it effectively suppresses noise and artifacts, maintains high-level spatial smoothness, and preserves structural information in images that have been restored. The quantitative measurements shown in Table 1, where MambaCT consistently performs better than other comparative models.

Concerning model complexity, MAP-NN has the highest MACs at 13.79G due to its numerous repeated modules, While MambaCT has the highest number of parameters at 62.08M, it remarkably uses the least MACs at 0.83G, demonstrating its computational efficiency. while WGAN-VGG has the greatest number of trainable parameters at 34.07M due to its use of VGG as a feature extractor. This balance between high performance and low complexity underscores the efficiency of MambaCT compared to other state-of-the-art methods, such as MAP-NN and WGAN-VGG and AD-NET, which exhibit higher complexity but lower performance.

5 Conclusion

In this work, we propose MambaCT, a model that integrates a Scaling Adapter within the VSS Block, combining the essential elements of Mamba and UNet. To further enhance feature integration, our method replaces conventional skip connections with an SCSPA module pathway, resulting in images that demonstrate superior performance both quantitatively and visually. Experimental results indicate that our model outperforms other cutting-edge models, achieving the lowest RMSE value and the highest SSIM value.

References

[1] T. M. Buzug. Computed tomography. In Springer Handbook of Medical Technology, pages 311–342. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[2] H. Zhang, L. Zhang, Y. Sun, and J. Zhang. Low dose CT image statistical reconstruction algorithms based on discrete shearlet. Multimedia Tools and Applications, 76:15049–15064, 2017.

[3] A. H. Behzadi, Z. Farooq, J. H. Newhouse, and M. R. Prince. MRI and CT contrast media extravasation: a systematic review. Medicine, 97(9):e0055, 2018.

[4] D. L. Donoho. Compressed sensing. IEEE Transactions on Information Theory, 52(4):1289–1306, 2006.

[5] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang. Low-dose X-ray CT reconstruction via dictionary learning. IEEE Transactions on Medical Imaging, 31(9):1682–1697, 2012.

[6] J. F. Cai, X. Jia, H. Gao, S. B. Jiang, Z. Shen, and H. Zhao. Cine cone beam CT reconstruction using low-rank matrix factorization: algorithm and a proof-of-principle study. IEEE Transactions on Medical Imaging, 33(8):1581–1591, 2014.

[7] Y. Chen, D. Gao, C. Nie, L. Luo, W. Chen, X. Yin, and Y. Lin. Bayesian statistical reconstruction for low-dose X-ray computed tomography using an adaptive-weighting nonlocal prior. Computerized Medical Imaging and Graphics, 33(7):495–500, 2009.

[8] J. Ma, H. Zhang, Y. Gao, J. Huang, Z. Liang, Q. Feng, and W. Chen. Iterative image reconstruction for cerebral perfusion CT using a pre-contrast scan induced edge-preserving prior. Physics in Medicine and Biology, 57(22):7519, 2012.

[9] Y. Zhang, Y. Xi, Q. Yang, W. Cong, J. Zhou, and G. Wang. Spectral CT reconstruction with image sparsity and spectral mean. IEEE Transactions on Computational Imaging, 2(4):510–523, 2016.

[10] E. Y. Sidky and X. Pan. Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization. Physics in Medicine and Biology, 53(17):4777, 2008.

[11] Y. Zhang, W. Zhang, Y. Lei, and J. Zhou. Few-view image reconstruction with fractional-order total variation. Journal of the Optical Society of America A, 31(5):981–995, 2014.

[12] Y. Zhang, Y. Wang, W. Zhang, F. Lin, Y. Pu, and J. Zhou. Statistical iterative reconstruction using adaptive fractional order regularization. Biomedical Optics Express, 7(3):1015–1029, 2016.

[13] Y. Zhang, W. H. Zhang, H. Chen, M. L. Yang, T. Y. Li, and J. L. Zhou. Few-view image reconstruction combining total variation and a high-order norm. International Journal of Imaging Systems and Technology, 23(3):249–255, 2013.

[14] P. Kaur, G. Singh, and P. Kaur. A review of denoising medical images using machine learning approaches. Current Medical Imaging Reviews, 14(5):675–685, 2018.

[15] H. Chen, Y. Zhang, W. Zhang, P. Liao, K. Li, J. Zhou, and G. Wang. Low-dose CT via convolutional neural network. Biomedical Optics Express, 8(2):679–694, 2017.

[16] C. Tan, M. Yang, Z. You, H. Chen, and Y. Zhang. A selective kernel-based cycle-consistent generative adversarial network for unpaired low-dose CT denoising. Precision Clinical Medicine, 5(2):pbac011, 2022.

[17] H. Chen, Y. Zhang, M. K. Kalra, F. Lin, Y. Chen, P. Liao, et al. Low-dose CT with a residual encoder-decoder convolutional neural network. IEEE Transactions on Medical Imaging, 36(12):2524–2535, 2017.

[18] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[19] C. You, Q. Yang, H. Shan, L. Gjesteby, G. Li, S. Ju, et al. Structurally-sensitive multi-scale deep neural network for low-dose CT denoising. IEEE Access, 6:41839–41855, 2018.

[20] Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, et al. Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. IEEE Transactions on Medical Imaging, 37(6):1348–1357, 2018.

[21] G. Wang and X. Hu. Low-dose CT denoising using a progressive Wasserstein generative adversarial network. Computers in Biology and Medicine, 135:104625, 2021.

[22] F. Fan, H. Shan, M. K. Kalra, R. Singh, G. Qian, M. Getzin, et al. Quadratic autoencoder (Q-AE) for low-dose CT denoising. IEEE Transactions on Medical Imaging, 39(6):2035–2050, 2019.

[23] C. Corti, M. Cobanaj, E. C. Dee, C. Criscitiello, S. M. Tolaney, L. A. Celi, and G. Curigliano. Artificial intelligence in cancer research and precision medicine: Applications, limitations and priorities to drive transformation in the delivery of equitable and unbiased care. Cancer Treatment Reviews, 112:102498, 2023.

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al. Attention is all you need. Advances in Neural Information Processing Systems, 30, 2017.

[26] D. Wang, F. Fan, Z. Wu, R. Liu, F. Wang, and H. Yu. CTformer: convolution-free Token2Token dilated vision transformer for low-dose CT denoising. Physics in Medicine and Biology, 68(6):065012, 2023.

[27] A. Luthra, H. Sulakhe, T. Mittal, A. Iyer, and S. Yadav. Eformer: Edge enhancement based transformer for medical image denoising. arXiv preprint arXiv:2109.08044, 2021.

[28] M. Jian, X. Yu, H. Zhang, and C. Yang. SwinCT: feature enhancement based low-dose CT images denoising with swin transformer. Multimedia Systems, 30(1):1, 2024.

[29] L. Jia, X. He, A. Huang, B. Jia, and X. Wang. Highly efficient encoder-decoder network based on multi-scale edge enhancement and dilated convolution for LDCT image denoising. Signal, Image and Video Processing, 18(8):6081-6091, 2024.

[30] H. Yan, C. Fang, and Z. Qiao. A multi-attention Uformer for low-dose CT image denoising. Signal, Image and Video Processing, 18(2):1429-1442, 2024.

[31] R. S. Jebur, M. H. B. M. Zabil, D. A. Hammood, and L. K. Cheng. A comprehensive review of image denoising in deep learning. Multimedia Tools and Applications, 83(20):58181-58199, 2024.

[32] J. Ruan and S. Xiang. Vm-unet: Vision mamba unet for medical image segmentation. arXiv preprint arXiv:2402.02491, 2024.

[33] Ş. Öztürk, O. C. Duran, and T. Çukur. DenoMamba: A fused state-space model for low-dose CT denoising. arXiv preprint arXiv:2409.13094, 2024.

[34] Y. Li, M. Yang, T. Bian, and H. Wu. Enhancing low-dose CT images by 4x using CACTSR: a deep learning model. In International Conference on Cloud Computing, Performance Computing, and Deep Learning (CCPCDL 2024), volume 13281, pages 295-301. SPIE, September 2024.

[35] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III, volume 18, pages 234-241. Springer, 2015.

[36] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, et al. Vmamba: Visual state space model. Advances in Neural Information Processing Systems, 37:103031-103063, 2024.

[37] J. Xie, L. Xu, and E. Chen. Image denoising and inpainting with deep neural networks. Advances in Neural Information Processing Systems, 25, 2012.

[38] V. Jain and S. Seung. Natural image denoising with convolutional networks. Advances in Neural Information Processing Systems, 21, 2008.

[39] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2):295-307, 2015.

[40] N. Shazeer. Glu variants improve transformer. arXiv preprint arXiv:2002.05202, 2020.

[41] S. Chen, C. Ge, Z. Tong, J. Wang, Y. Song, J. Wang, and P. Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. Advances in Neural Information Processing Systems, 35:16664-16678, 2022.

[42] A. F. Agarap. Deep learning using rectified linear units (ReLU). arXiv preprint arXiv:1803.08375, 2018.

[43] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), pages 3–19, 2018.

[44] Y. Liu, Z. Shao, and N. Hoffmann. Global attention mechanism: Retain information to enhance channel-spatial interactions. arXiv preprint arXiv:2112.05561, 2021.

[45] C. H. McCollough, A. C. Bartley, R. E. Carter, B. Chen, T. A. Drees, P. Edwards, ... and J. G. Fletcher. Low-dose CT for the detection and classification of metastatic liver lesions: results of the 2016 low dose CT grand challenge. Medical Physics, 44(10):e339-e352, 2017.

[46] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, ... and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems, 32, 2019.

[47] H. Shan, A. Padole, F. Homayounieh, U. Kruger, R. D. Khera, C. Nitiwarangkul, ... and G. Wang. Competitive performance of a modularized deep neural network compared to commercial algorithms for low-dose CT image reconstruction. Nature Machine Intelligence, 1(6):269-276, 2019.

[48] C. Tian, Y. Xu, Z. Li, W. Zuo, L. Fei, and H. Liu. Attention-guided CNN for image denoising. Neural Networks, 124:117-129, 2020.

[49] Z. Li, L. Yu, J. D. Trzasko, D. S. Lake, D. J. Blezek, J. G. Fletcher, et al. Adaptive nonlocal means filtering based on local noise level for CT denoising. Medical Physics, 41(1):011908, 2014.

[50] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing, 54(11):4311-4322, 2006.

[51] Y. Chen, X. Yin, L. Shi, H. Shu, L. Luo, J. L. Coatrieux, and C. Toumoulin. Improving abdomen tumor low-dose CT images using a fast dictionary learning based processing. Physics in Medicine and Biology, 58(16):5803, 2013.

[52] P. F. Feruglio, C. Vinegoni, J. Gros, A. Sbarbati, and R. Weissleder. Block matching 3D random noise filtering for absorption optical projection tomography. Physics in Medicine and Biology, 55(18):5401, 2010.

Transfer Learning for Multi-Script Identification: A Comparative Study

Faycel Abbas[1,2], Yahia Menassel[2], Abdeljalil Gattal[2], and Hadil Hattabi [1]

[1] ***LIMOSE Laboratory ,University M'Hamed Bougara,Boumerdès, Algeria ,***
fa.abbas@univ-boumerdes.dz , hattabihadil.dz@gmail.com
[2] ***Laboratoire de Vision et d'Intelligence Artificielle (LAVIA), Université Echahid Cheikh Larbi Tebessi, Tébessa, Algeria ,*** *{yahia.menassel, abdeljalil.gattal}@univ-tebessa.dz ,*

Abstract

Script identification is a critical step in document analysis and optical character recognition (OCR). This study evaluates the performance of transfer learning models for script identification in both handwritten and printed word images. We compare state-of-the-art pretrained models, including ResNet, EfficientNet, and VGG, on the imbalanced ICDAR 2021 Script Identification in the Wild (SIW 2021) dataset. Our results demonstrate that transfer learning models achieve high classification accuracy, balanced accuracy, and ROC AUC scores, particularly when fine-tuned on mixed handwritten and printed data. Data augmentation and external data further enhance performance, highlighting the potential of transfer learning for real-world applications. All source code and dataset links are publicly available.

Keywords: Script identification, Transfer learning, Pretrained models, Imbalanced dataset, Fine-tuning.

1 Introduction

Script identification is a crucial step in document analysis and optical character recognition (OCR) systems, particularly in multilingual and multi-script environments. It involves determining the script of a given text image, which is essential for subsequent processing steps such as text recognition and translation. With the increasing volume of multimedia data, including handwritten and printed documents, the need for robust script identification methods has grown significantly. This task is particularly challenging due to variations in text appearance, image quality, diverse text styles, complex backgrounds, and subtle script differences. For example, scripts like Arabic and Persian share similar characters, making it difficult to distinguish between them.

Traditional methods for script identification rely on handcrafted features such as texture, edges, and contours [4]. However, these methods often struggle with the complexity and variability of real-world data. In recent years, deep learning models, particularly Convolutional Neural Networks (CNNs), have revolutionized the field by automating feature extraction and enabling the fusion of multimodal features such as visual, structural, and linguistic cues [5]. Transfer learning models, such as ResNet and EfficientNet, have been widely used for script identification, leveraging pretrained weights from large-scale datasets like ImageNet to achieve high accuracy [2].

This paper focuses on evaluating the performance of transfer learning models for script identification. We compare state-of-the-art pretrained models, including ResNet, EfficientNet, and VGG, on the imbalanced ICDAR 2021 Script Identification in the Wild (SIW 2021) dataset [1]. Our results demonstrate that transfer learning models achieve high classification accuracy, balanced accuracy, and ROC AUC scores, particularly when fine-tuned on mixed handwritten and printed data. Data augmentation and external data further enhance performance, highlighting the potential of transfer learning for real-world applications.

2 Dataset Description

The **ICDAR 2021 Script Identification in the Wild (SIW 2021) dataset** [1] is one of the largest publicly available datasets for script identification, containing 13 scripts: Arabic, Bengali, Gujarati, Gurmukhi, Devanagari, Japanese, Kannada, Malayalam, Oriya, Roman, Tamil, Telugu, and Thai. The dataset includes both handwritten and printed word images, making it highly diverse and representative of real-world scenarios.

2.1 Dataset Composition

- **Total Images**: 86,675
  - **Training Set**: 60,643 images (70% of the dataset)
    - **Printed Images**: 21,974
    - **Handwritten Images**: 8,887
  - **Testing Set**: 26,012 images (30% of the dataset)
    - **Printed Images**: 27,070
    - **Handwritten Images**: 28,744

2.2 Script Distribution

The dataset is imbalanced, with some scripts having significantly more samples than others. For example, the **Roman** script has the highest number of samples (6,053 printed and 3,750 handwritten), while the **Gujarati** script has fewer samples (982 printed and 37 handwritten). This imbalance poses a challenge for model training and evaluation, as it requires robust techniques to handle underrepresented scripts.

2.3 Challenges

- **Class Imbalance**: The uneven distribution of scripts in the dataset can lead to biased models that perform well on majority classes but poorly on minority classes.
- **Variability in Handwritten Scripts**: Handwritten text introduces additional challenges due to variations in writing styles, stroke thickness, and character shapes.
- **Complex Backgrounds**: Some images have complex backgrounds, making it difficult to isolate and identify the script.

3 Methodology

The methodology for script identification involves a systematic approach, combining preprocessing, fine-tuning of transfer learning models, and comparative evaluation. The goal is to develop a robust and efficient model capable of accurately identifying scripts from input images.

3.1 Preprocessing

The first step in the methodology is preprocessing the input images. All images are resized to a uniform size of 128x128 pixels to ensure consistency in input dimensions. Additionally, pixel values are normalized to a range of [0, 1] by scaling them from their original range of 0-255. This normalization step is crucial for improving training stability and facilitating faster convergence during optimization.

3.2 Transfer Learning Models

We evaluate several state-of-the-art transfer learning models, including **ResNet-50**, **EfficientNet-B0**, **VGG-16**, **GoogleNet**, and **AlexNet**. These models are pretrained on the **ImageNet** dataset and fine-tuned on the **SIW 2021 dataset** to adapt them to the script identification task. This approach leverages the feature extraction capabilities of these well-established architectures while tailoring them to the specific dataset.

3.2.1 ResNet-50

ResNet-50 [2] is a deep residual network with 50 layers, known for its skip connections that help mitigate the vanishing gradient problem. The skip connections allow the network to learn residual functions, making it easier to train very deep networks. ResNet-50 has been widely used in various computer vision tasks due to its ability to extract high-level features effectively. In this study, ResNet-50 is fine-tuned on the SIW 2021 dataset, achieving an accuracy of 98.20%.

Table 1: Architecture of ResNet-50

Layer Type	**Output Shape**	**Parameters**
Input Layer	(128, 128, 3)	0
Conv2D	(64, 64, 64)	9,408
BatchNormalization	(64, 64, 64)	256
MaxPooling2D	(32, 32, 64)	0
Residual Block 1	(32, 32, 256)	215,296
Residual Block 2	(16, 16, 512)	1,187,840
Residual Block 3	(8, 8, 1024)	7,077,888
Residual Block 4	(4, 4, 2048)	14,942,208
GlobalAveragePooling	(2048)	0
Dense	(13)	26,637
Total Parameters	**25.6 Million**	

3.2.2 EfficientNet-B0

EfficientNet-B0 [8] is a lightweight and efficient model that uses compound scaling to balance depth, width, and resolution. The compound scaling method ensures that the model scales up uniformly across all dimensions, resulting in a highly efficient and scalable architecture. EfficientNet-B0 achieves the highest accuracy (98.60%) and ROC-AUC (99.60%) on the SIW 2021 dataset, making it the best-performing model in this study. Its lightweight architecture, with only 5.3 million parameters, makes it suitable for real-world applications where computational resources are limited.

Table 2: Architecture of EfficientNet-B0

Layer Type	**Output Shape**	**Parameters**
Input Layer	(128, 128, 3)	0
Conv2D	(64, 64, 32)	864
BatchNormalization	(64, 64, 32)	128
Conv2D	(32, 32, 16)	4,608
BatchNormalization	(32, 32, 16)	64
MaxPooling2D	(16, 16, 16)	0
Conv2D	(8, 8, 32)	4,640
BatchNormalization	(8, 8, 32)	128
Conv2D	(4, 4, 64)	18,496
BatchNormalization	(4, 4, 64)	256
GlobalAveragePooling	(64)	0
Dense	(13)	845
Total Parameters	**5.3 Million**	

3.2.3 VGG-16

VGG-16 [6] is a deep convolutional network with 16 layers, known for its simplicity and effectiveness in feature extraction. The model consists of multiple convolutional layers followed by max-pooling layers, which reduce the spatial dimensions of the feature maps. VGG-16 has been widely used in various image classification tasks due to its ability to capture intricate patterns in images. In this study, VGG-16 achieves an accuracy of 97.85% on the SIW 2021 dataset.

3.2.4 GoogleNet

GoogleNet [7] is a 22-layer deep network that uses inception modules to reduce computational cost. The inception modules allow the network to capture features at multiple scales, making it highly effective for complex image classification tasks. In this study, GoogleNet achieves an accuracy of 96.50% on the SIW 2021 dataset.

Table 3: Architecture of VGG-16

Layer Type	**Output Shape**	**Parameters**
Input Layer	(128, 128, 3)	0
Conv2D	(128, 128, 64)	1,792
BatchNormalization	(128, 128, 64)	256
MaxPooling2D	(64, 64, 64)	0
Conv2D	(64, 64, 128)	73,856
BatchNormalization	(64, 64, 128)	512
MaxPooling2D	(32, 32, 128)	0
Conv2D	(32, 32, 256)	295,168
BatchNormalization	(32, 32, 256)	1,024
MaxPooling2D	(16, 16, 256)	0
Conv2D	(16, 16, 512)	1,180,160
BatchNormalization	(16, 16, 512)	2,048
MaxPooling2D	(8, 8, 512)	0
Conv2D	(8, 8, 512)	2,359,808
BatchNormalization	(8, 8, 512)	2,048
MaxPooling2D	(4, 4, 512)	0
Flatten	(8192)	0
Dense	(4096)	33,558,528
Dense	(4096)	16,781,312
Dense	(13)	53,261
Total Parameters	**138 Million**	

Table 4: Architecture of GoogleNet

Layer Type	**Output Shape**	**Parameters**
Input Layer	(128, 128, 3)	0
Conv2D	(64, 64, 64)	9,408
BatchNormalization	(64, 64, 64)	256
MaxPooling2D	(32, 32, 64)	0
Inception Module 1	(32, 32, 256)	163,840
Inception Module 2	(16, 16, 480)	580,608
Inception Module 3	(8, 8, 512)	1,024,000
Inception Module 4	(4, 4, 512)	1,048,576
GlobalAveragePooling	(512)	0
Dense	(13)	6,669
Total Parameters	**7 Million**	

3.2.5 AlexNet

AlexNet [3] is one of the earliest deep learning models, with 8 layers. Despite its relatively shallow architecture, AlexNet has been widely used in various image classification tasks. In this study, AlexNet achieves an accuracy of 95.12% on the SIW 2021 dataset, making it the weakest-performing model among the transfer learning models evaluated.

3.3 Training Configuration

All models are trained using the **Adam optimizer** with a learning rate of 0.001, a batch size of 32, and 70 epochs. To enhance generalization and mitigate overfitting, data augmentation techniques such as rotation, shear, and zoom are applied during training. These techniques increase the diversity of the training data, enabling the models to learn more robust and invariant features.

3.4 Performance Evaluation

The performance of the models is evaluated using a comprehensive set of metrics, including **Correct Classification Accuracy (CCA)**, **F1 score**, **Balanced Accuracy (BA)**, and **ROC AUC score**.

Table 5: Architecture of AlexNet

Layer Type	**Output Shape**	**Parameters**
Input Layer	(128, 128, 3)	0
Conv2D	(64, 64, 96)	34,944
BatchNormalization	(64, 64, 96)	384
MaxPooling2D	(32, 32, 96)	0
Conv2D	(32, 32, 256)	614,656
BatchNormalization	(32, 32, 256)	1,024
MaxPooling2D	(16, 16, 256)	0
Conv2D	(16, 16, 384)	885,120
BatchNormalization	(16, 16, 384)	1,536
Conv2D	(16, 16, 384)	1,327,104
BatchNormalization	(16, 16, 384)	1,536
Conv2D	(16, 16, 256)	884,992
BatchNormalization	(16, 16, 256)	1,024
MaxPooling2D	(8, 8, 256)	0
Flatten	(16384)	0
Dense	(4096)	67,092,992
Dense	(4096)	16,781,312
Dense	(13)	53,261
Total Parameters	**61 Million**	

These metrics are particularly well-suited for assessing performance on imbalanced datasets, as they account for both precision and recall, ensuring a more holistic evaluation of the models' predictive capabilities.

4 Results and Discussion

The SIW 2021 dataset, containing 13 scripts (e.g., Arabic, Bengali, Gujarati), is used for evaluation. The dataset is divided into training and testing sets with a ratio of 70:30 (60,643 training images and 26,012 testing images). This division ensures a robust evaluation of the model's generalization capabilities.

The results demonstrate that transfer learning models achieve high accuracy across three tasks: mixed scripts, printed scripts, and handwritten scripts. The performance metrics for each task are summarized in Table 6.

Table 6: Performance Metrics for Transfer Learning Models (Input Size: 128x128 Pixels)

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**ROC-AUC (%)**
EfficientNet-B0	98.60	98.58	98.60	98.59	99.60
ResNet50	98.20	98.18	98.20	98.19	99.40
VGGNet19	97.90	97.88	97.90	97.89	99.25
VGGNet16	97.85	97.83	97.85	97.84	99.20
GoogleNet	96.50	96.48	96.50	96.49	98.80
AlexNet	95.12	95.10	95.12	95.11	98.50

4.1 Analysis of Results

The results highlight the effectiveness of transfer learning models in handling diverse script identification tasks. **EfficientNet-B0** achieves the highest accuracy (98.60%) and ROC-AUC (99.60%), followed closely by **ResNet50** (98.20%) and **VGGNet19** (97.90%). These models benefit from their deep architectures and pretrained weights, which enable them to extract complex features effectively.

- **EfficientNet-B0** stands out as the most efficient model, with only 5.3 million parameters and a training time of 70 seconds per epoch. Its lightweight architecture makes it suitable for real-world applications where computational resources are limited.

- **ResNet50** and **VGGNet19** also perform well but require significantly more parameters and longer training times, making them less efficient for large-scale deployments.
- **AlexNet**, being one of the earlier deep learning models, performs the weakest, with an accuracy of 95.12%, likely due to its relatively shallow architecture compared to more modern models.

4.2 Limitations and Future Work

While transfer learning models demonstrate strong performance, they have certain limitations. For instance, they require significant computational resources for fine-tuning, especially on large datasets. Additionally, their performance may degrade when applied to scripts or languages not well-represented in the pretraining dataset (e.g., ImageNet). Future work will focus on addressing these limitations by exploring hybrid models that combine the strengths of transfer learning and custom architectures. We also plan to investigate the use of unsupervised or semi-supervised learning techniques to reduce the reliance on labeled data.

5 Conclusion

Transfer learning models demonstrate robust performance in script identification for both handwritten and printed word images. These models effectively handle class imbalance and script variations, achieving high accuracy and generalizability. Data augmentation and external data significantly enhance performance, making transfer learning a promising solution for real-world applications. Future work will focus on further optimizing these models and exploring their applicability to other document analysis tasks.

References

[1] A. Das et al. Icdar 2021 competition on script identification in the wild. In *Document Analysis and Recognition – ICDAR 2021*, pages 738–753, 2021.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 25, pages 1097–1105, 2012.

[4] P. B. Pati and A. G. Ramakrishnan. Word level multi-script identification. *Pattern Recognition Letters*, 29(9):1218–1229, 2008.

[5] B. Shi, X. Bai, and C. Yao. Script identification in the wild via discriminative convolutional neural network. *Pattern Recognition*, 52:448–458, 2016.

[6] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[7] C. Szegedy et al. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[8] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 6105–6114, 2019.

Parameter-Efficient Fine-Tuning for LLM-Based Arabic-to-English Machine Translation

Moudjar Amina[1], Bahloul Belahcene[1], and Aliane Hassina[2]

[1] ***Djilali Bounaama University,*** *amina-moudjar@univ-dbkm.dz d.bahloul@univ-dbkm.dz*
[2] ***CERIST,*** *ahassina4@gmail.com*

Abstract

Large Language Models (LLMs) such as GPT-3, BLOOM, BERT... have revolutionized natural language processing (NLP), particularly in translation. However, fine-tuning these models for downstream tasks, such as Arabic-to-English translation, requires extensive computational resources. Traditional full fine-tuning methods that involve updating all parameters of the model pose significant computational and memory challenges, notably for models with billions of parameters. This study investigates the application of LoRA-based PEFT methods on two chosen models for Arabic to English translation, AraT5 and NLLB-200, with a focus on understanding the trade-offs between computational efficiency and translation quality.

Keywords: Large Language models, Efficiency, Computational challenges, Parameter-efficient fine-tuning, Arabic-to-English Machine translation.

1 Introduction

The advent of LLMs has revolutionized the field of natural language processing, they have large numbers of parameters and complex architectures to capture intricate language patterns, making them highly effective in translation tasks that need nuanced understanding and generation. Yet, fine-tuning LLMs for specific tasks requires enormous computational resources. Parameter-Efficient Fine-Tuning (PEFT) techniques have emerged to mitigate these challenges by selectively adjusting a small subset of parameters. Among PEFT methods, Low-Rank Adaptation (LoRA) and its variants stand out for their effectiveness, by introducing low-rank matrices to specific layers, allowing the model to learn task-specific adaptations efficiently. Quantized methods go further by quantizing the parameters involved in fine-tuning. Our study focuses on LLMs with encoder-decoder architecture: AraT5 and NLLB-200, which are ideal for Arabic-to-English translation. By applying the previous techniques, we aim to capture the trade-offs between computational efficiency and model performance in machine translation.
Our key contributions are:

1. We apply LoRA, DoRA, QLoRA, and QDoRA techniques to fine-tune the AraT5 and NLLB-200 models for Arabic-to-English translation.
2. We assess the computational efficiency of these methods, comparing them to traditional full fine tuning.
3. We evaluate translation quality across the different f ine-tuning methods, examining the trade-offs be tween efficiency and performance.

This paper is organized as follows: The first section reviews LLM-based machine translation approaches, the second section details parameter-efficient fine-tuning techniques, the third section presents our methodology, chosen datasets and models, and the fourth section compares resource usage and translation quality across methods and models.

2 Related works

Machine Translation has undergone significant transformation recently, primarily due to the rapid advancements in LLMs. These advancements have pushed research into LLM-based machine translation, focusing on two main paradigms: In-Context Learning (ICL) and Finetuning.
ICL leverages optimal in-context examples [2] [36] [18] , dictionary knowledge [13] [25], adaptive learning

[35] [27], and translation memories [34] to enhance translation accuracy. Traditional machine translation models, particularly those using statistical methods, struggle with contextually rich languages like Arabic, which demand effective capture of long-range dependencies and contextual nuances.Recent advancements in context-aware neural machine translation models, particularly those using self-attention mechanisms, have shown better performance by dynamically focusing on different parts of the input sentence and its context. These models benefit from incorporating larger context windows and external contextual information, such as linguistic annotations and discourse relations, significantly improving the translation quality.
Concurrently, finetuning has been instrumental in augmenting LLM's capability to translate unseen languages and domains [42] [26] and in building multilingual models [44] [46]. Additionally, research has delved into post-editing translation outcomes [28] [33] and utilizing LLMs for machine translation evaluation [12] [11]. The finetuning process for Arabic to English involves adapting pre-trained LLMs, such as T5, to the specific requirements of the translation task. This process includes further training on parallel Arabic-English datasets, which helps the model capture the syntactic, semantic, and contextual intricacies of both languages. Techniques like domain adaptation, advanced regularization methods, back-translation, and self-training enhance the performance and robustness of the finetuned models. The quality and size of the parallel corpus are crucial for achieving high translation accuracy, highlighting the importance of high-quality, diverse, and representative sentence pairs in the training data.

Recent advancements in multilingual machine translation have shown significant improvements in performance across various language families, including Afro-Asiatic languages and specific language pairs such as Arabic to English. Zhu et al. (2023) [45] evaluated several LLMs on the FLORES-101 dataset [14] using the SentencePiece BLEU (spBLEU) metric (both SentencePiece BLEU and sacreBLEU are libraries used for calculating BLEU scores). For Afro-Asiatic languages (which include Arabic), LLaMA2-7B [39] achieved the highest BLEU score of 57.72, followed by XGLM-7.5B [22] at 54.51 and Falcon-7B [3] at 38.62. Focusing on the Arabic-English pair, GPT-4 performed better than ChatGPT, with scores of approximately 45 and 40 respectively. These results come from remarkably large models, which explains the high BLEU score results.

3 Parameter-efficient Fine-tuning (PEFT)

PEFT adapts an LLM to downstream tasks by freezing the entire LLM backbone and updating only a small set of newly introduced parameters. PEFT methods can be classified into four categories: low-rank adaptation (LoRA [17]), adapter-based tuning (inserting trainable modules into LLMs to simplify fine-tuning [16]), prefix tuning (adding trainable vectors to each LLM layer that are adapted to specific tasks [21]), and prompt tuning (adjusting only the input layer by incorporating trainable prompt tokens that can be placed at the beginning or within the input text [20]).
PEFT methods necessitate careful consideration of several factors to balance performance and efficiency effectively. A major challenge lies in minimizing trainable parameters while maintaining Substantial performance [30]. Fine-tuning too few parameters can restrict the model's adaptability to the target task, while excessive fine-tuning can degrade the computational advantages of PEFT [9] [24]. The success of PEFT also relies on the quality and quantity of data available, particularly in domains with limited or noisy data where achieving the same accuracy as full fine-tuning can be tough. In such cases, careful selection of data augmentation techniques and transfer learning strategies is important [8] [4].

3.1 Low Rank Adaptation (LoRA)

Low-Rank Adaptation [17], proposed by Hu et al. (2021), is a widely used Parameter-Efficient Fine-Tuning approach aimed to optimize the adaptation of LLMs to specific tasks. During full fine-tuning, the model is initialized to pre-trained weights W_0 and updated to $W_0 + \Delta W$ The basic hypothesis behind LoRA is that during fine-tuning, A low-rank approximation can powerfully capture the necessary adjustments to the model's weights. This means that the changes in the weight matrix resulting from task-specific adaptation have a low "intrinsic rank" [1]. As shown in Figure 3 the information contained within the ΔW matrix can be represented using fewer dimensions than the original matrix and therefore, the full-dimensional update can be approximated by a product of two smaller matrices while keeping the original weights frozen.

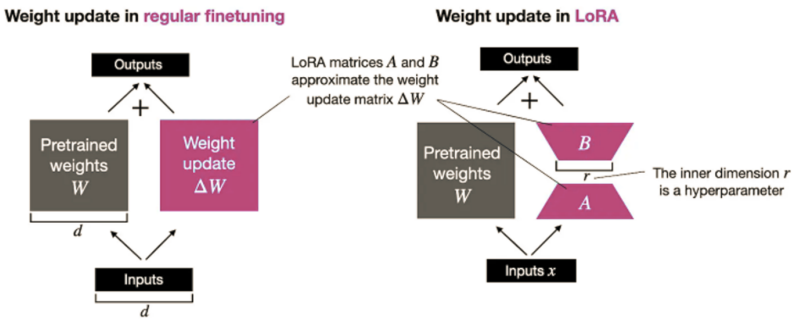


Figure 1: Comparison between the traditional fine-tuning approach and the LoRA method

3.1.1 Mathematical Formulation

Considering a pre-trained weight matrix W of a neural network. During fine-tuning, instead of updating W directly, LoRA introduces two trainable low-rank matrices $A \in R^{m \times r}$ and $B \in R^{r \times n}$. The weight update is then formulated as :

$$W_{updated} = W + \Delta W = W + AB \tag{1}$$

Here, r is the rank, a hyperparameter that defines the dimensionality of the low-rank approximation. Using this approach, the amount of trainable parameters is remarkably decreased, as only the matrices. A and B need to be learned, while the original weight matrix W remains frozen. This makes the fine-tuning process much more memory and computation efficient.

3.1.2 Reparametrization and Optimization

LoRA modifies the forward pass of the neural network by adding the low-rank update $\Delta W = AB$ to the original output. Specifically, if the original output is $h = W_0 x$, the updated output becomes:

$$W_{updated} = W_0 x + \Delta W x = W_0 x + ABx \tag{2}$$

In practice, during backpropagation, the frozen pre-trained weights W_0 remain untouched, and the loss is only used to update the B and A matrices introduced by LoRA. A is initialized with a random Gaussian distribution, while B is initialized to zero, ensuring that the initial value of ΔW is zero. The scaling factor α is introduced to balance the contribution of ΔW during training, which is crucial for controlling the impact of the low-rank updates and making sure that the fine-tuning process remains stable and effective.

3.1.3 Applying LoRA to a Transformer

In a transformer architecture of an LLM, it is more common to apply LoRA to the attention layers because they are computationally expensive and have a significant number of parameters, which makes them the prime targets for parameter-efficient fine-tuning. However, it's not limited to just attention layers; it could also be applied to other layers like feed-forward networks.

LoRA allows the fine-tuning process to require fewer parameters and less computational power while still achieving strong performance. However, as with many advancements in AI, researchers have recently introduced other innovative alternatives and derivatives of LoRA, such as DoRA [23], LoRA+ [15], QA-LoRA [41], QLoRA [6], QDoRA [23], and DyLoRA [40], depending on the model architecture itself and the area of focus.

3.2 Weight-Decomposed Low-Rank Adaptation (DoRA)

Weight-Decomposed Low-Rank Adaptation [23] was introduced and built on LoRA by introducing a decomposition of the weight matrix into two components to fine tune them: magnitude and direction, as illustrated in Figure 2. This decomposition separates the fine-tuning of these components, addressing issues in LoRA to make subtle adjustments to weight directions while efficiently handling parameter updates. The process behind DoRA is divided into two main steps. First, the weight matrix W_0 from a pretrained model is decomposed into two components:

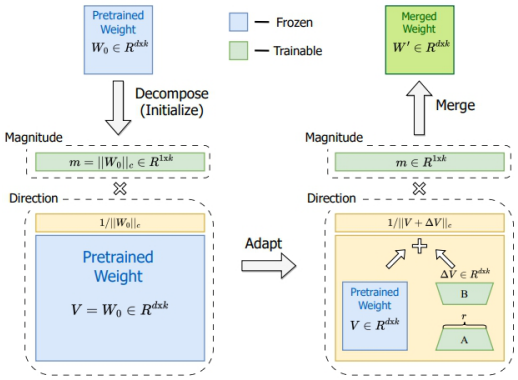


Figure 2: An overview of DoRA [23].

- Magnitude Vector m: This vector represents the norm or length of each column in the weight matrix, capturing the scale information.
- Directional Matrix V: Where each column vector of the weight matrix is normalized by dividing by its magnitude, keeping only the directional information.

Once the pretrained weights are decomposed, and given the substantial size of the directional component in terms of parameters, LoRA is applied exclusively to the directional matrix V, and m is trained as it is, which is feasible because it has just one dimension.
DoRA improves both the learning capacity and stability of LoRA, without causing any additional inference overhead. It enables fine-tuning a pretrained model in a way that is computationally efficient and potentially more responsive to new data, maintaining the strengths of the original model while adapting it to new tasks or datasets.

3.3 Quantized Low rank adaptation (QLoRA)

QLoRA reduces the memory footprint of LLMs by compressing weights from high-precision data types such as 32-bit floating point to lower-precision formats such as 4-bit integers or NormalFloat, while integrating trainable Low-Rank Adapters (LoRA) for fine-tuning. The pretrained weights remain frozen, and only the LoRA parameters are updated, minimizing memory requirements. During computational tasks, weights stored as 4-bit NormalFloat are dequantized to 16-bit BrainFloat (bfloat16) for both the forward and backward passes. However, only the LoRA parameter's gradients are computed. QLoRA achieves high-fidelity 4-bit fine tuning via two proposed techniques 4-bit NormalFloat (NF4) Quantization and Double Quantization. Paged Optimizers were also introduced to prevent memory spikes during gradient checkpointing from causing out-of-memory errors that have traditionally made fine tuning on a single machine difficult for large models.

3.3.1 4-bit NormalFloat Quantization

The NormalFloat (NF) data type builds on Quantile Quantization [5] which is an information-theoretically optimal data type that ensures each quantization bin has an equal number of values assigned from the input tensor. Quantile quantization works by estimating the quantile of the input tensor through the empirical cumulative distribution function. This technique helps compress model weights effectively, but the process of quantile estimation is computationally expensive. Fast quantile approximation algorithms, such as SRAM quantiles, help mitigate this cost, but approximation errors arise, especially for outliers, which are often critical.
QLoRA addresses these issues by recognizing that pre-trained LLM weights typically follow a zero-centered normal distribution with a standard deviation α . By transforming all weights to fit within a fixed range (e.g., $[-1, 1]$), accurate quantile estimation becomes feasible, eliminating the need for computationally expensive approximation algorithms. The result is the 4-bit NormalFloat (NF4) data type, which is specifically optimized for normally distributed data. It normalizes neural network weights into this fixed range and quantizes them accordingly, enabling precise weight compression with minimal performance loss.

3.3.2 Double Quantization

A method that reduces the average memory footprint by quantizing the quantization constants [7]. This saves approximately 0.37 bits per parameter, which translates to around 3 GB for a 65B model. It further reduces the memory overhead of quantization constants. By quantizing both the model weights and the quantization constants, QLoRA achieves higher memory efficiency without negatively impacting model performance.

3.3.3 Paged Optimizers

Fine-tuning LLMs can also generate memory spikes, primarily when processing long sequences or large mini-batches. QLoRA harnesses Paged Optimizers to handle these spikes efficiently, leveraging NVIDIA's unified memory system, which automatically transfers memory between the CPU and GPU. When the GPU runs out of memory, data is paged to the CPU and then moved back to the GPU when needed. This seamless paging mechanism guarantees that memory bottlenecks do not interrupt the training process, allowing QLoRA to handle larger models and batch sizes with fewer resources.

3.4 Quantized Weight-Decomposed Low-Rank Adaptation (QDoRA)

QDoRA combines the memory efficiency of QLoRA with the DoRA fine-tuning. QDoRA leverages quantization to compress weights into low-precision formats, significantly reducing memory footprint. However, it goes further by incorporating weight decomposition, as seen in DoRA, to achieve more granular optimization during fine-tuning. This allows QDoRA to maintain both high performance and low computational requirements, making it suitable for fine tuning and training large models like Llama 3 on consumer-grade GPUs.

4 Methodology

In this work, we explore the performance of small-sized LLMs in Arabic-to-English machine translation, focusing specifically on encoder-decoder-based architectures. Our research is based on two publicly available LLMs on HuggingFace: the Arabic-focused AraT5v2 Base and the multilingual NLLB-200 distilled-600M models. To evaluate these models, we conducted experiments using the United Nations Parallel Corpus [47] with AraT5v2 [10] and the OPUS-100 corpus [43] with NLLB-200 [38]. Our objective was to evaluate the trade-offs between computational resource efficiency, such as GPU power consumption and memory allocation, and translation performance using BLEU, and ROUGE and Perplexity scores, without an explicit aim to improve translation quality. We applied four fine-tuning techniques in addition to Full fine-tuning (which served as a baseline used for comparison): LoRA, DoRA, and quantized variants: QLoRA and QDoRA.

4.1 Experiments on AraT5v2 with United Nations Parallel Corpus

We used the United Nations Parallel Corpus with the AraT5v2 model. We adopted the following dataset splitting strategy to ensure a balanced and effective model training, validation, and testing. We started by loading the first 20 000 examples from the corpus. The dataset was split as into:

- Training Set: 15,000 examples (75%) were dedicated to training, ensuring that most of the data is used for model learning.
- Validation Set: 2,500 examples (12.5%) were reserved for validation. This validation set is used to track the model's performance on unseen data, serving it to prevent overfitting.
- Test Set: 2,500 examples (12.5%) were set aside for testing. The test set is used for the final evaluation.

AraT5v2 built on a foundation set by the original AraT5 model [29]. AraT5 was inspired by the T5 (Text-to-Text Transfer Transformer) model [32], which reframes all NLP tasks into a text-to-text format, offering a unified framework for language modeling tasks.

Tokenization is an important step in converting raw text into a format that a model like AraT5v2 can process. AraT5v2 relies on tokenized inputs for both the source language (Arabic) and the target language (English).

- Task-Specific Prefix: The T5 model needs a clear prompt of the task it is performing. In this case, we prepend the Arabic input text with the prefix: "translate Arabic to English: "
- Tokenization Using AraT5v2's Tokenizer: We use Hugging Face's AutoTokenizer to tokenize the Arabic input sentences and their corresponding English translations. AutoTokenizer is a generic tokenizer class in the Huggingface Transformers library that automatically selects the proper tokenizer for a given model. The original T5 models (including AraT5, which is based on T5) typically use SentencePiece tokenizers [37], which is a text tokenization algorithm widely used in modern NLP models, particularly in models that need to handle complex languages with rich morphology (like Arabic). Unlike traditional tokenizers that split text based on spaces or punctuation, SentencePiece treats the entire text as a continuous stream of characters and learns how to break it into subword units.
- Truncation and Padding: To ensure that the input sequences fit within the model's constraints, we limit the maximum sequence length to 128 tokens. Sequences longer than this are truncated, while shorter ones are padded to a uniform length. This ensures consistency across batches during training.

The tokenized dataset consists of pairs of input and output sequences, where each sequence is a list of tokens representing either an Arabic sentence (input) or its English translation (output). These tokenized sequences are then ready for training the AraT5v2 model.
We fine tuned AraT5v2 using different approaches: Full fine-tuning, LoRA, DoRA, and QDoRA:

1. Full fine-tuning means updating all the parameters of the model based on the specific wanted task. We defined several key hyperparameters (used for other techniques as well): Once the training setup

Table 1: Hyperparameters Used for Model Training

Learning rate	$2 \cdot 10^{-4}$
Batch size	2
Number of epochs	5

   is complete, the model is trained on the training set by backpropagating through the entire model, including all attention and feed-forward layers, updating every weight in the network. Training is followed by evaluation on the validation set after each epoch to track its performance.

2. We used LoRA to fine-tune AraT5v2, focusing on the following layers involved in the attention mechanism: the key (k), query (q), value (v), and output (o) layers. After freezing the core model parameters, LoRA introduces learnable low-rank matrices that adjust the key, query, value, and output layers of the attention mechanism. These matrices are updated during training, while the original parameters remain untouched. The low-rank structure allows for efficient fine-tuning with fewer parameters to update.

3. In another experiment we applied DoRA by freezing the original model parameters and decomposing the weight matrices involved in the key (k), query (q), value (v), and output (o) layers of the attention mechanism. By applying DoRA to the Target Modules, we decomposed the weight matrices associated with them into their magnitude and directional components. The fine-tuning process updates the directional matrices using LoRA, while the magnitude vector is trained directly. After training, all the components are recombined to form the updated weight matrices which is used for inference.

4. Lastly, QDoRA is applied to the AraT5v2 model by combining two techniques: quantization and Weight Decomposed Low-Rank Adaptation (DoRA). We applied these techniques to the model by quantizing the model's weights to 4-bit precision, then freezing the core weights of the model (from its pre-trained state), the Decomposition of Weights, LoRA is then applied specifically to the directional matrices within the attention layers chosen, and alongside the updates to the directional matrices, the magnitude vector (which represents the scale of the weights) is also fine-tuned. Since the magnitude has significantly fewer parameters, it can be trained directly without requiring the same low-rank approximations used for the directional matrices.

Table 3 summarizes the previous techniques.

4.2 Experiments on NLLB-200 with OPUS-100 Corpus

The NLLB-200 distilled-600M model was fine-tuned using the OPUS-100 dataset. We splitted the dataset into model training, validation, and testing. We started by loading the first 11200 examples from the corpus. The dataset splitting was done as following:

- Training Set: 8000 examples (71%) were dedicated to training.
- Validation Set: 1600 examples (almost 15%) were reserved for validation.
- Test Set: 1600 examples (almost 15%) were set aside for testing.

The NLLB-200-distilled-600M represents a distilled version of the full NLLB-200 model of 600 million parameters. NLLB-200 is an innovative solution to the complex challenges of multilingual machine translation, especially in low-resource languages.
We use a fast tokenizer NllbTokenizerFast to process the text, specifying Arabic as the source language (src_lang) and English as the target language (tgt_lang). It is based on the BytePairEncoding [31], which preserves common words in their full form, while splitting less frequent words into subword units, achieving a balance between vocabulary size and representational efficiency. The tokenizer also handles truncation, ensuring that sentences are cut off at the model's maximum input length to avoid issues during training. By tokenizing the entire dataset, we prepare it for the subsequent steps.
Once the dataset is prepared, we move on to the fine-tuning stage:

1. Full fine-tuning allows all the layers of the model to be updated during training. The training is configured controlling various aspects (same configuration for other fine tuning methods of the model):

Table 2: Hyperparameters Used for Model Training

Learning rate	$2 \cdot 10^{-5}$
Batch size	1
Number of epochs	3

   Evaluation is performed at the end of each epoch.

2. We applied LoRA where instead of updating all the model's parameters, only the low-rank matrices are updated during fine-tuning, which drastically reduces the number of parameters that need to be trained. In our case, the target layers for LoRA are:Query projection (q_proj), Value projection (V_proj). Only the low-rank matrices in the q_proj and v_proj layers are updated, while the rest of the model remains frozen. This reduces the number of trainable parameters while retaining enough capacity to specialize for the translation task.

3. In another experiment, we applied QLoRA which is an advanced version of the original LoRA technique (same as QDoRA). QLoRA focuses on fine-tuning specific layers of the model; this approach extends the benefits of LoRA by further reducing the model's memory footprint through quantization, while still fine-tuning only a small fraction of the model's parameters. The core difference in QLoRA is the use of 4-bit quantization that reduces memory requirements even further while maintaining precision for computation.

4. In another experiment, DoRA was applied to fine-tune the NLLB-200 distilled 600M model for Arabic to English translation. By introducing a decomposition of the weight matrix into magnitude and direction, DoRA enhances the fine-tuning process. DoRA allows for independent updates to the scale and directional aspects of the model's weights. The directional matrix is fine-tuned using LoRA's low-rank matrices, while the magnitude vector is directly trained.

Table 4 summarizes these techniques.

4.3 Evaluation

To track CPU, GPU and Memory usage, we used Weights & Biases, a platform for machine learning developers to help them build better models faster. It provides lightweight, interoperable tools for

Table 3: AraT5v2 Fine-Tuning Methods

Fine-Tuning Method	Parameters Updated	Low-Rank Dimension (r)	Scaling Factor (lora_alpha)	Dropout (lora_dropout)	Quantization	DoRA Enabled	Full Parameters Updated
Full Fine-Tuning	All Parameters	-	-	-	No	No	Yes
LoRA	Attention Layers (k, q, v, o)	5	32	0.06	No	No	No
DoRA	Magnitude + Direction	5	32	0.06	No	Yes	No
QDoRA	Quantized + Magnitude + Direction	5	32	0.06	Yes	Yes	No

Table 4: NLLB200-600M Fine-Tuning Methods

Fine-Tuning Method	Parameters Updated	Low-Rank Dimension (r)	Scaling Factor (lora_alpha)	Dropout (lora_dropout)	Quantization	DoRA Enabled	Full Parameters Updated
Full Fine-Tuning	All Parameters	-	-	-	No	No	Yes
LoRA	Attention Layers (q, v)	8	32	0.1	No	No	No
DoRA	Magnitude + Direction	8	32	0.1	No	Yes	No
QLoRA	Quantized + Attention Layers (q, v)	8	32	0.1	Yes	No	No

tracking experiments, versioning datasets and models, evaluating model performance, and visualizing results.Additionally to the W&B system tracking, we evaluated the optimization techniques using Perplexity, that measures the model's uncertainty in generating the next token, the lower the value the better. In addition, we also used the following evaluation metrics: ROUGE-L, which measures the overlap of longest common subsequences between the system output and reference texts, indicating the quality of text summarization or other generation tasks. And SacreBLEU score, which is an extended implementation that builds upon the basic BLEU metric and provides additional features, such as multiple reference support, better handling of tokenization, and more fine-grained control over the evaluation process.

5 Results and discussion

5.1 Models efficiency comparison

Using W&B SDK, we tracked system metrics for every technique on both chosen models.

5.1.1 AraT5v2 Base

In full fine tuning, all the parameters are trained and updated, while Table 5 presents trainable Parameters in the other techniques. Figures 3 and 4 araT5v2 variations showcase varying trade-offs between

Table 5: Comparison of the Trainable Parameters in AraT5v2

Optimization	Trainable Parameters	Total Parameters	Percentage of Trainable Parameters
LoRA	1,105,920	368,614,656	0.30%
DoRA	1,105,920	368,614,656	0.30%
QDoRA	1,216,512	368,725,248	0.33%

power consumption, speed, and resource usage. The full fine tuning of araT5v2 is the fastest, completing

training in 2 hours but consumes high power (around 120W) due to its computational intensity. In contrast, araT5v2-Base-with-LoRA balances speed and resource efficiency, using moderate power (50-60W) while finishing second fastest , making it suitable for scenarios where moderate optimization is needed without heavy resource use. Although araT5v2-Base-with-DoRA was slower than full fine tuning and LoRA, DoRA prioritizes power efficiency over speed by applying more optimized updates (magnitude and direction), making it ideal when power usage is a concern and training time is flexible. Lastly, araT5v2-Base-with-QDoRA is the most power-efficient (60W) but takes over 10 hours to train, which makes it useful for resource-constrained settings where power efficiency is extremely crucial. GPU mem-

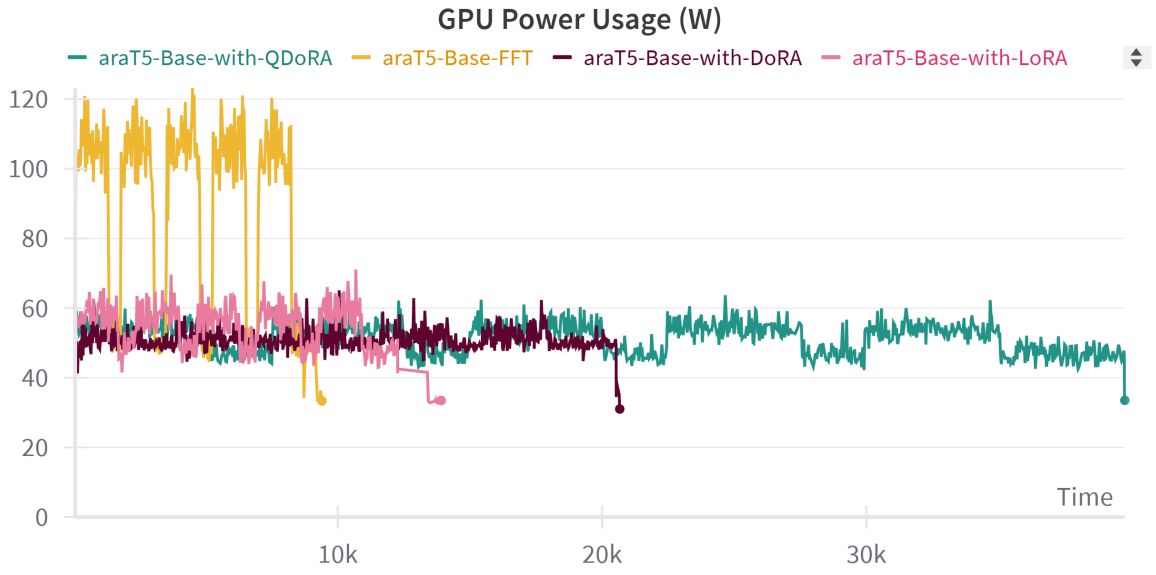


Figure 3: GPU power usage in Watt in araT5v2

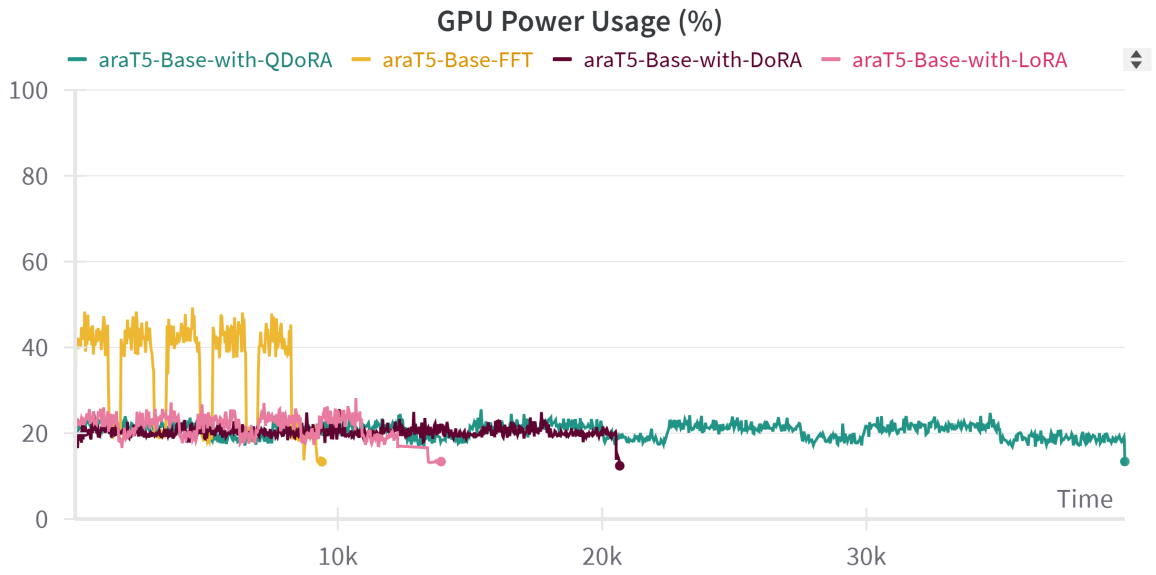


Figure 4: GPU power usage in % in araT5v2

ory directly impacts real-time performance, determining factors like batch size, computation speed, and

whether the model can fit into memory. During fine-tuning, main components such as model parameters, activations, gradients, and optimizer states are stored in GPU memory. Techniques like LoRA, QLoRA, or QDoRA, which reduce the number of trainable parameters or use quantization, significantly lower GPU memory consumption, allowing larger models to be fine-tuned on more affordable hardware. The graph presented in Figure 5 shows GPU memory allocation, providing insight into how much memory each finetuning method allocates during the training process. The araT5v2 variations exhibit varying

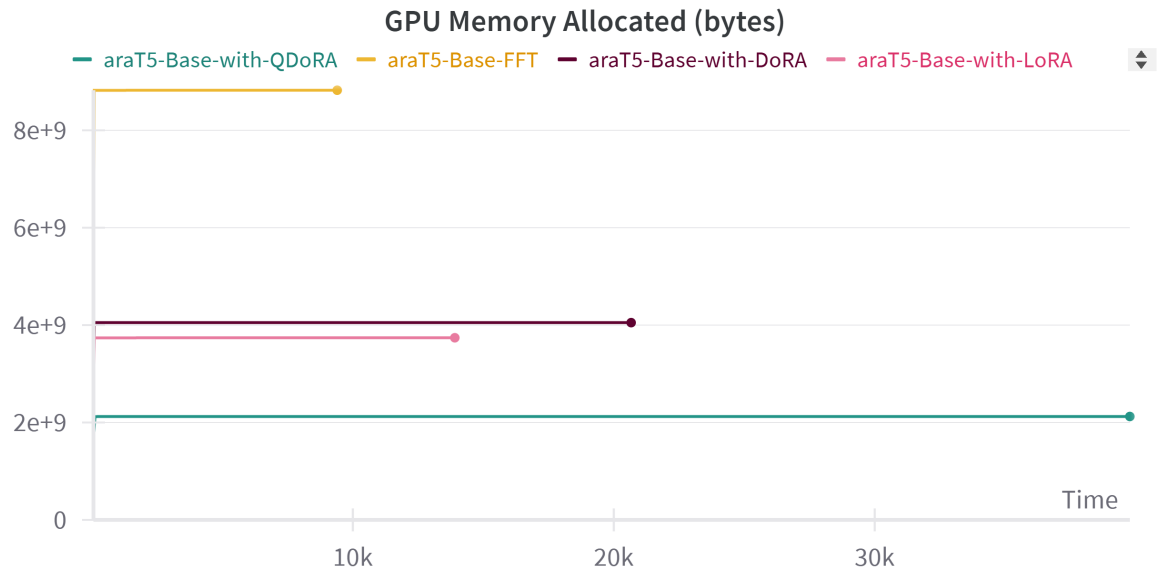


Figure 5: GPU memory allocation in bytes in araT5v2

but stable memory usage based on their tuning strategies. Full fine tuning consumes the most GPU memory (around $8 \cdot 10^9$ bytes) due to its full fine-tuning. LoRA uses less memory (around $3.74 \cdot 10^9$ bytes) by fine-tuning low-rank weight matrices. DoRA is similar to LoRA, using around $4 \cdot 10^9$ bytes, reflecting its efficient parameter updates. QDoRA has the lowest memory allocation (around $2 \cdot 10^9$ bytes) due to fine-tuning applying quantization, but this results in slower training. Full fine tuning is best for scenarios with abundant resources, while LoRA and DoRA balance memory efficiency and speed, making them suitable for more constrained environments. QDoRA excels in memory and power efficiency but sacrifices speed.

LLM's carbon footprint comes from energy consumption during training, primarily using GPUs, highlighting the need for accurate impact assessments for environmental mitigation [19].The GPU temperature graph in Figure 6 illustrates the energy efficiency of four models during training by tracking GPU temperatures over time. AraT5v2-Base-with-QDoRA maintains a stable temperature around $44 - 48°C$, indicating steady, efficient GPU usage over 10 hours. AraT5v2 full fine tuning shows the highest temperature spikes, peaking at $65°C$ during its fast 2-hour training, reflecting high power consumption and a larger carbon footprint. QDoRA and LoRA had moderate temperatures, between 40 and 51 degrees , offering efficient GPU usage but slightly more power than LoRA. AraT5v2-Base-with-DoRA operates at the lowest temperature of $36 - 37°C$, making it the most energy-efficient model with the smallest environmental impact. The graph in the Figure 7 below shows GPU utilization for the different techniques applied on araT5v2 Base. The Full fine tuning peaks at 71%, reflecting again its high computational intensity and fast training time of around 2 hours. In contrast, LoRA and DoRA demonstrate moderate GPU utilization, with LoRA having quicker training than DoRA. Lastly, QDoRA operates between 27-34%, with the longest training time, over 10 hours, prioritizing energy efficiency. Each variant balances speed and resource consumption differently, with Full fine tuning being the fastest and QDoRA the most power-efficient and the slowest.

All variations of araT5v2 show fluctuating CPU usage over time according to Figure 8, with peaks reaching 100% utilization at various points, indicating varying computational intensity during different phases of training, with full fine tuning finishes fast and QDoRA takes the longest period as shown before. These techniques primarily affect the duration of CPU usage rather than the intensity of CPU

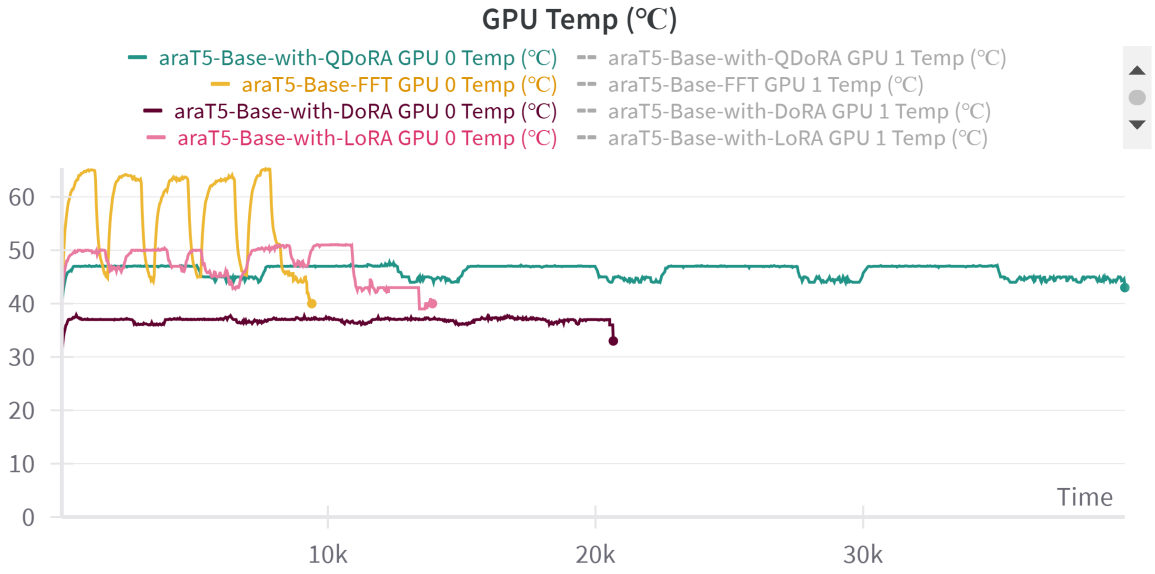


Figure 6: GPU Temperature in araT5v2

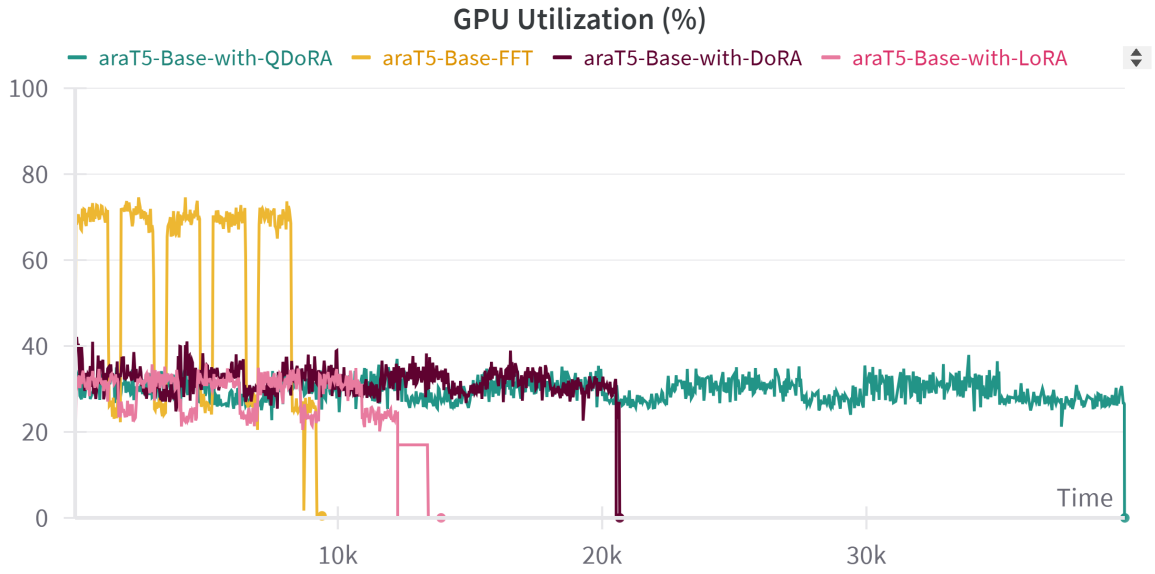


Figure 7: GPU Usage in % in araT5v2

utilization.

5.1.2 NLLB200 600M

The table 6 presents trainable Parameters in the other techniques, in full fine tuning, all the parameters are trained and updated. We examined the trade-offs between GPU power consumption and training speed for the NLLB200-600M model using different fine-tuning techniques. Full fine-tuning consumes the most power, reaching up to 95%, but trains the fastest by updating all model parameters, making it computationally intensive. LoRA uses the lowest power (40-50%) while having the slowest training speed, making it more resource-efficient. DoRA, with power usage around 55-68%, is faster than LoRA

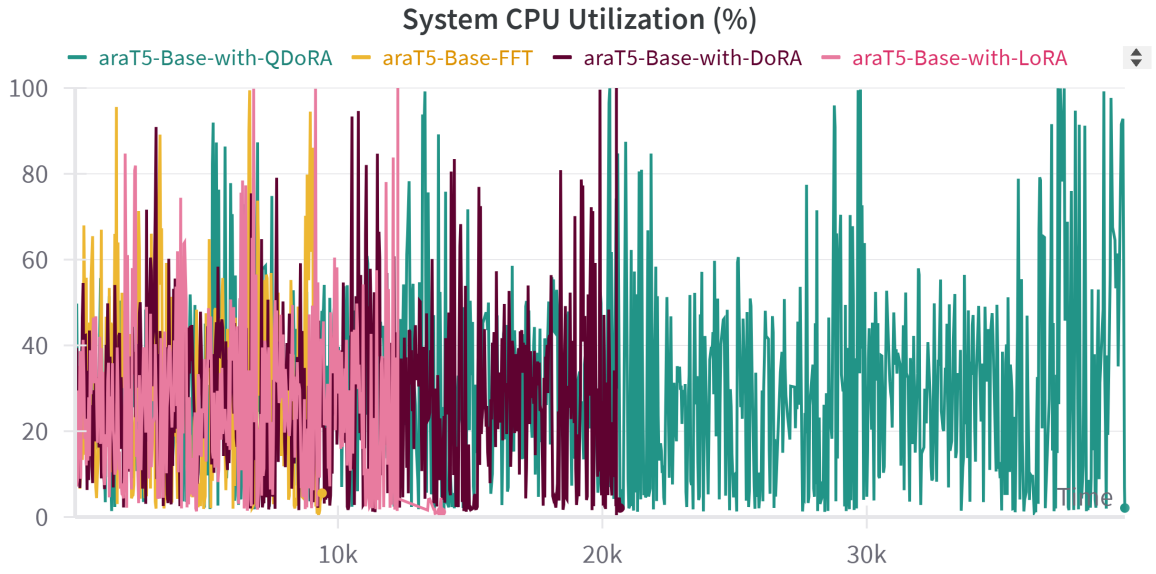


Figure 8: System CPU Usage in % in araT5v2

Table 6: Comparison of trainable Parameters on NLLB200-600M

Optimization	Trainable Parameters	Total Parameters	Percentage of Trainable Parameters
LoRA	1,179,648	616,253,440	0.19%
DoRA	1,253,376	616,253,440	0.20%
QLoRA	1,179,648	616,253,440	0.19%

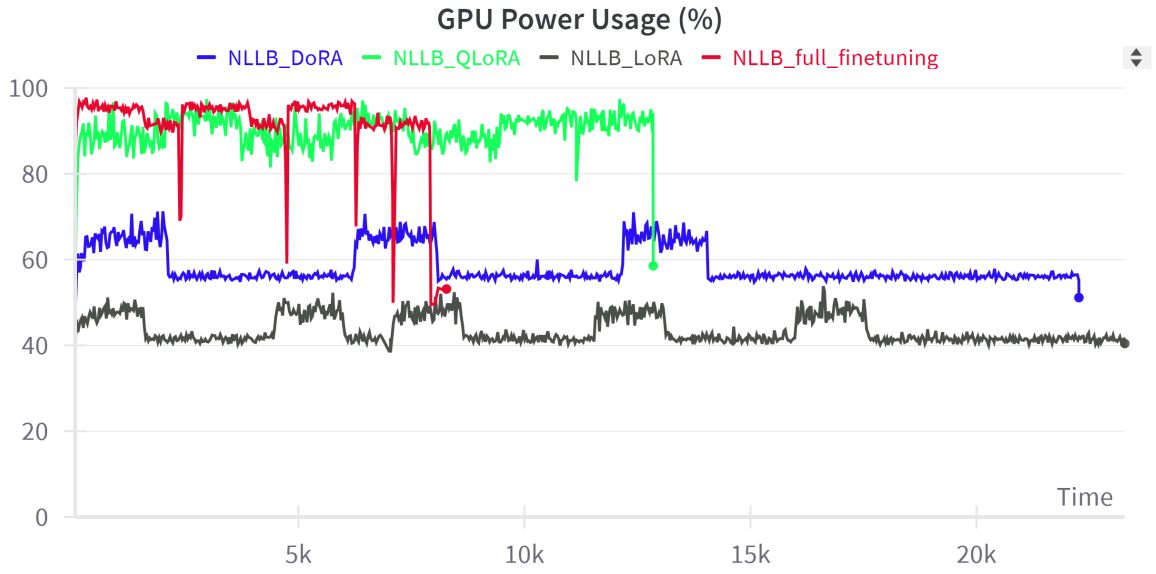


Figure 9: GPU power consumption in % in NLLB200-600M

due to optimized parameter updates while still being more power-efficient than full fine-tuning. QLoRA, despite leveraging quantization for memory efficiency, has high power consumption (82-97%) and requires substantial GPU power at times. Overall, DoRA offers a better balance of power efficiency and speed,

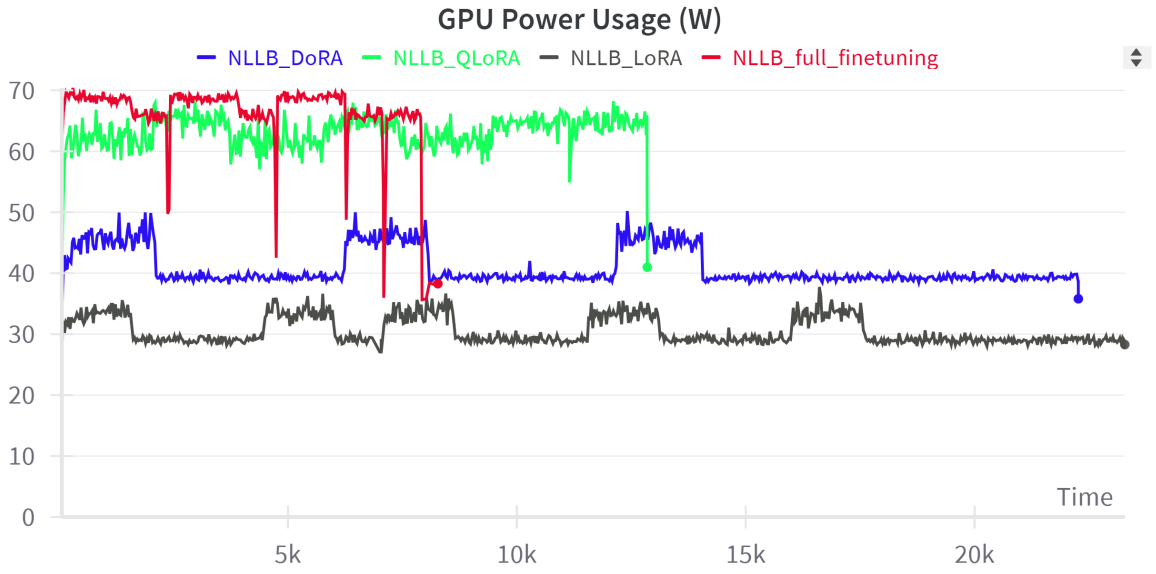


Figure 10: GPU power consumption in Watt in NLLB200-600M

while full fine-tuning and QLoRA prioritizes speed at the cost of high resource use.

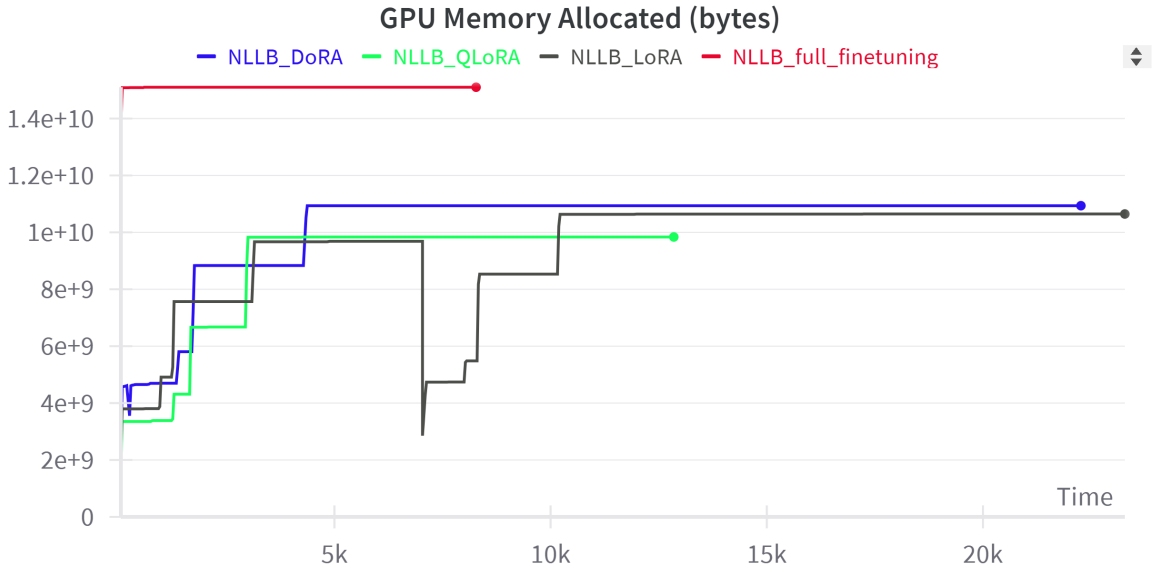


Figure 11: GPU memory allocated in NLLB200-600M

The graph presented in Figure 11 shows the GPU memory allocation over time for various fine-tuning methods of the NLLB200-600M model. Full fine-tuning consumes the most memory, peaking at $1.51 \cdot 10^{+10}$ bytes and remaining constant till the end of its training. DoRA and QLoRA exhibit gradual memory increases, stabilizing at $1.09 \cdot 10^{+10}$ and $9.38 \cdot 10^{+9}$ bytes, respectively, making them more memory efficient (especially QLoRA). LoRA also shows a gradual increase but experiences fluctuations before stabilizing around $1.06 \cdot 10^{+10}$ bytes. Overall, QLoRA is notably the most memory efficient technique, while LoRA shows some instability before leveling off, but takes more time training.

Figure 12 shows the GPU temperature over time for different training methods: The DoRA, QLoRA

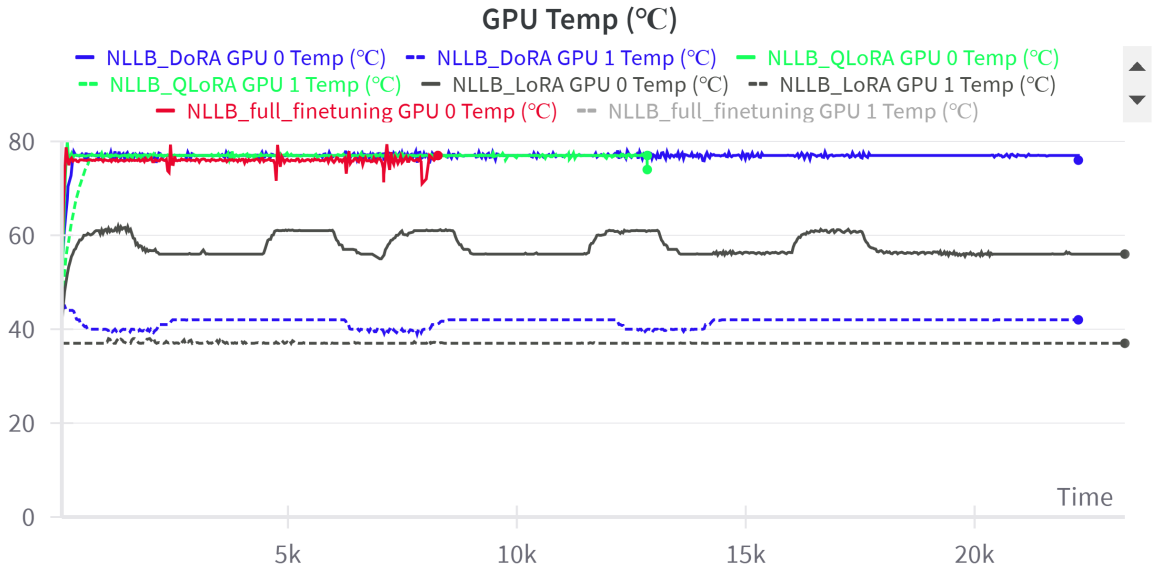


Figure 12: GPU temperature in NLLB200-600M

and full finetuning methods maintain a relatively high and stable temperature, hovering around $75-77°C$. There are minor fluctuations, but overall, the temperature remains steady. The LoRA method shows more balance in temperature, ranging between $56-61°C$, with noticeable dips and rises. It operates at a generally lower temperature compared to the other methods. The Figure 13 shows GPU utilization

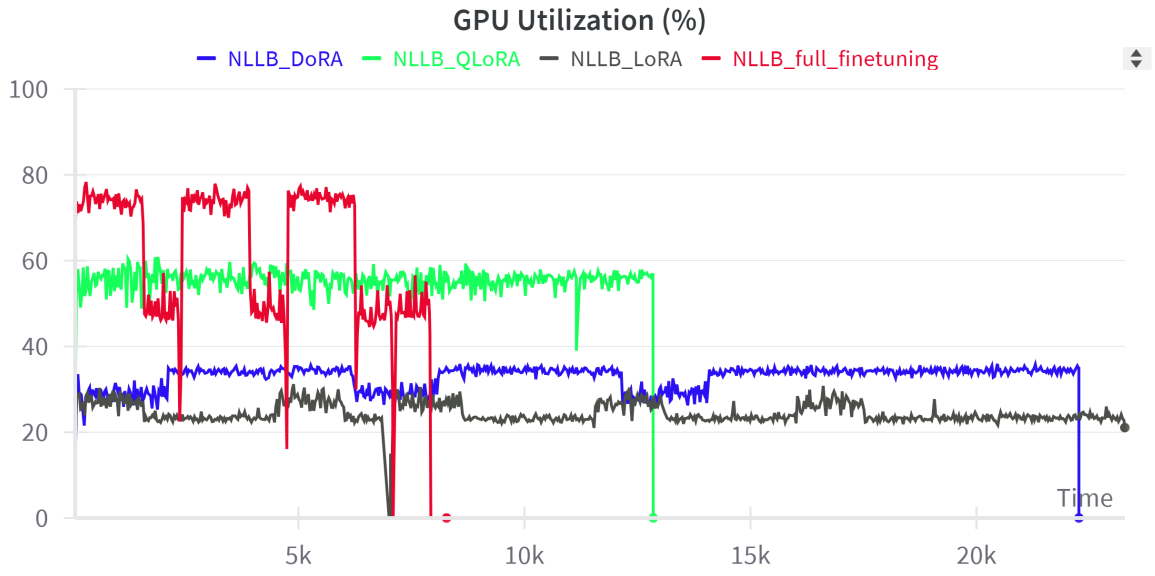


Figure 13: GPU utilization in NLLB200-600M

over time for different methods: Full fine tuning has high GPU utilization, often reaching around 78%. There are occasional drops, but it generally maintains a high level of resource usage, indicating intensive processing. QLoRA also shows relatively high utilization, fluctuating between 49% and 60% and DoRA has moderate utilization, around 26%-35%, indicating a balanced approach between performance and resource usage. LoRA shows the lowest GPU utilization, generally staying below 30%, reflecting a more

optimized use of resources. In general, Full finetuning and QLoRA make the most aggressive use of GPU resources.

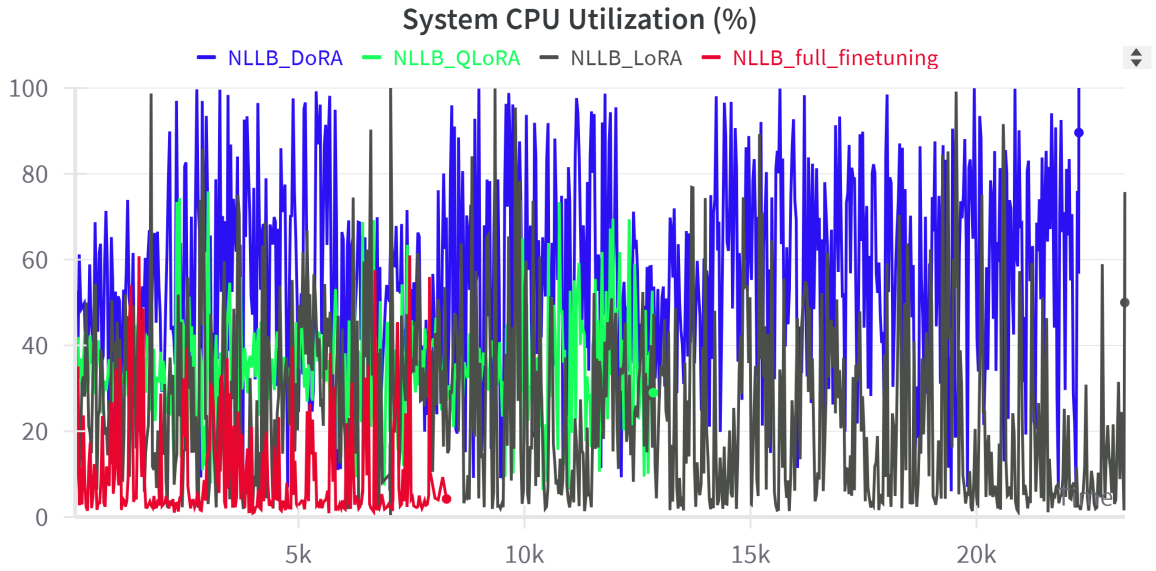


Figure 14: System CPU utilization in NLLB200-600M

The graph in Figure 14 displays system CPU utilization over time for different methods: DoRA and LoRA exhibit high variability with frequent spikes in CPU usage, often reaching above 80, this indicates intensive CPU involvement and processing. QLoRA in Figure 15 maintains the most balanced CPU utilization, with fewer and lower spikes. NLLB200-600M full finetuning displays low and most stable CPU utilization. In total, DoRA and LoRA show more intensive and variable CPU usage, while QLoRA and full finetuning use CPU resources more efficiently and steadily.
In a comparative analysis of fine-tuning methods across models, distinct trade-offs show up in terms of performance, resource usage, and training efficiency. Full Fine-Tuning updates all model parameters, making it the most computationally intensive approach. It achieves the fastest training speed but consumes the most GPU power and memory, limiting its practicality in memory-constrained environments. LoRA and DoRA, which fine-tune a smaller subset of parameters, and QLoRA, which also introduces quantization to further reduce memory and computation demands, offer a compromise. They significantly reduce GPU memory consumption and power usage, with LoRA showing moderate memory access. DoRA has slightly higher power usage than LoRA but converges faster in some cases, making it ideal for scenarios where power efficiency is crucial but training speed remains important. QDoRA, which introduces quantization to further reduce memory and computation demands, uses the least memory, but the added quantization complexity causes occasional GPU power spikes and slower training speeds compared to LoRA. This makes it suitable for extreme memory-constrained environments, regardless of trade-off in speed and performance.

5.2 Translation Quality comparison

5.2.1 Evaluation metrics results

According to the following Bar chart :
Full Fine-Tuning achieves the highest ROUGEL and SacreBLEU scores compared to the other methods. QDoRA and DoRA show nearly identical performance. LoRA shows a similar ROUGE score as DoRA and QDoRA but seems to slightly underperform them in SacreBLEU and Perplexity. While Full Fine-Tuning provides the best results, the gap between LoRA-based methods and Full Fine-Tuning is not large. This reinforces the idea that LoRA-based methods offer a good balance between performance and computational efficiency, especially when full fine-tuning is resource-prohibitive.

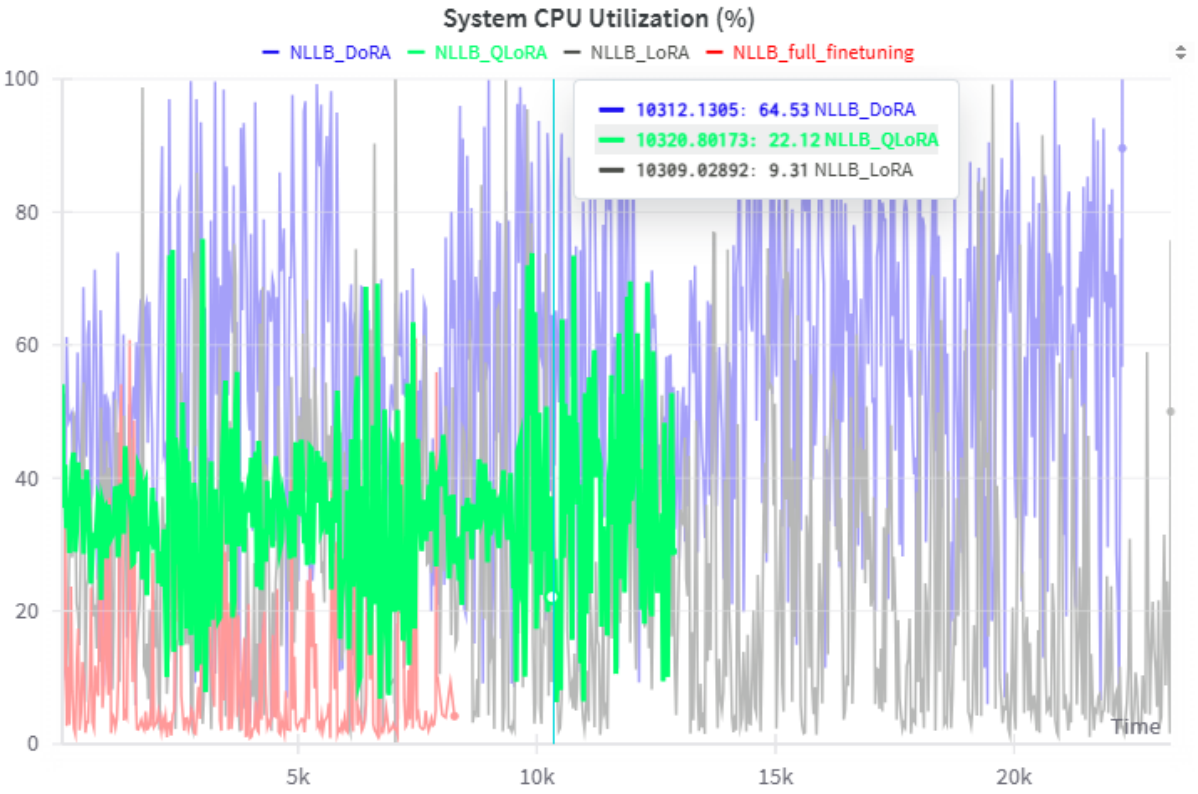


Figure 15: NLLB200-600M-QLoRA system CPU utilization

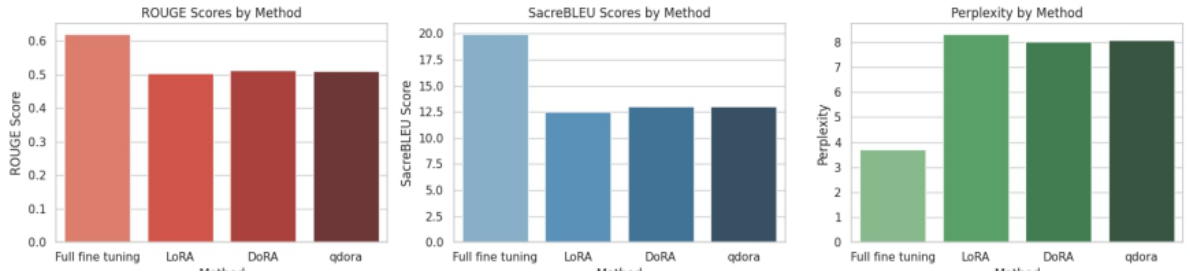


Figure 16: Bar Chart showing comparative araT5v2 Evaluation Using ROUGEL, SacreBLEU, in addition to Perplexity

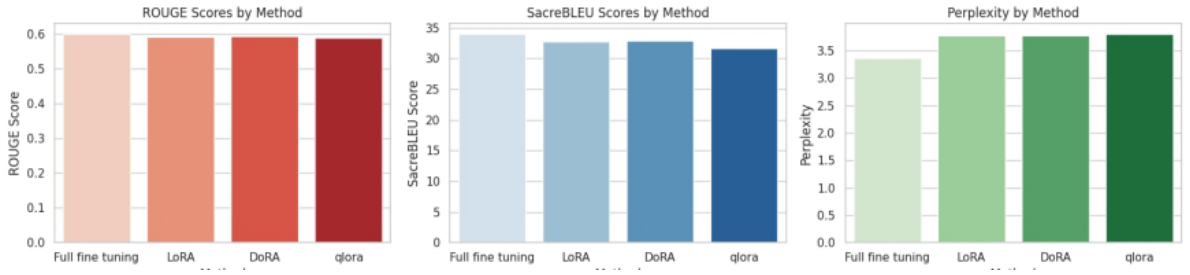


Figure 17: Bar Chart showing comparative NLLB200-600M Evaluation Using ROUGEL, SacreBLEU, in addition to Perplexity

According to Figure 17, The full fine-tuning method starts at a higher ROUGEL and SacreBLEU

scores and remains ahead of all parameter-efficient methods. DoRA and LoRA show almost identical performance, with DoRA slightly edging out. QLoRA remains the lowest performer. The perplexity shows that the uniformity of perplexity scores across all methods shown here is notable. It implies that, in terms of understanding and predicting the structure of the target language, all methods are equally effective.
Full fine-tuning offers the best evaluation scores results, but parameter-efficient methods like DoRA and LoRA are closely following, suggesting they provide a competitive trade-off between performance and resource usage.

5.2.2 Human evaluation results

In our inference tests, Arabic sentences were translated using AraT5v2 and compared with Google Translate. For simple sentences, all fine-tuning methods (Table 7) performed similarly, showing that lightweight techniques like LoRA and QDoRA can handle basic translations effectively. However, errors in complex sentences - especially with idiomatic expressions - revealed significant limitations. Google Translate generally outperformed the fine-tuned models in these cases, delivering more fluent translations, while the models often struggled with figurative language, producing more literal or incomplete outputs.
Noticeably, some Arabic words remained untranslated across all fine-tuning methods, indicating gaps in vocabulary handling. Full Fine-Tuning provided the best overall performance but occasionally overcomplicated translations, likely due to overfitting.
For NLLB200-600M (Table 8), all fine-tuning methods closely matched Google Translate on straightforward sentences but diverged on abstract phrases. Overall, Full Fine-Tuning excelled but was resource-heavy, while LoRA and DoRA offered a balanced trade-off, and QLoRA proved best suited for extreme memory constraints but was slower to converge.

Table 9 presents comparison between GPT models BLEU scores and ours, due to hardware limitations, we were unable to use similarly powerful models in our experiments, resulting in comparatively lower performance to the results we presented in the related works.

6 Conclusion

This study explored the trade-offs between computational efficiency and Arabic-to-English translation quality in LLMs, focusing on parameter-efficient fine-tuning techniques, particularly variations of LoRA. While full fine-tuning showed optimal translation accuracy, LoRA and DoRA achieved comparable quality with reduced computational costs remarkably. QLoRA offered additional memory efficiency, though at the expense of longer training time. Future work will explore scaling these methods to larger models and enhancing their capacity to capture complex linguistic structures. All code and models are provided in a GitHub repository as open source, editable Jupyter notebooks.

Table 7
Translations generated by araT5v2

Arabic original text	DoRA	Full fine tuning	LoRA	QDoRA	Google Translate
قررت أن أغير وظيفتي لأبحث عن فرص جديدة	I have decided to change my job to find new opportunities	I decided to change my job to seek new opportunities	I decided to change my job to find new opportunities.	I decided to change my job to find new opportunities.	I decided to change my job to look for new opportunities
السَيفُ أَصْدَقُ إِنْبَاءً مِنَ الكُتُبِ في حدِه الحدُ بينَ الجِدِ واللَعْبِ	The spend of books is stronger from the table of books, in the same line between the study and the game	The cost estimate provides for the highest number of newspapers at the extent of	The spend of books is expenditure at the line of the line between the and game.	The printing of books are expenditure to the time and the game.	The sword is more truthful than books in its sharpness, the line between seriousness and play.
السيف اصدق انباء من الكتب في حده الحد بين الجد واللعب	السيف is the most accurate of books, in the scale of the time and game.	The sword is the most best source of books at the time of serious and play	The السيف is most accurate of books in the same line between serious and game.	The السيف is the best evidence of books in the same line between the argument and play	The sword is more truthful than books in its sharpness, the limit between seriousness and play

Table 8
Translations generated by NLLB200-600M

Arabic original text	DoRA	Full fine tuning	LoRA	QLoRA	Google Translate
قررت أن أغير وظيفتي لأبحث عن فرص جديدة	I decided to change jobs to look for new opportunities.	I decided to change jobs to look for new opportunities	I decided to change jobs to look for new opportunities.	I decided to change jobs to look for new opportunities.	I decided to change my job to look for new opportunities
السَيفُ أَصْدَقُ إِنْبَاءً مِنَ الكُتُبِ في حدِه الحدُ بينَ الجِدِ واللَعْبِ	The sword I believe is a book narrative about the boundary between grandfather and play.	The sword. I believe a prophecy from a book alone about the boundary between seriousness and play	I believe the sword is a book narrative about the boundary between grandfather and play.	I believe the sword is a book narrative about the boundary between grandfather and play.	The sword is more truthful than books in its sharpness, the line between seriousness and play
السيف اصدق انباء من الكتب في حده الحد بين الجد واللعب	The sword believes prophecies from books about the boundary between grandpa and play.	The sword believed the prophecies of the books in the boundary between grandfather and play.	The sword believes prophecies from books about the boundary between grandfather and play.	Sword believes prophecies from books about the boundary between grandfather and play.	The sword is more truthful than books in its sharpness, the limit between seriousness and play

Table 9
BLEU Scores for GPT Language Models compared to araT5v2 Base and NLLB200-600M

Model	BLEU score
GPT-4	∼**45**
ChatGPT	∼**40**
araT5v2 Full fine tuning	**19.951**
araT5v2 Full LoRA	12.531
araT5v2 Full DoRA	13.006
araT5v2 Full QDoRA	**13.027**
NLLB200-600M Full fine tuning	**34.245**
NLLB200-600M LoRA	32.676
NLLB200-600M DoRA	**32.812**
NLLB200-600M QLoRA	31.595

References

[1] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning, 2020.

[2] Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. In-context examples selection for machine translation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8857–8873, Toronto, Canada, July 2023. Association for Computational Linguistics.

[3] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. The falcon series of open language models, 2023.

[4] Golla Anjali, Santosh Sanjeev, Akuraju Mounika, Gangireddy Suhas, G. Pradeep Reddy, and Yarlagadda Kshiraja. Infant cry classification using transfer learning. In *TENCON 2022 - 2022 IEEE Region 10 Conference (TENCON)*, pages 1–7, 2022.

[5] Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization, 2022.

[6] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

[7] Tim Dettmers and Luke Zettlemoyer. The case for 4-bit precision: k-bit inference scaling laws, 2023.

[8] Raman Dutt, Ondrej Bohdal, Sotirios A. Tsaftaris, and Timothy Hospedales. Fairtune: Optimizing parameter efficient fine tuning for fairness in medical image analysis, 2024.

[9] Raman Dutt, Linus Ericsson, Pedro Sanchez, Sotirios A. Tsaftaris, and Timothy Hospedales. Parameter-efficient fine-tuning for medical image analysis: The missed opportunity, 2024.

[10] AbdelRahim Elmadany, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. Octopus: A multitask model and toolkit for Arabic natural language generation. In Hassan Sawaf, Samhaa El-Beltagy, Wajdi Zaghouani, Walid Magdy, Ahmed Abdelali, Nadi Tomeh, Ibrahim Abu Farha, Nizar Habash, Salam Khalifa, Amr Keleg, Hatem Haddad, Imed Zitouni, Khalil Mrini, and Rawan Almatham, editors, *Proceedings of ArabicNLP 2023*, pages 232–243, Singapore (Hybrid), December 2023. Association for Computational Linguistics.

[11] Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André F. T. Martins, Graham Neubig, Ankush Garg, Jonathan H. Clark, Markus Freitag, and Orhan Firat. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation, 2023.

[12] Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. Gptscore: Evaluate as you desire, 2023.

[13] Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. Dictionary-based phrase-level prompting of large language models for machine translation, 2023.

[14] Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538, 2022.

[15] Soufiane Hayou, Nikhil Ghosh, and Bin Yu. Lora+: Efficient low rank adaptation of large models, 2024.

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR, 09–15 Jun 2019.

[17] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[18] Vivek Iyer, Pinzhen Chen, and Alexandra Birch. Towards effective disambiguation for machine translation with large language models, 2023.

[19] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning, 2019.

[20] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning, 2021.

[21] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation, 2021.

[22] Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual language models, 2022.

[23] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation, 2024.

[24] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too, 2023.

[25] Hongyuan Lu, Haoran Yang, Haoyang Huang, Dongdong Zhang, Wai Lam, and Furu Wei. Chain-of-dictionary prompting elicits translation in large language models, 2024.

[26] Zhuoyuan Mao and Yen Yu. Tuning llms with contrastive alignment instructions for machine translation in unseen, low-resource languages, 2024.

[27] Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. Adaptive machine translation with large language models. In Mary Nurminen, Judith Brenner, Maarit Koponen, Sirkku Latomaa, Mikhail Mikhailov, Frederike Schierl, Tharindu Ranasinghe, Eva Vanmassenhove, Sergi Alvarez Vidal, Nora Aranberri, Mara Nunziatini, Carla Parra Escartín, Mikel Forcada, Maja Popovic, Carolina Scarton, and Helena Moniz, editors, *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland, June 2023. European Association for Machine Translation.

[28] Yasmin Moslem, Gianfranco Romani, Mahdi Molaei, John D. Kelleher, Rejwanul Haque, and Andy Way. Domain terminology integration into machine translation: Leveraging large language models. In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz, editors, *Proceedings of the Eighth Conference on Machine Translation*, pages 902–911, Singapore, December 2023. Association for Computational Linguistics.

[29] El Moatez Billah Nagoudi, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. Arat5: Text-to-text transformers for arabic language generation, 2022.

[30] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024.

[31] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[32] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.

[33] Vikas Raunak, Amr Sharaf, Yiren Wang, Hany Awadalla, and Arul Menezes. Leveraging GPT-4 for automatic translation post-editing. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12009–12024, Singapore, December 2023. Association for Computational Linguistics.

[34] Abudurexiti Reheman, Tao Zhou, Yingfeng Luo, Di Yang, Tong Xiao, and Jingbo Zhu. Prompting neural machine translation with translation memories, 2023.

[35] Raphael Reinauer, Patrick Simianer, Kaden Uhlig, Johannes E. M. Mosig, and Joern Wuebker. Neural machine translation models can learn to be few-shot learners, 2023.

[36] Gabriele Sarti, Phu Mon Htut, Xing Niu, Benjamin Hsu, Anna Currey, Georgiana Dinu, and Maria Nadejde. RAMP: Retrieval and attribute-marking enhanced prompting for attribute-controlled translation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1476–1490, Toronto, Canada, July 2023. Association for Computational Linguistics.

[37] Mike Schuster and Kaisuke Nakajima. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152, 2012.

[38] NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation, 2022.

[39] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[40] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. Dylora: Parameter efficient tuning of pre-trained models using dynamic search-free low-rank adaptation, 2023.

[41] Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, Xiaopeng Zhang, and Qi Tian. Qa-lora: Quantization-aware low-rank adaptation of large language models, 2023.

[42] Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. Bigtranslate: Augmenting large language models with multilingual translation capability over 100 languages, 2023.

[43] Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. Improving massively multilingual neural machine translation and zero-shot translation, 2020.

[44] Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, and Yang Feng. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models, 2023.

[45] Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Multilingual machine translation with large language models: Empirical results and analysis, 2024.

[46] Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Extrapolating large language models to non-english by aligning languages, 2023.

[47] Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. The United Nations parallel corpus v1.0. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

A Survey on Approaches to Modeling Collaborative Practices in E-Learning Platforms

Sara Ghaoui [1], Sofiane Mounine Hemam[2], and Tarek Djouad[3]

[1]*ICOSI Laboratory, Abbes Laghrour University, Khenchela, Algeria,* sara.ghaoui@univ-khenchela.dz
[2]*Abbes Laghrour University, Khenchela, Algeria, National High School of Cyber-Security, Algiers, Algeria. ,* hemam.sofiane@univ-khenchela.dz, sofiane.hemam@enscs.edu.dz
[3]*ICOSI Laboratory, Abbes Laghrour University, Khenchela, Algeria,* tarek.djouad@univ-khenchela.dz

Abstract

Evaluating collaborative practices for peers/groups of learners in collaborative e-learning is a crucial issue in distance learning platforms. It is a complex task that requires the development of advanced methods and tools to ensure continuous and real-time evaluation of collaboration. The aim of our work is to propose and implement an algorithm, method, and tool for evaluating collaborative learning. We seek to identify and extract collaborative fragments by applying operators to modeled traces in order to pinpoint sequential episodes of collaboration. Additionally, we aim to design and compute collaboration indicators. The objective of this work is to simplify the process of evaluating a group of learners on a collaborative distance-learning platform, enabling non-computer scientists to design their own collaboration indicators and automate their calculation.

Keywords: Collaborative e-learning, distance learning platforms, Sequential episodes of collaboration, collaboration indicator.

1 Introduction

Distance learning platforms are environments that support, accompany, and validate learning, where learners collaborate to achieve a common goal[13]. Collaborative e-learning, as a pedagogical approach, relies on the sharing and construction of knowledge among learners using technology[11].

Collaboration is defined as "the mutual engagement of participants in a coordinated effort to solve a problem together"[20]. Cooperative work, on the other hand, "is defined as a form of work organization where each operator is responsible for his or her part. Collaborative work, in contrast, is a form of work organization in which everyone is responsible for the whole"[8]. In e-learning, the term 'collaboration' is generally preferred over 'cooperation,' despite both terms meaning 'working together.' The aim of learning is not simply to complete a task collectively and produce a final product, but to ensure that all learners achieve the same concepts and reach the desired objectives.

The main goal of collaborative learning is to enable a group of learners to work together through a computer system to achieve a collaborative task. This task may involve completing a project, solving an exercise, or understanding a concept. Collaboration can take the form of sharing, exchanging, or discussing information, ideas, and concepts, enabling learners to develop the cognitive skills and knowledge necessary to enhance their competencies. According to [2], collaborative working increases employee productivity and results in higher-quality outcomes. Moreover, regardless of their status, employees report higher levels of satisfaction and responsiveness.

The lack of information about the level of collaboration within a group or between groups presents challenges for teachers who wish to evaluate learners' collaborative behavior. They must answer questions such as: Who participates? Who doesn't? Who helped whom? Who did what? These questions are often difficult to answer when analyzing the dynamics of a collaborative group.

To understand the behavior of a learner or group of learners involved in e-learning, and to provide relevant and adequate information to the teacher or trainer monitoring progress, whether globally or individually, it is necessary to track traces. These traces can be defined as a set of temporally situated elements.

1.1 Research Problem

The problem addressed in much of the research in related fields is how to evaluate the collaborative work of one or more groups of learners on a collaborative e-learning platform.

In the context of our work, several research questions were posed:

1. How can collaborative activities be detected within an e-learning platform?
2. How can these collaborative practices be evaluated?

The central problem concerns how to analyze the traces obtained during a collaborative learning session in order to answer the above questions.

Our main contribution is to propose an algorithm, a method, models, and a tool for extracting and evaluating collaborative practices through modeled traces.

1.2 Research Objectives

To achieve these results, we have set multiple objectives:

1. Proposing an algorithm for extracting sequential episodes of collaboration in order to identify collaboration fragments.
2. Proposing an MDA-based method for calculating collaboration indicators.
3. Proposing a tool for the design and automatic calculation of collaboration indicators.

1.3 Paper Organization

To achieve the above objectives, the rest of the paper is organized as follows:

- The first part provides the theoretical framework for collaborative work and its evaluation.
- The second part presents our contribution and the proposed approach to achieving our goals.
- Finally, we conclude the paper with a summary of our work.

2 State of the Art

2.1 Computer-Assisted Collaborative Learning

Collaborative learning is a learner-centered approach in which students actively construct their knowledge, with the instructor playing the role of facilitator. This model contrasts with the traditional teacher-centered approach. The integration of Information and Communication Technologies (ICT) in distance learning platforms has transformed pedagogy by fostering the emergence of collective learning through tools such as forums, wikis, and blogs. These platforms overcome obstacles like physical distance and learner diversity, enhancing collaboration and mutual support. The role of different actors (teachers, tutors, and learners) is crucial for ensuring a conducive learning environment. This section explores the advantages, limitations, and challenges associated with collaborative e-learning, with a particular focus on the evaluation of collaboration to prevent isolation and improve learner engagement. It also discusses the definition of collaborative learning, the approaches to learning supported by distance learning platforms, the roles of various actors in these learning environments, and the importance of evaluating collaboration in an e-learning context.

2.1.1 Definition of Collaborative Learning

There are several approaches to learning, including traditional (teacher-centered) and collaborative (learner-centered) approaches.

According to Henri and Lundgren-Cayrol[11], "Collaborative learning is an active approach in which the learner works to construct his or her own knowledge. The trainer plays the role of learning facilitator, while the group participates as a source of information, a motivator, a means of mutual help and support, and a privileged space for the collective construction of knowledge."

In this type of learning, the learner takes responsibility for their own personal development and engages in collaboration with group members to achieve a common goal—learning. Throughout this process, collaboration within the group allows members to share, negotiate, and validate their newly constructed knowledge.

2.1.2 Towards Collaborative Learning Supported by Distance Learning Platforms

The deployment of Information and Communication Technologies (ICT) in distance learning platforms has brought about significant changes in pedagogy.

The variety of collaborative tools available on these platforms, such as forums, wikis, blogs, and others, has fostered the emergence of collective learning. A collaborative learning environment supported by a distance learning platform promotes the desire to exchange, communicate, and share, as well as to participate and collaborate.

2.1.3 Actors in a Collaborative Learning Situation

In a collaborative learning environment supported by a distance learning platform, the following roles are typically considered the main ones: teacher, IT designer, tutor, learner, and administrator[20].

2.1.4 The Role of the Tutor

The online tutor assumes various roles, such as coach, facilitator, instructor, and evaluator. They adopt and implement strategies aligned with the learning/teaching paths chosen for the collaborative learning situation. Additionally, they hold a supervisory role, supporting learners, stimulating learning, and communicating rules within the learning environment[18].

2.1.5 Advantages of Collaborative E-Learning

In addition to the flexibility of time and place that collaborative e-learning offers to learners, it also fosters cognitive and personal growth. Learners develop by working together toward a common goal[8]. In this collaborative learning process, the learner adapts to the benefits and demands of collaboration and learns to use discussion and negotiation in their interactions with group members to build knowledge.

2.1.6 Limitations of Collaborative Learning

Despite the advantages of collaborative e-learning, there are several limitations to consider, whether in terms of balance, heterogeneity, group size, or assessment procedures[8]. This work focuses more specifically on the limitations related to assessment procedures.

2.1.7 Why We Should Evaluate Collaboration in an E-Learning Situation

One of the main problems with most e-learning platforms is student drop-out. A primary factor contributing to this issue is the absence of support and social relationships, which can lead to feelings of isolation.

2.2 Evaluating Collaborative Processes

Evaluating the collaborative e-learning process is a delicate task that has prompted researchers to engage in various theoretical and methodological investigations to address its challenges.

Before the discovery of the concept of M-traces, the description of elements stored during an e-learning session was limited to textual documentation, such as log files or RSS feeds. This made human exploitation of the data very challenging and almost impossible when dealing with large volumes of interactions.

In 2006, Yannick Prié and his colleagues [21] introduced a new computer object called the "M-trace" (modeled trace), which associates each collection of observed elements with a model to formally describe the structure and content of the trace.

2.3 Interaction Indicators

In the context of learning, the DPULS project[4] has provided a clearer definition of the concept of an indicator: "It is a pedagogically significant variable, calculated or established using observed data, that reflects the quality of interaction, activity, and learning." Indeed, a collaboration indicator is one that provides information on the level of participation, collaboration, and the degree of involvement of learners in collaborative work.

According to Djouad[6], each indicator has a name, a textual specification, and a calculation rule. As shown in Figure 2, to arrive at the final value of an indicator, several stages must be considered: from the collection of the necessary traces to their processing, to the formalization of the calculation methods, and finally, to the visualization and interpretation of the data obtained. In all these stages, the central step is the modeling and calculation of indicators.

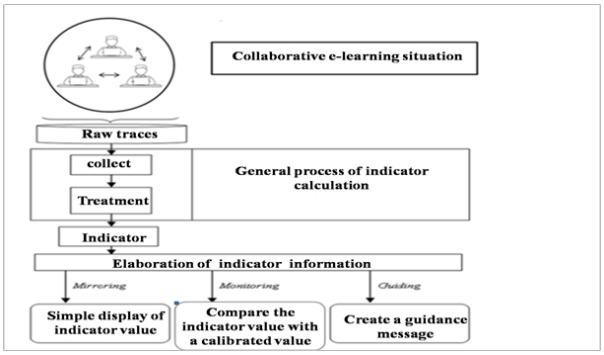


Figure 1: Indicator life cycle

3 Related work and scientific positioning

Recently, numerous studies have focused on evaluating the effectiveness of monitoring indicators in collaborative systems[22]. Integrating these evaluation metrics has proven valuable for tracking and enhancing learner engagement in online environments, making them essential for adaptive and personalized learning experiences[16].

In [17], the authors examine how automated analytics can assess collaboration skills by analyzing group speech data. They analyze communication patterns, detect engagement levels, and identify collaboration dynamics in real time.

The ICALTS project [12] identified indicators through the analysis of students' interactions at the metacognitive level, which could help learners self-regulate or evaluate their activity. Similarly, in [3], the authors proposed indicators to assess learning activity based on discussion forum posts. These indicators are also used to validate the quality of asynchronous discussions without requiring an in-depth content analysis.

In [5], the authors developed a mechanism allowing students to examine a collaborative task from different perspectives. They proposed a set of metric-based indicators to evaluate the group's output as the final product of collaborative work. Meanwhile, in [9], researchers focused on calculating collaboration indicators in Moodle by utilizing learning analytics and data mining techniques to define specific collaboration indicators such as participation rate, interaction frequency, response time, message length, and role distribution in group discussions.

In [14], specific algorithms were designed to extract and compute collaboration indicators within a structured framework spanning multiple dimensions.

However, these indicators are often designed in an ad hoc manner and are typically tailored to a specific platform, with little consideration for reusability.

A second category of research focuses on developing methods and tools to facilitate the design and computation of collaboration indicators. The authors of [6] proposed a model-driven engineering approach to simplify the analysis of traces in learning situations. Their method involves saving the transformations applied to the traces to facilitate their reuse, allowing the transition from raw trace data to indicator models.

In [10], a computational tool called Genidic was developed to assist users in the development, management, and computation of indicators. It employs a rule-based system where traces serve as facts, and indicator calculation processes are defined as rules. Similarly, [19] introduced Usage Tracking Language (UTL), which allows the definition of indicators in a design pattern-like format to enhance capitalization and reuse. However, UTL initially lacked formal tools to specify how indicators should be computed from collected traces. To address this limitation, [15] proposed a new version called DCL4UTL, which enables indicators to be modeled in a structured way that supports automation and reuse, providing valuable insights for teachers and tutors.

Other approaches rely on multi-agent systems. For example, [7] proposed a system based on fuzzy logic techniques to evaluate the level of learner collaboration. The inputs to this system are calculated indicators derived from trace analysis. Similarly, in [14], the authors developed a cloud-based Learning Management System (LMS) that integrates a multi-agent system to collect, analyze, and filter traces, facilitating the computation of interaction indicators that promote collaboration.

The below table 1 summarizes and compares the above work with our proposition.

Work	The suggested method supports the design /calculation of the indicators	Indicator type	Used approach	The implementation system is open or closed
[19]	The design	Indicators in CEHL	• Designing of indicator in a form similar to a design pattern	opened
[6]	The calculation	Indicators in MOODLE platform	• Oriented Model Transformation. • based on MDE. • Using an m-trace-based system(self-developed).	Closed
[10]	The design and calculation	Collaboration indicators	• Using a collaboration indicator pattern for the design. • Using a Computation method oriented Artificial intelligence and based on rule-logic. • Using a rule-based system(self-developed)	Opened
[15]	The design and calculation	Indicators in CEHL	• Enrich UTL with formal language to formally describe the calculation method. • Indicator computation using a DCL4UTL interpreter (self-developed) integrated into a trace analysis tool.	Opened

[1]	The calculation	Collaboration indicators in collaborative e-learning systems	Ad-hoc Manner	Closed
[14]	The calculation	Collaboration indicators in e-learning systems	• Oriented Model Transformation. • Basing on MDE. • Using A multi-agent system (self-developed) and an m-trace based system(KTBS).	Closed
Our work	The design and calculation	Collaboration indicators in an e-learning systems	• The design is based on the DCIN Model(self proposal) • The calculation is oriented model-transforma- tion based on MDE. • The sequences of transforma- tions are automatiquelly generated using DCIN-AGSET(self developped).	Opened

Table 1: The comparaison between related works and our proposition

Compared to the above works, our proposition considers the design and computation of collaborative indicators in e-learning systems. Therefore, our approach can be summarized as follows:

- The first aspect focuses on the computation of collaboration indicators in e-learning systems, independently of any platform used. For this purpose, we propose to apply a model transformation approach and an MDA-based process to obtain collaboration indicator models.
- The second aspect focuses on the design of collaboration indicators in e-learning systems. For this purpose, we propose a formal model that facilitates the design of valid and meaningful collaboration indicators according to the teacher's observation needs.
- Based on the application development process supported by Model-Driven Architecture, the calculation of the collaboration indicator can be seen as a model transformation process, where the trace model is passed through a sequence of transformations to arrive at the collaboration indicator model. For the same previous need of automatic computation of collaboration indicators and the acquisition of specific computer skills that cannot be achieved by a non-computer scientist teacher, we propose to automate the generation of sequences of transformations.
- The third aspect focuses on how to obtain sequences of transformations. For this, we will propose a system that ensures the automatic generation of sequences of transformations to be applied in the m-trace base system to arrive at the indicator model.

4 Our Contribution

Our main objective is to propose an algorithm, a method, and a tool to facilitate the evaluation of learners' collaborative behavior in a collaborative learning environment.

To achieve this goal, our contribution will be divided into three main parts:

4.1 The first contribution

The first part of our contribution involves proposing an algorithm for extracting collaborative episodes from a trace. The application of this algorithm will extract collaborative fragments within an interaction trace. This algorithm will be:

- The first tool that teachers can use to detect the collaborative behavior of their learners, and
- Supported by statistical and mining functions to detect specific aspects of the trace, such as extracting frequent collaborative episodes, the number of frequent sequential episodes, etc.

The collaborative fragment extraction algorithm we propose is based on the frequent sequential episode extraction algorithm, with the key difference that the result is a "collaborative sequential episode."

Let's take the following example:

- Let the following trace be:

- Let the collaboration obsels be:

- Applying the frequent sequential episode extraction algorithm, we obtain the following frequent episode with a seuil greater than or equal to 2:

- Applying our method to propose, we'll obtain the following episodes with a length greater than or equal to 2:
- The calculation of collaboration indicators will be based on these results.

4.2 The Second Contribution

The second part of our contribution involves proposing an approach that enables non-computer-scientist teachers to design and calculate their own collaboration indicators to detect and evaluate learners' collaborative behavior in collaborative e-learning environments.

Our approach will be based on a model-driven architecture, where the calculation of indicators can be seen as a series of model transformations. After the design stage, we aim to automate the calculation of the teacher-designed indicators by automatically generating the transformation sequence needed to obtain the corresponding indicator model.

4.3 The Third Contribution

The third part consists of proposing a real case study using a learning platform with an online collaborative learning situation and an evaluation grid to implement and adjust the proposed algorithms and methods. This part is organized as follows:

1. Proposing a collaborative learning situation on a learning platform to collect the necessary data interaction traces during collaborative learning sessions.

2. Proposing a model for the concept of 'collaborative trace' within the trace based system'KTBS', which will serve as an ontology for validating trace models.

3. Developing operators that facilitate the construction of collaborative traces, including a confidence factor.

4. Implementing the evaluation algorithms and integrating several proposed indicators to assess the collaborative learning process (Figure 2 summarizes these steps).

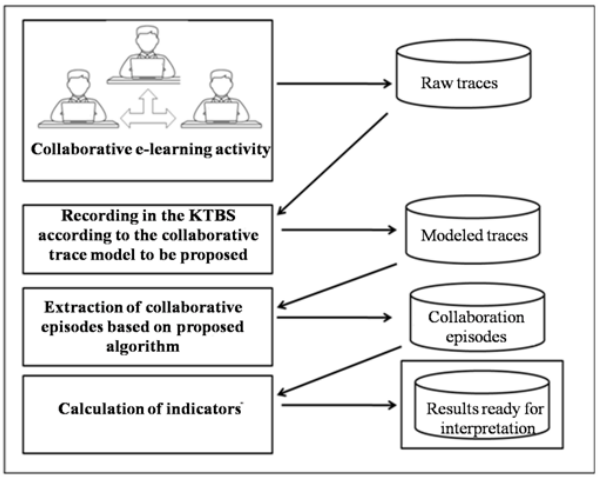


Figure 2: Collaboration evaluation stage in a collaborative learning situation.

5 Conclusion

Collaborative distance learning represents a valuable approach for enhancing group work and developing collaborative skills. After years of research, the interactions between learners using various distance learning tools can be captured through a computer object called M-Trace.

In this research, we focused on evaluating collaborative practices within a learning activity. Our first contribution is the development of an algorithm to detect and extract collaborative fragments from interaction traces. By adapting the principle of frequent sequential episode extraction, we propose a novel algorithm tailored to extract sequential episodes of collaboration.

The second contribution lies in evaluating collaborative practices. We propose a Model-Driven Architecture (MDA)-based method to calculate collaboration indicators, coupled with a tool that enables non-computer-scientist teachers to design meaningful collaboration indicators. This will allow teachers to generate the transformation sequence necessary to obtain the corresponding collaboration indicator model.

Finally, we plan to conduct a case study to further refine and adjust the evaluation algorithms. This case study will help validate the proposed methods and tools, ensuring their practical applicability in real-world educational settings.

References

[1] A. Acosta and J. Lee. Multimodal learning analytics for predicting student collaboration satisfaction. In *Proceedings of the 17th International Conference on Educational Data Mining (EDM 2024)*, 2024.

[2] M. Arnaud. Current limitations of online collaborative learning (in french). *STICEF (Information and Communication Sciences and Technologies for Education and Training)*, 10:7, 2003.

[3] T. Bratitsis and A. Dimitracopoulou. Monitoring and analyzing group interactions in asynchronous discussions with the dias system. In Y. A. Dimitriadis, I. Zigurs, and E. Gómez-Sánchez, editors, *Groupware: Design, Implementation, and Use*, volume 4154, pages 54–61. Springer Berlin Heidelberg, 2006.

[4] C. Choquet and S. Iksal. Modeling and constructing usage traces of a learning activity: A language approach for reengineering a cehl (in french). 2007.

[5] C. A. Collazos, L. A. Guerrero, J. A. Pino, S. Renzi, J. Klobas, M. Ortega, M. A. Redondo, and C. Bravo. Evaluating collaborative learning processes using system-based measurement. 18, 2007.

[6] T. Djouad, A. Mille, C. Reffay, and M. Benmohamed. Engineering of activity indicators from modeled traces in a computer environment for human learning (in french). 16(1):103–139, 2009.

[7] A. El Mhouti, M. Erradi, and N. El Makhfi. A multi-agent system of semantic analysis and filtering of modeled traces to calculate interaction indicators favoring collaboration in lms. In *International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, pages 1–7, 2019.

[8] C. Gangloff-Ziegler. The obstacles of collaborative work, steps and organizations. 10(3):95–112, 2009.

[9] S. García-Sastre, A. Martínez-Monés, and Y. Dimitriadis. Detecting collaboration skills to calculate indicators in moodle. In *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*, pages 1–8. ACM, 2017.

[10] É. Gendron. *A conceptual framework for the development of collaboration indicators from activity traces (In French)*. PhD thesis, Claude Bernard University- Lyon I, 2010.

[11] F. Henri. Collaborative learning in virtual mode (fr). 2002.

[12] ICALT. Proceedings of the ieee international conference on advanced learning technologies. 2004.

[13] D. Laurillard, O. Martin, W. Barbara, and H. Ulrich. *Implementing Technology-Enhanced Learning*. 2009.

[14] I. Matazi, A. Bennane, R. Messoussi, R. Touahni, I. Oumaira, and R. Korchiyne. Multi-agent system based on fuzzy logic for e-learning collaborative system. In *International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, pages 1–7, 2018.

[15] D. P. T. Ngoc. Specification and design of analysis services for the use of a computer environment for human learning (in french). Technical report, 2004.

[16] D. Nguyen, S. Yingchareonthawornchai, V. Tekken Valapil, and S. S. Kulkarni. Precision, recall, and sensitivity of monitoring partially synchronous distributed programs. *Distributed Computing*, 34(4):319–348, 2021.

[17] S. PraharaJ, M. Scheffel, and M. et al. Schmitz. Towards automatic collaboration analytics for group speech data using learning analytics. *Sensors*, 21(9):3156, 2021.

[18] G. M. Rafique and M. N. A. Khan. Integrating learning analytics and collaborative learning for improving student's academic performance. *International Journal of Advanced Computer Science and Applications*, 12(12), 2021.

[19] N. Randriamalaka, S. Iksal, and C. Choquet. Elicitation of indicators for the re-engineering of educational scenarios: A trace-based approach using utl (in french). 2008.

[20] S. A. Salloum, M. Al-Emran, K. Shaalan, and A. Tarhini. Factors affecting the e-learning acceptance: A case study from uae. *Education and Information Technologies*, 2019.

[21] L. S. Settouti. Trace-based systems for human learning (in french). 2006.

[22] M. Vásquez-Bermúdez, J. Hidalgo-Larrea, F. Orozco Lara, and S. Segura Santana. Effectiveness of monitoring indicators in the architecture of a collaborative system. In *Technologies and Innovation*, pages 191–202. Springer, Cham, 2022.

IoT Applications in the Education Sector: Architectures, Challenges, and Emerging Paradigms

Hicham Medkour*, Mawloud Belabbas , Bachir Rahmi and Kada Becharef

Educative Technology Division, National Institute for Research in Education
Oued Roumane, El-Achour, Algiers-Algeria
**Corresponding author: hicham.medkour@inre.dz*

Abstract

The rapid integration of Internet of Things (IoT) technology in educational environments is revolutionizing traditional pedagogies and institutional operations. This paper explores IoT's role in augmenting educational delivery, enhancing resource management, and enabling personalized learning through interconnected sensor-based infrastructures. It critically evaluates real-world deployments, security and privacy implications, and future prospects within the smart education paradigm. A multi-layered IoT architecture is proposed, and recommendations for sustainable adoption are discussed.

Keywords: IA, RNN.Internet of Things, Education, Smart Learning, Adaptive Systems, Ubiquitous Learning, Edge Computing, Data Privacy.

1 Introduction

The Internet of Things (IoT) is experiencing growing adoption in the education sector, transforming traditional learning environments into intelligent, interactive, and personalized ecosystems. Educational IoT relies on an interconnected network of sensors, smart devices, and software platforms that collect and process real-time data to optimize teaching and administrative processes. In a global context marked by inequalities in access to education and accelerated digitalization, leveraging these technologies represents a major opportunity to foster inclusivity, learner motivation, and academic performance. Recent studies have explored the potential of IoT in education [1]. Among them, [2] demonstrates how a pilot project at a Malaysian university led to a 23 percent improvement in energy efficiency while enhancing student engagement through behavior-tracking sensors. Other studies [3], [4], [5] highlight the emergence of digital twins, artificial emotional intelligence, and intelligent tutoring systems as drivers of pedagogical transformation. However, despite these advances, large-scale deployments of educational IoT remain limited and often experimental. This study identifies a scientific gap in the systemic understanding of the conditions for success, technical, ethical, and pedagogical challenges, and the evaluation criteria applicable to IoT projects in higher education. The central issue revolves around the sustainable, inclusive, and ethical integration of IoT technologies in educational institutions: How can we design and evaluate an educational IoT architecture that is both efficient, ethical, and adaptable to diverse pedagogical contexts? To address this issue, an analytical and comparative approach has been adopted. This work is based on a structured review of scientific literature, a critical analysis of international case studies, and the application of multi-dimensional evaluation frameworks. Special attention is given to issues of security, data governance, standardization, and scalability in resource-constrained contexts. The primary objective of this study is to provide an in-depth synthesis of IoT architectures, applications, and emerging trends in education, while identifying the technical, ethical, and organizational barriers that need to be overcome. Through this approach, the study aims to enlighten academic decision-makers, instructional engineers, and researchers on best practices for the design, deployment, and evaluation of connected educational solutions.

2 IoT Architecture for Smart Education

2.1 Layered Design

The architecture of the Internet of Things (IoT) in educational environments is commonly structured into a layered model to streamline data flow and system interaction. This model generally consists of three core layers: the perception layer, the network layer, and the application layer [6].

2.1.1 Perception Layer

The perception layer serves as the foundational component of the IoT architecture. It includes various smart sensing devices such as Near Field Communication (NFC) tags, Radio Frequency Identification (RFID) systems, sensors, and cameras. These devices are responsible for gathering real-time data on a wide range of educational metrics, including student attendance, physical movement, classroom environmental conditions (like temperature and light), and device usage patterns. By collecting this data at the source, the perception layer enables real-time monitoring and situational awareness within the educational context. This visibility is instrumental in supporting responsive and adaptive educational services that align with learners' needs.

2.1.2 Network Layer

Next, the network layer is responsible for the secure and efficient transmission of the collected data from the perception layer to the application layer. This layer utilizes multiple communication protocols, including Zigbee, Wi-Fi, 5G, and LoRaWAN, depending on the specific network requirements and constraints of the educational institution. The network layer not only ensures data transfer but also addresses critical aspects such as latency, data packet loss, and bandwidth optimization. Secure communication is prioritized through encryption and tunneling techniques, minimizing the risks associated with data breaches or interception.

2.1.3 Application Layer

At the top of the stack, the application layer translates raw data into actionable insights tailored for educators, administrative staff, and learners. This layer often employs edge computing or cloud-based analytics to process and visualize data in user-friendly formats. Integration with existing Learning Management Systems (LMS) is common, enabling a comprehensive digital learning ecosystem where decision-making and educational customization are driven by real-time insights. Dashboards, reporting tools, and AI-driven recommendation engines fall under this layer, offering tailored feedback to improve both teaching and learning processes.

2.2 Interoperability and Middleware

Given the diverse and often incompatible nature of IoT devices and systems used in education, ensuring interoperability is a significant challenge. Middleware platforms such as FIWARE and Kaa play a pivotal role in addressing this issue [7]. These platforms act as intermediaries that harmonize communication between heterogeneous devices and educational software applications. Middleware provides standardized interfaces and APIs, which enable developers and administrators to integrate new hardware and software without disrupting existing systems.

In addition to facilitating interoperability, middleware solutions enable context-awareness by understanding and adapting to the educational environment. For instance, middleware can interpret contextual signals like user behavior patterns or environmental changes, helping the system respond dynamically to varying scenarios. Furthermore, orchestration capabilities embedded in middleware platforms allow for the automated coordination of processes, including device synchronization, data fusion, and event-driven responses. As a result, middleware enhances the flexibility, scalability, and reliability of IoT implementations in educational settings, paving the way for seamless user experiences and efficient system performance.

3 Key Applications of IoT in Education

3.1 Smart Classrooms

Smart classrooms represent one of the most tangible and transformative applications of IoT in education. These environments utilize a range of interconnected devices such as environmental sensors, AI-powered dashboards, interactive displays, and intelligent control systems for lighting and air conditioning. By continuously monitoring variables such as room temperature, humidity, noise levels, and lighting, these systems help maintain optimal learning conditions.

Moreover, AI-driven analytics tools provide instructors with real-time insights into student engagement and classroom dynamics. Teachers can adapt their instructional strategies on the fly—modifying

pacing, introducing new materials, or adjusting classroom configurations based on data analytics [8]. Ultimately, smart classrooms enhance interactivity, engagement, and learner outcomes by creating an environment that is both responsive and student-centered.

3.2 Attendance and Identity Verification

Traditional methods of tracking student attendance—manual roll calls or sign-in sheets are timeconsuming and prone to errors. IoT technologies like RFID and biometric authentication systems revolutionize this process by automating attendance tracking. Students equipped with RFID-enabled ID cards or biometric markers (e.g., fingerprint or facial recognition) are identified upon entering the classroom, and their presence is recorded in real-time [9].

This automation streamlines administrative tasks and allows teachers to focus on instructional duties. Additionally, it enhances data accuracy, supports behavioral analytics, and contributes to the development of personalized educational pathways. Integration with centralized school databases ensures seamless updating of attendance records and the generation of performance and behavior reports for students and parents.

3.3 Adaptive Learning and Wearables

Wearable IoT devices, such as smartwatches, biometric bands, and augmented reality (AR) headsets, are increasingly used to monitor learners' physiological and emotional states. These devices can track parameters such as heart rate variability, galvanic skin response, and motion patterns to infer cognitive load and emotional engagement [10].

By feeding this data into adaptive learning systems, educational platforms can dynamically adjust content difficulty, format, and delivery methods to match each learner's needs and current state. For example, if a student's stress indicators are elevated, the system might recommend a break or switch to a less cognitively demanding task. This personalization fosters more effective learning and helps mitigate stress and burnout. Teachers also benefit from detailed analytics on student engagement trends, enabling timely interventions and improved learner support.

3.4 Facility and Asset Management

Educational institutions manage a wide range of physical assets—from classroom equipment to campus infrastructure. IoT sensors embedded in furniture, audio-visual (AV) equipment, and utility systems can provide continuous updates on usage patterns, operational status, and maintenance needs [11].

Real-time monitoring supports efficient allocation of resources and prevents downtime by enabling predictive maintenance. For instance, a sensor-equipped projector may notify administrators of a potential malfunction before it occurs, allowing for timely intervention. Furthermore, energy consumption data gathered from HVAC systems or lighting fixtures can inform sustainability strategies, leading to reduced operational costs and environmental impact.

3.5 Inclusive and Remote Learning

IoT technologies are instrumental in promoting inclusivity and accessibility in education. Assistive devices such as Braille-enabled e-readers, hearing aids linked to classroom audio systems, and voicecontrolled learning applications ensure that students with disabilities have equitable access to educational content [12].

Additionally, IoT-enabled remote learning solutions provide consistent and immersive experiences for students learning outside the traditional classroom. Smart conferencing devices, AI-powered tutoring systems, and collaborative platforms facilitate engagement and maintain the continuity of instruction. These tools became particularly vital during the COVID-19 pandemic and continue to support hybrid and distance learning models.

4 Security and Privacy Considerations

While IoT offers substantial benefits to the educational sector, it also introduces significant ethical, privacy, and security challenges that must be addressed to ensure safe and responsible deployment.

4.1 Data Privacy Risks

The use of IoT in education involves the collection of sensitive student data, including location information, biometric identifiers, academic performance, and behavioral patterns. These data points are vulnerable to unauthorized access, theft, or misuse if not properly safeguarded. To address these concerns, institutions must adhere to data protection regulations such as the General Data Protection Regulation (GDPR) in Europe and the Family Educational Rights and Privacy Act (FERPA) in the United States [13].

Data anonymization, encryption, and strict access control mechanisms are essential to protect personal information. Additionally, transparency in data collection practices and obtaining informed consent from students and guardians are critical steps toward ethical IoT usage.

4.2 Attack Surfaces

The proliferation of interconnected devices in educational environments increases the potential attack surface for malicious actors. Common threats include Distributed Denial-of-Service (DDoS) attacks, spoofing, unauthorized access, and malware infiltration. These threats are often exacerbated by inadequate encryption, unpatched firmware, and the use of outdated devices [14].

Attackers can exploit vulnerabilities to disrupt learning activities, steal confidential data, or gain control over institutional infrastructure. As such, a proactive approach to cybersecurity—including regular updates, threat monitoring, and penetration testing—is vital to safeguarding IoT systems.

4.3 Mitigation Strategies

To address these vulnerabilities, educational institutions can implement a range of mitigation strategies aimed at enhancing security and trust in IoT deployments. Key approaches include:

- Lightweight Cryptography: Suitable for resource-constrained IoT devices, lightweight cryptographic algorithms ensure data confidentiality and integrity without overloading system resources.

- Network Segmentation via SDN: Software Defined Networking (SDN) allows for dynamic network segmentation, reducing the spread of attacks and enabling granular access control.

- Blockchain-Based Audit Trails: Blockchain technology can be employed to create immutable logs of data access and transactions, promoting transparency and accountability [15].

By adopting a security-by-design philosophy and continuously evaluating emerging threats, institutions can harness the full potential of IoT while maintaining ethical standards and legal compliance in educational contexts.

5 Critical Evaluation of Case Studies

A comparative analysis of IoT deployments across various universities reveals several key success factors that shape the effectiveness of these systems.

Firstly, the integration of **Localized Edge Computing** significantly reduces latency, ensuring real-time responsiveness in applications such as behavioral feedback systems and adaptive learning tools. Rather than sending all data to a centralized cloud, local edge devices process and respond to data closer to the source, minimizing delays and conserving bandwidth.

Secondly, **faculty training in IoT ethics and data governance emerges** as a critical component. Successful IoT adoption hinges not only on technological readiness but also on the awareness and ethical responsibility of educators. Institutions that incorporate data governance training and ethics workshops are better equipped to manage privacy concerns and ensure equitable student treatment.

Thirdly, universities are increasingly implementing **hybrid infrastructures** that combine cloud and fog computing. This architecture helps balance the scalability and computational power of cloud platforms with the responsiveness and localized processing offered by fog nodes. Such systems allow for seamless data management while upholding student privacy and adapting to network constraints.

In [15], a notable case study conducted at a Malaysian university showcased the practical benefits of such strategies. The pilot project implemented sensor-based behavioral feedback mechanisms to encourage energy-saving behaviors among students. As a result, energy efficiency improved by 23 percent. Furthermore, the interactive feedback loop—facilitated by IoT devices and dashboards—boosted student

participation, underscoring how well-designed IoT systems can influence not only operational metrics but also learner engagement and institutional culture.

6 Emerging Trends and Out-of-the-Box Applications

6.1 Cognitive IoT and AI

The convergence of Artificial Intelligence (AI) and IoT, commonly referred to as **AIoT**, is driving innovation in personalized learning. Intelligent tutoring systems now incorporate AI algorithms—particularly reinforcement learning—to tailor quiz difficulty and content delivery based on real-time assessments of student performance and behavior. These systems dynamically adapt to individual learning curves, offering more precise and motivating educational experiences.

6.2 Digital Twins for Learning

A particularly novel development is the use of **digital twins**—virtual counterparts of physical learning environments. These digital replicas, enabled by IoT and immersive technologies such as Augmented Reality (AR) and Virtual Reality (VR), allow remote learners to engage with classroom environments in real-time. Students can interact with virtual lab equipment, observe real-world classroom dynamics, or even participate in collaborative activities via avatars. This concept redefines remote learning by making it more experiential and presence-driven [2].

6.3 Gamification and Emotional AI

Another frontier in educational IoT is the blending of **gamification** techniques with **emotional AI**. IoT devices such as smart cameras or wearable sensors can detect emotional cues like facial expressions, vocal tone, or physiological stress markers. These inputs feed into gamified learning systems that adjust tasks, difficulty levels, or feedback styles to maintain engagement and motivation. For instance, if a student appears frustrated, the system might reduce task complexity or offer encouraging prompts. The combination of real-time emotion recognition and motivational design principles fosters deeper learner immersion and satisfaction [3].

7 Methodological Framework for IoT Evaluation in Education

A robust evaluation of IoT deployments in education requires a multi-dimensional framework that integrates both technical performance and educational outcomes. Four primary metric categories are recommended:

• Technical KPIs: Metrics such as latency, throughput, and device uptime assess the operational efficiency of IoT systems.

• Pedagogical Metrics: These include indicators like student engagement rates, retention, learning gains, and academic performance improvements.

• Usability Metrics: These gauge system intuitiveness, ease of navigation, and user satisfaction, particularly among faculty and students.

• Socio-Ethical Metrics: Inclusivity, equity, transparency in data use, and adherence to ethical standards fall under this category.

To gather these metrics effectively, a mixed-method research approach is advised. Quantitative data from system logs and user analytics can be complemented by qualitative feedback collected via surveys, interviews, and classroom observations. Longitudinal studies, in particular, are crucial to understanding the sustained impact of IoT systems on learning outcomes and educational equity [4].

8 Challenges and Future Directions

8.1 Scalability vs. Affordability

One of the primary barriers to widespread IoT adoption in education is the cost of deployment—particularly in low-resource settings. To address this, institutions are turning to **cost-effective, open-source hardware platforms** such as Raspberry Pi, Arduino, and ESP32. These devices support

essential IoT functionalities while minimizing financial strain. Moreover, optimized firmware and modular architectures can extend device lifespans and reduce maintenance requirements.

8.2 Standardization

Another significant challenge is the **lack of unified standards** for educational IoT systems. This hampers interoperability, making it difficult for institutions to integrate heterogeneous devices and platforms. Initiatives such as **oneM2M** and **IEEE P2413** are working toward global IoT standards, but adaptation for academic use remains in early stages [16]. Collaborative efforts among universities, industry, and standards bodies are necessary to accelerate this process and promote compatibility

8.3 Ethical Pedagogical Design

Finally, as educational technologies become increasingly data-driven, there is a growing need for **ethical pedagogical frameworks**. IoT-based learning systems must not only be efficient but also aligned with human-centered values. This includes ensuring informed consent, fostering inclusive design, andresisting over-surveillance. Educators and technologists should co-create curricula and systems that prioritize students' well-being, autonomy, and equitable access to learning opportunities.

9 Conclusion

The Internet of Things holds transformative potential for the education sector by enabling smart, responsive, and personalized learning environments. However, its integration is not without challenges. Success depends on well-designed system architecture, robust ethical governance, and collaborative innovation among educators, technologists, and policymakers. As educational institutions embrace IoT, they must remain vigilant about issues of equity, privacy, and long-term sustainability. Future research should focus on human-centered design, ethical standards, and scalable frameworks to ensure IoT serves as a tool for inclusive and impactful education worldwide.

References

[1] A. M. Alghamdi, "IoT-Based Smart Classrooms: A Systematic Review," IEEE Access, vol. 10, pp. 95011–95030, 2022.
[2] F. Rahim et al., "Smart Campus Pilot in Malaysia: An IoT Energy Optimization Study," IEEE Access, vol. 8, pp. 115624–115638, 2020.
[3] H. Li and W. Wang, "Digital Twins and IoT in Immersive Education," IEEE Transactions on Industrial Informatics, vol. 17, no. 6, pp. 4182–4191, Jun. 2021.
[4] D. G. López et al., "Gamified Learning with IoT Feedback Systems," IEEE Transactions on Learning Technologies, vol. 14, no. 1, pp. 88–98, Jan.–Mar. 2021.
[5] S. A. Kazi et al., "Mixed-Method Evaluation of IoT-Enabled Learning Spaces," Computers and Education, vol. 161, p. 104064, 2021.
[7] M. S. Hossain et al., "IoT in Education: A Framework for Smart Learning Environment," IEEE Internet of Things Journal, vol. 7, no. 8, pp. 7103–7112, Aug. 2020.
[8] K. P. Tripathi et al., "Middleware for IoT in Education: A Survey," IEEE Communications Surveys and Tutorials, vol. 23, no. 2, pp. 1431–1457, 2021.
[9] H. A. Khattak et al., "Smart Learning Environments Using IoT and Context-Aware Systems," Computers in Human Behavior, vol. 112, p. 106481, Jan. 2020.
[10] N. D. Patel et al., "RFID-Based Attendance Monitoring in Higher Education," International Journal of Engineering Research and Technology, vol. 9, no. 5, pp. 1023–1027, 2021.
[11] J. W. Kim et al., "Wearable IoT in Education: A Smart Glove for Learning Sign Language," Sensors, vol. 20, no. 11, p. 3132, 2020.
[12] Y. Zhang and X. Zhao, "IoT Asset Management in Smart Campuses," IEEE Sensors Journal, vol. 21, no. 15, pp. 16851–16860, 2021.
[13] B. L. Perry et al., "IoT for Inclusive Learning: A Review," IEEE Transactions on Learning Technologies, vol. 13, no. 4, pp. 723–734, Oct. 2020.
[14] G. Spanakis et al., "Privacy Challenges in IoT-Enabled Education," IEEE Security and Privacy, vol.

18, no. 6, pp. 33–41, 2021
[15] A. Shahrbabaki et al., "Security in IoT for Education: A Review of Threats and Mitigations," IEEE Communications Magazine, vol. 59, no. 2, pp. 76–81, 2021.
[16] A. M. Rahmani et al., "Security Solutions for Smart Learning Systems," IEEE Systems Journal, vol. 14, no. 3, pp. 3676–3687, 2020.
[16] M. R. Islam et al., "IoT Standards and Frameworks for Education," IEEE Standards Magazine, vol. 3, no. 3, pp. 24–31, Sept. 2021.

A Comprehensive Review of Knowledge Graph Integration in Large Language Models for Trust: Challenges and Future Directions

Touameur Ouissem[1] **and Harrag Fouzi**[2]

[1] ***Department of Computer Science, University of Farhat Abbas Setif,*** *ouissem.touameur@univ-setif.dz*

[2] ***Department of Computer Science, University of Farhat Abbas Setif,*** *fouzi.harrag@univ-setif.dz*

Abstract

Large Language Models (LLMs) have transformed the landscape of natural language processing, enabling significant application advancements. However, LLMs' inherent complexity and opacity raise critical concerns regarding their trustworthiness, including bias, misinformation, and lack of interpretability. This paper presents a comprehensive state-of-the-art review on the integration of knowledge graphs (KGs) to enhance trust in LLMs. We explore how KGs can serve as structured frameworks for contextualizing the knowledge embedded in LLMs, providing a means to verify and validate their outputs against reliable sources of information. The review highlights various methodologies that leverage KGs to address trust-related challenges in LLMs, including mechanisms for improving factual consistency, reducing biases, and enhancing interpretability. Additionally, we analyze recent advancements and empirical studies that demonstrate the efficacy of knowledge graph integration in fostering transparency and reliability in LLM applications. By synthesizing the current landscape of research, this paper aims to identify future directions and key challenges in developing trust-aware systems that utilize the synergistic potential of LLMs and knowledge graphs.

Keywords: LLMs, Trustworthiness, Knowledge graph, NLP, reliability.

1 Introduction

The rapid advancement of Large Language Models (LLMs), such as OpenAI's GPT-3 and Google's BERT, has revolutionized the field of natural language processing (NLP) by enabling sophisticated text generation, comprehension, and interaction capabilities [3, 6]. Despite their impressive performance, significant concerns regarding the trustworthiness of these models have emerged. Issues such as inherent biases, the propensity to generate misleading or factually incorrect information, and the lack of interpretability challenge their deployment in sensitive applications [2, 9].

Trust is essential in AI systems, particularly in applications involving critical decision-making, such as healthcare, finance, and law. Users need to rely on the outputs of these models, necessitating a framework that can ensure reliability and transparency. Knowledge graphs (KGs), which represent knowledge in a structured format through entities and their relationships, offer a promising solution for enhancing the trustworthiness of LLMs. KGs can provide contextual information, facilitate the verification of facts, and support the interpretation of model outputs, thereby addressing some of the primary trust-related concerns associated with LLMs [16].

Recent research has explored various ways to integrate KGs with LLMs, highlighting their potential to mitigate issues such as misinformation and bias by providing a robust knowledge base against which LLM outputs can be validated [22]. For instance, KGs can serve as sources of grounding information, enabling LLMs to generate more contextually accurate responses [25]. Furthermore, KGs can enhance model interpretability by elucidating the reasoning behind generated outputs and demonstrating the relationships between concepts [8].

This paper aims to provide a comprehensive review of the current state of research on trust in LLMs, focusing specifically on the integration of knowledge graphs. We will examine the methodologies employed, their effectiveness in enhancing trust, and the challenges that remain in this evolving field. The paper is structured as follows: first, we explore the concept of trust in LLMs, followed by an overview of approaches using knowledge graphs within LLMs. Next, we review related work, present a discussion on key insights, and outline challenges and future directions. Finally, we conclude with a summary of the findings.

2 Understanding Trust in LLMs

As large language models (LLMs) such as GPT, BERT, and T5 become increasingly embedded in applications affecting daily life, fostering user trust in these models has become crucial. Trust in LLMs rests on several key factors—transparency, reliability, accountability, and interpretability. Each of these elements contributes uniquely to how users perceive, rely on, and interact with LLMs.

2.1 Transparency

Transparency refers to the model's ability to reveal its internal workings, data sources, and decision-making processes. In LLMs, transparency encompasses both the interpretability of the model architecture and the visibility of the training data. Transparent LLMs allow users and developers to understand the source of the information, the decisions the model makes, and its reasoning process. For example, the work by Piktus et al. [17] on explaining machine learning classifiers provides insight into how interpretability methods can make black-box models more transparent by offering understandable explanations for predictions. Efforts toward transparency often involve open-sourcing datasets and providing documentation, such as Google's Model Cards [14], which standardize model reporting to include information about intended use cases, biases, and limitations.

2.2 Reliability

Reliability reflects the consistency and accuracy of an LLM's performance across various tasks and contexts. Users are more likely to trust models that yield accurate and reproducible results, especially when deployed in high-stakes areas such as healthcare or legal domains. For instance, studies on adversarial robustness by Jia & Liang [21] illustrate that LLMs may be vulnerable to small changes in input data, which can lead to unreliable or erroneous outputs. Addressing these issues, some approaches focus on adversarial training and robustness testing to ensure that LLMs perform reliably under diverse conditions [23]. Reliable models thus bolster user trust by providing stable and dependable outcomes.

2.3 Accountability

Accountability in LLMs relates to the capacity for tracing and attributing the model's outputs to its developers or the data it was trained on. Given the wide-reaching influence of LLM-generated content, accountability becomes essential for maintaining public trust and mitigating risks associated with biased or harmful outputs. Studies on responsible AI, such as the work by Binns [20], emphasize the need for accountability mechanisms to ensure that LLMs can be held answerable for their outputs, particularly when they affect individuals or communities. Additionally, frameworks like "explainable artificial intelligence" (XAI) aim to create models that produce both understandable and accountable outputs, contributing to responsible and trustable AI [4].

2.4 Interpretability

Interpretability is crucial for understanding and trusting LLMs, as it determines how easily a human can make sense of a model's predictions. In natural language processing, interpretability approaches such as attention visualization [7] provide insights into which parts of the input data an LLM considers most relevant for making predictions. Techniques like Local Interpretable Model-agnostic Explanations (LIME) [27] further support interpretability by explaining individual predictions and model behavior. Interpretable models empower users to verify the model's reasoning and assess its appropriateness in specific contexts, thus enhancing trust.

These four factors—transparency, reliability, accountability, and interpretability—are interdependent and collectively contribute to building and maintaining user trust in LLMs. Ongoing research in each area aims to advance these dimensions of trustworthiness, bringing us closer to AI systems that users feel confident in adopting and integrating into decision-making processes.

3 Trust in LLMs with Knowledge Graph

3.1 Definition of Knowledge Graphs

Knowledge Graphs (KGs) are structured representations of real-world entities, their attributes, and the relationships between them. Typically organized in a graph format, KGs enable machines to model and reason about domain-specific knowledge by connecting data points through edges representing meaningful relations [5]. They consist of nodes (entities) and edges (relationships) that create a network of interconnected information, thereby facilitating structured queries and efficient knowledge retrieval [18]. KGs are increasingly used with large language models (LLMs) to provide a structured and interpretable backbone for enhancing reasoning and factual correctness.

3.2 Enhancing Trust with Knowledge Graphs

Integrating KGs with LLMs can significantly improve transparency and interpretability by providing factual grounding and context for LLM responses. KGs serve as external sources of structured information that can back model predictions, enhancing user trust. By linking facts to entities in a KG, LLMs can provide evidence for their responses, making the output more transparent and understandable for users [26]. For instance, grounding an LLM's responses with Wikipedia-derived KGs allows for entity resolution, where references in the model's output can directly link back to a KG entity, thus verifying the source of the information [11]. This grounding ensures that users can trace the origin of certain statements, thus increasing the system's transparency.

3.3 Examples and Techniques

To enhance trust in large language models (LLMs) using knowledge graphs (KGs), several techniques and examples can be employed:

1. **KG-Enhanced LLM Interpretability**: By integrating KGs into the inference process of LLMs, researchers can improve the interpretability of the model's outputs. For instance, when an LLM generates a response, the relevant facts from the KG can be highlighted to show the basis for the generated information. This transparency helps users understand how the model arrived at its conclusions, thereby increasing trust.

2. **Factual Verification**: Techniques such as LLM-facteval can be used to automatically generate probing questions from KGs. These questions can then be used to evaluate the factual knowledge stored in LLMs. By systematically assessing the accuracy of the information provided by LLMs against a trusted KG, users can gain confidence in the model's reliability.

3. **Grounding Responses in KGs**: Approaches like KagNet and QA-GNN ground the results generated by LLMs at each reasoning step using KGs. This means that the reasoning process is made explicit by linking the generated outputs to specific entities and relationships in the KG. Such grounding provides a clear rationale for the model's responses, enhancing user trust.

4. **Knowledge Graph-Based Probing**: Tools like BioLAMA and MedLAMA utilize domain-specific KGs to probe LLMs for factual knowledge in specialized fields, such as medicine. By evaluating the model's performance against a trusted medical KG, these techniques can help ensure that the LLM provides accurate and reliable information in critical applications.

5. **Instruction-Tuning with KGs**: Integrating KGs into the training objectives of LLMs can help improve their factual accuracy. For example, instruction-tuning methods can be employed where LLMs are trained to generate responses that align with the structured knowledge in KGs. This technique can help reduce the occurrence of hallucinations and improve the overall trustworthiness of the model [16].

4 Related Works

The integration of Large Language Models (LLMs) with Knowledge Graphs (KGs) has become a significant area of research, focusing on enhancing the capabilities of LLMs in terms of factual accuracy,

reasoning, and trustworthiness. This section reviews notable works in this domain and proposes a taxonomy based on their contributions.

We propose a taxonomy to categorize integration approaches of LLMs and KGs, comprising three main dimensions: Type of Integration, Focus Area, and Application Domain. This structured framework helps in understanding the diverse methodologies within LLM and KG integration.

4.1 Performance Evaluation of LLMs

Yang et al. [24] discuss the integration of knowledge graphs (KGs) with large language models (LLMs) to improve their ability to recall and utilize factual knowledge. The authors highlight that while LLMs like ChatGPT exhibit impressive conversational abilities, they struggle with generating knowledge-grounded content due to limitations in factual recall. To address this, the paper reviews existing methods for enhancing pre-trained language models (PLMs) with KGs and proposes the development of knowledge graph-enhanced large language models (KGLLMs).

Hou et al. [10] examine the limitations of LLMs in biomedical contexts, where accuracy is crucial. The methodology involved conducting experiments where ChatGPT answered questions from the "Alternative Medicine" sub-category of Yahoo! Answers, while BKGs were queried for relevant knowledge records. Additionally, a prediction scenario was created to evaluate the models' abilities to suggest potential drug and dietary supplement repurposing candidates for Alzheimer's Disease (AD). The results indicated that while ChatGPT (especially GPT-4) outperformed earlier versions and provided existing information effectively, BKGs demonstrated higher reliability and accuracy. ChatGPT struggled with novel discoveries and reasoning, particularly in establishing structured links between entities.

4.2 Trustworthiness and Credibility in LLM Outputs

Dr. Carlo Lipizzi [13] presents a novel approach to evaluating the trustworthiness of Large Language Models (LLMs) by integrating knowledge graphs, RDF triplets, and a human-in-the-loop system. It emphasizes the importance of accurately representing domain-specific knowledge through knowledge graphs and involves subject matter experts (SMEs) to validate this representation and assess the compatibility of LLM outputs with established knowledge. By focusing on quantitative measures of trustworthiness, the proposed system aims to address the growing concerns regarding the reliability of LLMs, particularly in critical applications. The paper highlights the innovative nature of this approach while acknowledging challenges such as subjectivity, scalability, and the complexity of knowledge representation.

Zhang et al. [28] investigate how KGs can enhance the factual recall of LLMs, addressing challenges in integrating various data representations to improve output accuracy.

4.3 Type of Integration

The integration strategies can be categorized based on their timing and methodology:

- **Before-Training Enhancement:** Zafar et al. [26] present a novel architecture that integrates large language models (LLMs), knowledge graphs (KGs), and role-based access control (RBAC) to enhance the capabilities of conversational AI systems. It highlights the importance of combining the linguistic proficiency of LLMs with the structured knowledge representation of KGs to address challenges such as explainability, data privacy, and contextual accuracy. The architecture aims to foster user trust and ensure the ethical use of AI technologies. Additionally, the paper introduces LLMXplorer, a comprehensive tool for evaluating various LLMs, which contributes to transparency and informed decision-making in the deployment of conversational AI.

- **Post-Training Enhancement:** Li et al. [12] introduce XTRUST, the first comprehensive benchmark designed to evaluate the multilingual trustworthiness of large language models (LLMs). The authors highlight the remarkable capabilities of LLMs in various natural language processing (NLP) tasks and emphasize the growing concern regarding their trustworthiness, especially in sensitive fields such as healthcare and finance. XTRUST encompasses a wide range of topics, including illegal activities, hallucination, out-of-distribution robustness, mental and physical health, toxicity, fairness, misinformation, privacy, and machine ethics, across ten different languages. The paper presents an empirical evaluation of five widely used LLMs, revealing that many struggle with low-resource languages like Arabic and Russian, indicating significant room for improvement in their multilingual trustworthiness.

Alghamdi et al. [1] develop AraTrust, a trustworthiness benchmark for Arabic LLMs. The paper introduces AraTrust, the first comprehensive benchmark designed to assess the trustworthiness of Large Language Models (LLMs) specifically for the Arabic language. It comprises 516 human-written multiple-choice questions that cover eight critical categories of trustworthiness: truthfulness, ethics, physical health, mental health, unfairness, illegal activities, privacy, and offensive language. The authors highlight the inadequacies of existing English-centric benchmarks, which fail to address the unique cultural and contextual factors relevant to Arabic users. By providing a culturally aligned and automated assessment framework, AraTrust aims to enhance the safety and reliability of Arabic LLMs and promote further research in this area. The findings indicate that while proprietary models like GPT-4 perform well, many open-source models struggle to meet the benchmark's standards, underscoring the need for improved trustworthiness in Arabic LLMs.

- **Real-time Enhancement:** Large language models (LLMs) have achieved impressive results across various natural language processing tasks. However, once deployed, LLMs interact with users who possess personalized factual knowledge, which is reflected in their interactions. To enhance the user experience, it is crucial to implement real-time model personalization that allows LLMs to adapt user-specific knowledge based on feedback received during these interactions.

  Current methods primarily rely on back-propagation to fine-tune model parameters, leading to significant computational and memory overhead. Additionally, these methods often lack interpretability, which can negatively affect model performance as users accumulate personalized knowledge over time.

  To tackle these challenges, we introduce Knowledge Graph Tuning (KGT), a novel approach that utilizes knowledge graphs (KGs) to personalize LLMs. KGT extracts personalized factual knowledge triples from user queries and feedback, optimizing the KGs without altering the LLM parameters. This method enhances computational and memory efficiency by circumventing back-propagation while ensuring interpretability by making KG adjustments understandable to humans.

  Experiments with state-of-the-art LLMs, including GPT-2, Llama2, and Llama3, demonstrate that KGT significantly enhances personalization performance while reducing latency and GPU memory usage. In conclusion, KGT presents a promising solution for effective, efficient, and interpretable real-time LLM personalization during user interactions [19].

4.4 Focus Area

The focus of these studies can also be categorized based on their objectives:

- **Factual Recall:** Hou et al. [10] reveal strengths and weaknesses in information retrieval for LLMs.
- **Trustworthiness Assessment:** Lipizzi [13] emphasizes the importance of validation mechanisms for LLM outputs.
- **Model Personalization:** Sun et al. [19] introduces KGT to enhance real-time model personalization using KGs.

4.5 Application Domain

The studies vary in their application domains:

- **General Knowledge:** Zafar et al. [26] and Kommineni et al. explore integration methodologies to enhance conversational AI systems.
- **Biomedical Research:** Hou et al. [10] address factual recall challenges in biomedical contexts.
- **Domain-Specific Applications:** Li et al. [12] and Alghamdi et al. [1] focus on trustworthiness in healthcare and Arabic language contexts.

Future research should develop efficient integration methodologies that enhance factual grounding while considering computational efficiency.

Table 1: Summary of Related Works on LLMs and KGs Integration

Reference	Type of Integration	Focus Area	Key Contributions
Yang et al. [24]	Before-Training	Factual Recall	KG-enhanced LLMs for improved factual recall
Hou et al. [10]	Post-Training	Factual Recall	Comparison of ChatGPT and BKGs in biomedical contexts
Zafar et al. [26]	Before-Training	Trustworthiness	LLMs and KGs integration for conversational AI
Li et al. [12]	Post-Training	Trustworthiness	XTRUST benchmark for multilingual LLMs
Alghamdi et al. [1]	Post-Training	Trustworthiness	AraTrust benchmark for Arabic LLMs
Lipizzi et al. [13]	Real-time	Trustworthiness	Human-in-the-loop for trust assessment in LLMs
Sun et al. [19]	Real-time	Model Personalization	KGT for personalized LLMs using KGs

5 Discussion

The integration of Large Language Models (LLMs) with Knowledge Graphs (KGs) represents a transformative approach to enhancing the capabilities of AI systems, particularly in domains requiring high accuracy and trust. The reviewed methods showcase significant advancements in the factual recall and reasoning abilities of LLMs, highlighting how structured knowledge representations can mitigate the inherent limitations of these models. For instance, studies by Yang et al. [24] and Hou et al. [10] demonstrate that KGs not only improve information retrieval but also facilitate more contextually relevant outputs, particularly in specialized fields such as biomedical research.

Zafar et al. [26] emphasize the comprehensive integration of LLMs and KGs, which enhances explainability and data privacy through robust access control measures, while also allowing for iterative learning and practical applications in real-world scenarios. However, the architecture's generalizability remains limited as it has predominantly been tested within specific contexts, like media and journalism, raising questions about its performance across diverse industries. Similarly, Li et al. [12] present the XTRUST benchmark, addressing the critical gap in multilingual LLM evaluations and underscoring the importance of trustworthiness in sensitive domains such as healthcare and finance. Yet, their evaluation of only five widely used models and a limited scope of languages may restrict the findings' applicability.

Alghamdi et al. [1] highlights the structured representation of knowledge in KGs, enhancing contextual understanding and facilitating reasoning, but they also point out challenges related to data quality,

scalability, and user acceptance, which can undermine trust. Furthermore, approaches focusing on trustworthiness, such as those proposed by Lipizzi et al. [13], underscore the necessity of integrating human expertise and domain-specific knowledge to assess and enhance the reliability of LLMs.

Despite these strengths, challenges persist, including knowledge noise within KGs, which can lead to inaccuracies in model outputs, as noted by Yang et al. This highlights the urgent need for robust filtering mechanisms and dynamic updating processes to ensure that KGs remain relevant and accurate. Moreover, the computational complexity associated with integrating KGs with LLM architectures raises concerns about scalability and real-time application, necessitating more efficient methodologies. Addressing the subjectivity involved in trust assessments is equally critical, as inconsistent evaluations may hinder the applicability of these frameworks across diverse contexts. Therefore, future research should prioritize the development of standardized metrics for trust evaluation, optimized integration algorithms, and cross-domain applications of KG-LLM frameworks. By tackling these limitations, researchers can unlock the full potential of integrating KGs with LLMs, paving the way for more reliable, transparent, and contextually aware AI systems.

6 Challenges and future directions

6.1 Challenges

While integrating Knowledge Graphs (KGs) with Large Language Models (LLMs) offers potential benefits, several significant challenges persist. Knowledge noise within KGs can lead to inaccuracies in LLM outputs, necessitating robust filtering mechanisms [24], while the complexity of integrating KGs with LLM architectures can introduce substantial computational demands [13]. Accurately capturing the intricacies of a domain in a KG is challenging, as nuanced relationships and exceptions may result in oversimplifications or inaccuracies. Moreover, KGs are often incomplete, leading to gaps in the knowledge accessible to LLMs, which can produce misleading outputs. The dynamic nature of knowledge necessitates continuous updating of KGs to avoid outdated conclusions, a resource-intensive process. Integration complexity arises when aligning structured data from KGs with unstructured data processed by LLMs, requiring sophisticated methods for effective utilization. Additionally, scalability issues become apparent as the size and complexity of KGs increase, complicating maintenance and validation processes. Subjectivity in knowledge selection can lead to inconsistencies in representation, and biases in the underlying data can undermine the trustworthiness of LLM outputs. Variability in knowledge quality further exacerbates trust issues, as inaccurate or outdated information can result in erroneous conclusions. Interpretability challenges emerge when attempting to understand how KGs influence LLM decision-making, compounded by limitations in human validation due to the availability of subject matter experts. Furthermore, the computational overhead of querying and analyzing KGs may impact the efficiency of trust assessments, and user skepticism regarding the reliability of KGs can hinder acceptance, particularly when users are unfamiliar with the methodologies employed in their construction. Finally, interoperability issues can complicate the integration of KGs built using different standards and formats, posing challenges for comprehensive trust assessments across various domains. In summary, while the integration of KGs with LLMs holds promise for enhancing trustworthiness, addressing these multifaceted challenges is critical for achieving effective and ethical outcomes.[15, 16, 1]

6.2 Future Directions

Future directions for using Large Language Models (LLMs) in conjunction with Knowledge Graphs (KGs) to enhance trustworthiness can focus on several key areas. Developing methods for creating and maintaining dynamic knowledge graphs that can automatically update in response to new information, research findings, or changes in domain knowledge is essential. This could involve leveraging real-time data sources and machine-learning techniques to ensure that the knowledge graph remains current and relevant. Additionally, improving the integration of LLM outputs with knowledge graphs through advanced natural language processing techniques could enhance the alignment of LLM-generated content with the structured data in KGs, enabling more accurate assessments of trustworthiness based on contextual relevance. Furthermore, exploring automated or semi-automated validation processes for knowledge graphs, potentially using machine learning algorithms to identify inconsistencies or gaps in the knowledge representation, could reduce reliance on human evaluators and enhance scalability. Encouraging collaboration between domain experts, data scientists, and AI researchers is vital to creating more robust knowledge graphs that accurately reflect the complexities of various fields. This interdisciplinary

approach can help ensure that the knowledge represented is comprehensive and trustworthy. Developing user-centric metrics for trust that take into account individual user needs, preferences, and contexts can also enhance the trustworthiness of LLM outputs. Focusing on enhancing the explainability of LLM outputs about the knowledge graph by providing users with clear explanations of how LLM responses are derived from the knowledge graph can build trust through transparency. Moreover, creating cross-domain knowledge graphs that can integrate information from multiple fields allows LLMs to provide more comprehensive and contextually aware responses. This could enhance the trustworthiness of outputs in interdisciplinary applications. Addressing ethical considerations related to trust in LLMs and KGs, including the identification and mitigation of biases in both the knowledge representation and the model outputs, is crucial for broader acceptance. Conducting extensive real-world testing of LLMs combined with knowledge graphs in various applications, such as healthcare, finance, and education, is necessary. Gathering empirical data on their performance and trustworthiness can inform further improvements and refinements. Finally, encouraging community contributions to knowledge graphs, allowing users to add, edit, and validate information, can enhance the richness and accuracy of the knowledge represented, fostering a sense of ownership and trust among users. By pursuing these future directions, the integration of LLMs and knowledge graphs can lead to more reliable, trustworthy, and user-friendly systems that effectively support decision-making across various domains.

7 Conclusion

In this paper, we explored the integration of Large Language Models (LLMs) with Knowledge Graphs (KGs) to enhance the trustworthiness of information generated in various domains. We established that while LLMs demonstrate impressive capabilities in natural language processing tasks, their outputs can be limited by biases and inaccuracies inherent in the training data. By combining LLMs with KGs, we can leverage the structured, semantically rich information contained within knowledge graphs to improve the reliability and contextual relevance of LLM-generated content. Our investigation highlighted several promising future directions, including the dynamic updating of knowledge graphs, the enhancement of natural language processing techniques for better integration, and the development of user-centric trust metrics. Additionally, we emphasized the importance of interdisciplinary collaboration and ethical considerations in the deployment of these integrated systems. Through extensive empirical testing and community engagement, we aim to create more robust and trustworthy systems that effectively support decision-making across various fields. The findings of this study pave the way for future research aimed at bridging the gap between LLMs and KGs, ultimately fostering trust and improving the quality of information accessible to users.

References

[1] Emad A. Alghamdi, Reem I. Masoud, Deema Alnuhait, Afnan Alomairi, Ahmed Ashraf, and Mohamed Zaytoon. Aratrust: An evaluation of trustworthiness for llms in arabic. In *Proceedings of the Arabic Language and AI Conference*, 2023.

[2] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 149–159, 2018.

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, P. Dhariwal, and D. Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

[4] Erik Cambria, Soujanya Poria, Alexander Gelbukh, and Awais Hussain. Xai meets llms: A survey of the relation between explainable ai and large language models. *arXiv preprint arXiv:2407.15248*, 2024.

[5] Xiaojun Chen, Shengbin Jia, and Yang Xiang. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, 141:112948, 2020.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

[7] Kaito Fujiwara, Katsuya Nakamura, Jun Matsui, Tatsuya Matsumoto, and Masahiro Hara. Measuring the interpretability and explainability of model decisions of five large language models. *arXiv preprint*, 2024.

[8] Jorge Garnica, Ana Vega, and Juan Gutiérrez. Knowledge graphs for explainable artificial intelligence: A survey. In *Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2022.

[9] Kurt Holstein, Jennifer Wortman Vaughan, Hal Daumé III, and Mike Dudik. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

[10] Yu Hou, Jeremy Yeung, Hua Xu, Chang Su, Fei Wang, and Rui Zhang. From answers to insights: Unveiling the strengths and limitations of chatgpt and biomedical knowledge graphs. In *Proceedings of the Annual Conference on Artificial Intelligence in Medicine*, 2023.

[11] Tuan Manh Lai. *Knowledge Acquisition for Natural Language Understanding*. PhD thesis, University of Illinois at Urbana-Champaign, 2023.

[12] Yahan Li, Yi Wang, Yi Chang, and Yuan Wu. Xtrust: On the multilingual trustworthiness of large language models. In *Proceedings of the International Conference on Multilingual NLP*, 2023.

[13] Carlo Lipizzi. Tell me the truth: A system to measure the trustworthiness of large language models. In *Proceedings of the International Conference on Trustworthy AI*, 2023.

[14] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 220–229, 2019.

[15] Jeff Z. Pan et al. Large language models and knowledge graphs: Opportunities and challenges. *arXiv preprint arXiv:2308.06374*, 2023.

[16] Shirui Pan, Zhiwei Liu, Wei Zhuang, Rui Yang, Lei Zhang, Jialiang Li, and Haifeng Wang. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[17] Aleksandra Piktus, Yi Wang, Vladimir Karpukhin, Ravi Pasunuru, Thibaut Mialon, Nisan Stiennon, Wen-tau Yih, Laleh Koc, and Matthew Efron. The roots search tool: Data transparency for llms. *arXiv preprint arXiv:2302.14035*, 2023.

[18] Ridho Reinanda, Edgar Meij, and Maarten de Rijke. Knowledge graphs: An information retrieval perspective. *Foundations and Trends® in Information Retrieval*, 14(4):289–444, 2020.

[19] Jingwei Sun, Zhixu Du, and Yiran Chen. Knowledge graph tuning: Real-time large language model personalization based on human feedback. *arXiv preprint arXiv:2405.19686*, 2024.

[20] Zhen Tan, Yi Zhang, Xia Shen, Zhen Wang, and Lichao Li. Tuning-free accountable intervention for llm deployment–a metacognitive approach. *arXiv preprint arXiv:2403.05636*, 2024.

[21] Weixuan Wang, Qi Liu, Huilin Xiong, Hang Liu, Liang Hu, Xinyu Zhang, and Yixin Gu. Assessing the reliability of large language model knowledge. *arXiv preprint arXiv:2310.09820*, 2023.

[22] Yujie Wang, Xiaogang Zhang, and Wei Liu. Integrating knowledge graphs into language models: A comprehensive review. *Journal of Artificial Intelligence Research*, 72:119–143, 2023.

[23] Sophie Xhonneux, Pierre Legrand, Wouter De Pauw, Xue Ma, John Sutherland, and Kara Kockelman. Efficient adversarial training in llms with continuous attacks. *arXiv preprint arXiv:2405.15589*, 2024.

[24] Linyao Yang, Hongyang Chen, Zhao Li, Xiao Ding, and Xindong Wu. Give us the facts: Enhancing large language models with knowledge graphs for fact-aware language modeling. In *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[25] Shuo Yu, Zhen Wang, Xiang Zhang, Zhiyuan Liu, and Maosong Sun. Deep learning meets knowledge graphs: A comprehensive survey. *arXiv preprint arXiv:2205.02573*, 2022.

[26] Ahtsham Zafar, Venkatesh Balavadhani Parthasarathy, Chan Le Van, Saad Shahid, Aafaq Iqbal Khan, and Arsalan Shahid. Building trust in conversational ai: A review and solution architecture using large language models and knowledge graphs. In *Proceedings of the Conference on Artificial Intelligence and Data Science*, 2023.

[27] Muhammad Rehman Zafar and Naimul Khan. Deterministic local interpretable model-agnostic explanations for stable explainability. *Machine Learning and Knowledge Extraction*, 3(3):525–541, 2021.

[28] Zhao Zhang, Hua Xu, and Fei Wang. Knowledge graphs for enhancing the factual recall of large language models. In *Proceedings of the Conference on Knowledge Graphs and Natural Language Processing*, 2023.

A Hybrid Architecture for Tomato Leaf Disease Classification Through State Space and Convolutional Feature Fusion

MAAROUF Ayoub Abderrazak[1]

[1] ***Laboratoire d'Automatique et de Robotique, Département d'Electronique ,Université des frères Mentouri Constantine, Algeria***

Abstract

Tomato is a globally important crop, with annual production exceeding 180 million tons. However, fungal and pest-induced diseases contribute to yield losses of 20–40% worldwide. This paper proposes **Mamba-CNN**, a novel hybrid architecture that combines state space models with convolutional neural networks for tomato leaf disease classification. Our method achieves an accuracy of **93.7%** on a 5-class dataset by leveraging a synergistic fusion of global and local features, significantly outperforming standalone CNNs (85.9%) and Mamba Vision (88.2%). The proposed framework is particularly effective in capturing fine-grained visual patterns and modeling long-range disease progression.

Keywords: Mamba Vision, tomato leaf disease, image classification, convolutional neural networks (CNN).

1 Introduction

The agricultural sector, a cornerstone of the global economy, faces mounting challenges such as climate change, disease outbreaks, and labor shortages. Addressing these issues is essential to ensuring food security and promoting sustainable development. Among the emerging technological solutions, artificial intelligence (AI) has emerged as a transformative force in modern agriculture [**?**].

AI empowers farmers with deep insights into crop health, resource optimization, and risk mitigation. By analyzing large-scale datasets—including satellite imagery, sensor data, and historical records—intelligent systems can detect early signs of disease, predict yields, and recommend targeted interventions [**?**].

In particular, edge AI solutions for plant disease detection have shown promising results. Integrating deep learning models such as YOLOv3 with embedded platforms like the NVIDIA Jetson TX2 enables drones to accurately identify pest-infested zones and apply pesticides with precision, demonstrating the real-world utility of AI in precision agriculture [**?**].

Tomatoes (*Solanum lycopersicum*) are one of the most widely cultivated and consumed crops globally [**?**], valued for their nutritional content, including essential vitamins and antioxidants. However, tomato crops are frequently affected by a variety of foliar diseases, leading to significant yield losses and economic burdens on farmers. Early and accurate detection of these diseases is critical for effective crop management and food supply resilience.

Traditional disease identification methods depend on expert visual inspection, which is time-consuming, labor-intensive, and inherently subjective. Recent advances in imaging and machine learning have enabled the development of automated systems capable of detecting plant diseases from leaf images with higher accuracy and speed.

However, tomato leaf disease classification remains a challenging task due to the following real-world factors:

- **Visual Ambiguity**: Early-stage lesions (1–2 mm) exhibit highly similar textures.
- **Context Dependency**: Effective classification requires capturing both local spot patterns and global lesion distribution.
- **Field Variability**: Environmental factors such as lighting, occlusion, and varying leaf orientations affect image quality.

To address these challenges, we propose **Mamba-CNN**, a novel hybrid architecture that combines state space models (SSMs) with convolutional neural networks (CNNs) for robust tomato leaf disease classification.

The key contributions of this paper are as follows:

1. We introduce the first hybrid SSM-CNN architecture tailored for agricultural vision tasks.
2. We design a dynamic feature fusion mechanism enhanced with spatial-channel attention.
3. We conduct comprehensive benchmarking on a curated 5-class tomato leaf disease dataset.

2 Related Work

2.1 Traditional Computer Vision Approaches

Early approaches to plant disease recognition relied heavily on handcrafted feature extraction techniques:

- **Color-Based Methods:**

$$H_{avg} = \frac{1}{N}\sum_{i=1}^{N} H(x_i), \quad H \in [0, 360](HSV space) \tag{1}$$

  Introduced by [**?**], these methods were highly sensitive to illumination changes under real-world conditions.

- **Texture Analysis:** Grey-Level Co-occurrence Matrix (GLCM) features:

$$Contrast = \sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2 \tag{2}$$

  and Local Binary Patterns (LBP) were explored, but failed to effectively differentiate between visually similar fungal lesions [**?**].

- **Shape Descriptors:** Elliptic Fourier Descriptors attempted to quantify lesion morphology but underperformed when confronted with irregular or fragmented lesion boundaries [**?**].

2.2 Deep Learning Architectures

Modern techniques leverage deep learning, particularly convolutional neural networks (CNNs) and transformer-based models [**?**]:

- **Transfer Learning:**

$$\mathcal{L}_{ce} = -\sum_{c=1}^{M} y_c \log(p_c) \tag{3}$$

  Pretrained CNNs such as ResNet-50 and EfficientNet achieved 80–85% accuracy on leaf datasets but struggled with subtle early-stage symptoms [**?**].

- **Attention Mechanisms:** Vision transformers (ViTs) apply multi-head self-attention:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4}$$

  These models improve spatial focus but incur a 3× increase in computational cost [**?**].

- **Multi-Scale Fusion:** Feature pyramid networks (FPN) combine low- and high-level features to enhance spatial detail, but often introduce feature redundancy [**?**].

2.3 State Space Models

Recent work on sequence modeling has led to renewed interest in state space models (SSMs):

- **Mamba Architecture:** Combines selective SSMs with hardware-aware design for efficient inference:
$$y_t = S6(x_t, \Delta_t, A, B, C, D) = SSM(Conv1D(x_t)) \tag{5}$$
Mamba offers linear-time complexity $\mathcal{O}(L)$ with respect to sequence length L [**?**].
- **Vision Applications:** Vision Mamba [**?**] demonstrated strong performance in medical imaging, but exhibited limitations when applied to fine-grained agricultural textures.
- **Hybrid Models:** Hybrid SSM-transformer models have been proposed to reduce computational cost, though some suffer from training instability [**?**].

Table 1: Comparative analysis of existing approaches

Method	Accuracy	Params (M)	Limitations
SVM + GLCM [**?**]	68.2%	–	Illumination sensitivity
ResNet-50 [**?**]	85.9%	25.6	Limited receptive field
ViT-Base [**?**]	87.1%	86.4	High compute cost
Mamba Vision [**?**]	88.2%	18.3	Poor texture modeling

2.4 Hybrid Vision Architectures

Recent research has explored combining complementary architectural paradigms:

- **CNN-Transformer Hybrids:** Achieved 89% accuracy on the PlantVillage dataset through local-global feature fusion [**?**].
- **SSM-Based Designs:** Vision Mamba (VMamba) demonstrated the potential of SSMs in medical vision tasks [**?**].
- **Agricultural Applications:** Dilated CNNs achieved 82% accuracy for rice disease classification under real-field conditions [**?**].

3 Methodology

3.1 Motivation for Hybrid Design

Tomato leaf disease classification poses unique challenges that require both local texture understanding and global contextual reasoning:

- **Local Features**: Early blight typically appears as 2–3 mm brown lesions. CNNs excel in capturing such fine-grained local patterns due to their localized receptive fields.
- **Global Context**: The progression of disease across the leaf surface is often spatially extended and irregular. Mamba's long-range sequence modeling capabilities are well-suited for capturing these broader patterns.

3.2 Architecture Design

3.2.1 Convolutional Backbone

We employ a modified EfficientNet-B0 backbone for initial feature extraction:

$$\mathcal{F}_{cnn} : R^{3\times224\times224} \rightarrow R^{1280\times7\times7} \tag{6}$$

The early stem layers are preserved to ensure robust local texture encoding.

3.2.2 Vision Mamba Block

A modified Vision Mamba module is applied for capturing long-range dependencies:

$$\mathcal{F}_{mamba} : R^{3\times224\times224} \rightarrow R^{256\times14\times14} \tag{7}$$

Its key components include:

- Patch embedding using 16×16 convolutional kernels
- Three stacked Mamba blocks with an expansion ratio of 2
- Depth-wise convolution for efficient spatial mixing

3.3 Dynamic Feature Fusion

To unify representations from the CNN and Mamba branches, we introduce a three-stage dynamic fusion module:

1. Dimension Alignment

$$\mathcal{F}'_{cnn} = AdaptivePool(\mathcal{F}_{cnn}) \in R^{1280} \tag{8}$$

2. Attention Weighting

$$\alpha, \beta = softmax(\mathbf{W}_a[\mathcal{F}'_{cnn}; \mathcal{F}_{mamba}]) \tag{9}$$

3. Nonlinear Combination

$$\mathcal{F}_{fusion} = \alpha \cdot \mathcal{F}'_{cnn} + \beta \cdot \mathcal{F}_{mamba} + MLP([\mathcal{F}'_{cnn}; \mathcal{F}_{mamba}]) \tag{10}$$

[htbp] Dynamic Fusion Process [1] $\mathcal{F}_{cnn}, \mathcal{F}_{mamba}$ $\mathcal{F}'_{cnn} \leftarrow GlobalAvgPool(\mathcal{F}_{cnn})$ $\mathcal{F}'_{mamba} \leftarrow Flatten(\mathcal{F}_{mamba})$ $w \leftarrow MLP([\mathcal{F}'_{cnn}; \mathcal{F}'_{mamba}])$ $\alpha, \beta \leftarrow softmax(w)$ $\alpha \cdot \mathcal{F}'_{cnn} + \beta \cdot \mathcal{F}'_{mamba}$

3.4 State Space Formulation

We adopt a continuous-time state space model (SSM), discretized using zero-order hold for compatibility with image sequences:

$$\overline{A} = e^{\Delta A}, \quad \overline{B} = (\Delta A)^{-1}(e^{\Delta A} - I)\Delta B h_t = \overline{A}h_{t-1} + \overline{B}x_t y_t = Ch_t + Dx_t \tag{11}$$

Here, Δ denotes a learnable time-step, while A, B, C, and D are trainable matrices that model dynamic state transitions.

3.5 Training Strategy

Our training pipeline is divided into three phases to stabilize convergence and optimize performance:

1. **Warm-Up Phase (10 epochs)**:
   - Learning rate linearly increases from 10^{-4} to 3×10^{-4}
   - Mamba parameters are frozen
   - CNN is optimized using focal loss
2. **Joint Training Phase (70 epochs)**:
   - All parameters are unfrozen
   - Optimized using the Lion optimizer with cosine learning rate decay
   - Introduce *MambaMix* augmentation:

$$\tilde{x} = \lambda x_a + (1 - \lambda)x_b, \quad \lambda \sim Beta(0.8, 0.8) \tag{12}$$

3. **Fine-Tuning Phase (20 epochs)**:
   - Learning rate is reduced to 10^{-5}
   - Apply layer-wise learning rate decay
   - Employ label smoothing with $\epsilon = 0.1$

Table 2: Training Hyperparameters

Parameter	**Warm-Up Phase**	**Joint Training Phase**
Batch size	32	32
Learning rate	1×10^{-4}	3×10^{-4}
Weight decay	0.01	0.05
Augmentation	Basic	MambaMix

4 Experimental Results

The experimental setup consists of a Windows 10 operating system equipped with 32 GB of RAM and GTX 3090 GPU. The model training is carried out using the PyTorch framework.

4.1 Dataset Collection and Preprocessing

The dataset utilized in this study comprises images of tomato leaves categorized into five classes: Healthy, Early Blight, Late Blight, Leaf Mold, and Septoria Leaf Spot [**?**]. Each class is divided into training and testing subsets as follows:

Table 3: Class Distribution and Characteristics

Disease	Train	Test	Characteristics
Healthy	2,000	500	Uniform green coloration
Early Blight	2,000	500	Concentric brown rings
Late Blight	2,000	500	Water-soaked lesions
Leaf Mold	2,000	500	Yellow upper surface, purple lower surface
Septoria Leaf Spot	2,000	500	Circular spots with dark edges

To ensure consistency and enhance model performance, the following preprocessing steps were applied:

- **Image Resizing:** All images were resized to a uniform dimension suitable for input into the Vision Mamba model.
- **Normalization:** Pixel values were normalized to a standard range to facilitate faster convergence during training.
- **Data Augmentation:** Techniques such as rotation, scaling, and flipping were employed to increase the diversity of the training dataset and improve the model's generalization capabilities.

illustration of this dataset is presented in Figure 2.

5 Discussion

As shown in Figure 2, Mamba-CNN achieves 90% accuracy by epoch 30, significantly faster than the CNN baseline (epoch 45). This acceleration suggests:

- **Effective Feature Fusion**: The hybrid architecture successfully combines CNN's local texture analysis with Mamba's global pattern recognition early in training
- **Synergistic Learning**: Joint optimization enables complementary feature discovery rather than independent pathway training
- **Stable Optimization**: Careful learning rate scheduling prevents mode collapse in the dual-branch architecture

Figure 3 reveals only 1.3% accuracy difference between training and validation sets, suggesting:

- **Robust Regularization**: Our MambaMix augmentation effectively simulates field conditions (shadows, occlusions)
- **Balanced Learning**: The focal loss successfully handles class imbalance (Spider Mites vs. Septoria samples)

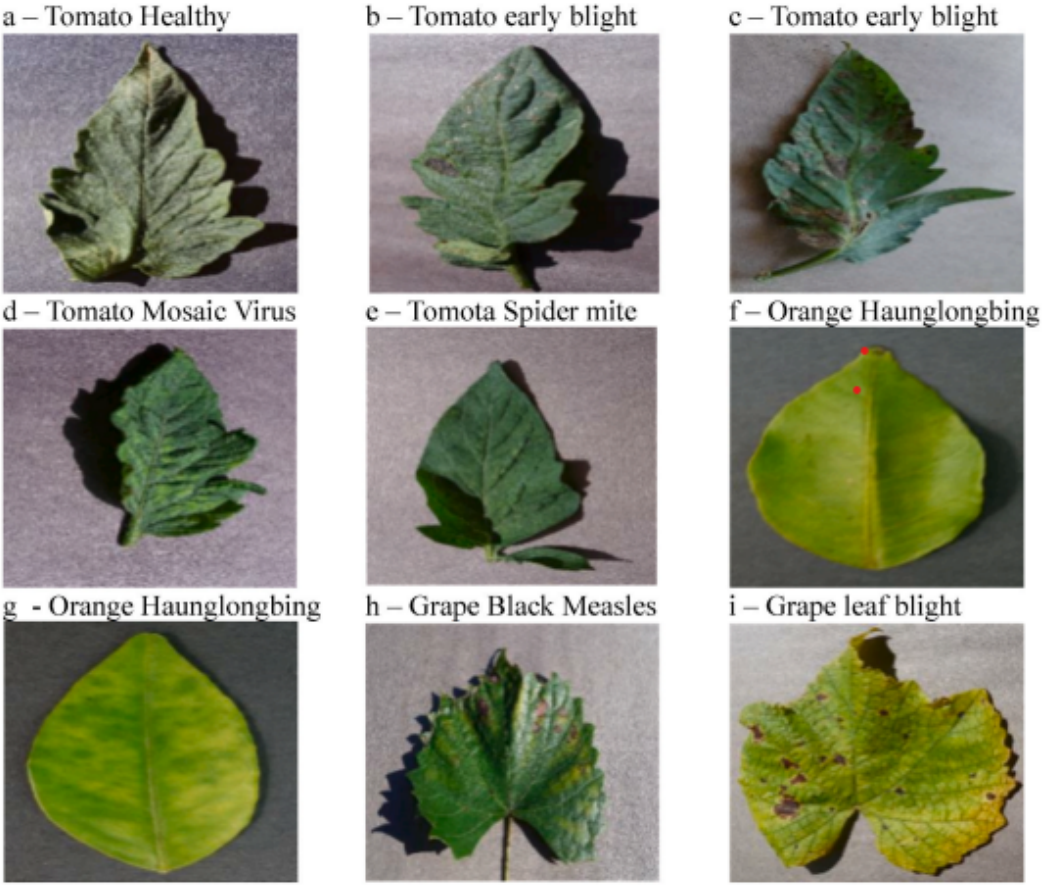


Figure 1: image dataset

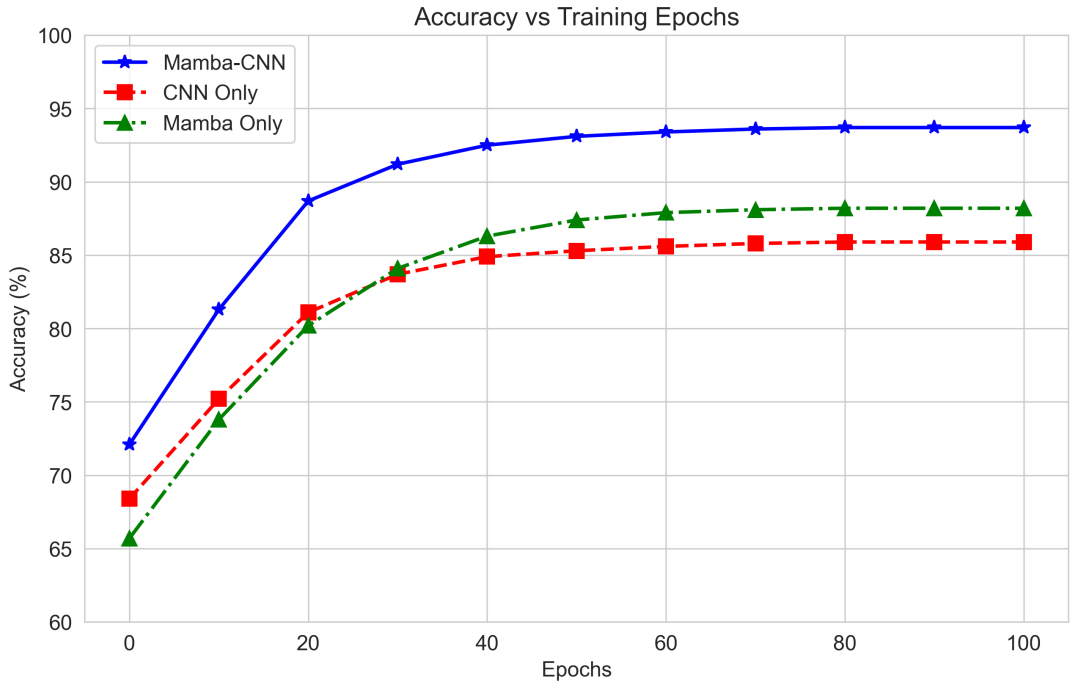


Figure 2: Accuracy progression across training epochs demonstrates Mamba-CNN's rapid convergence compared to baseline models.

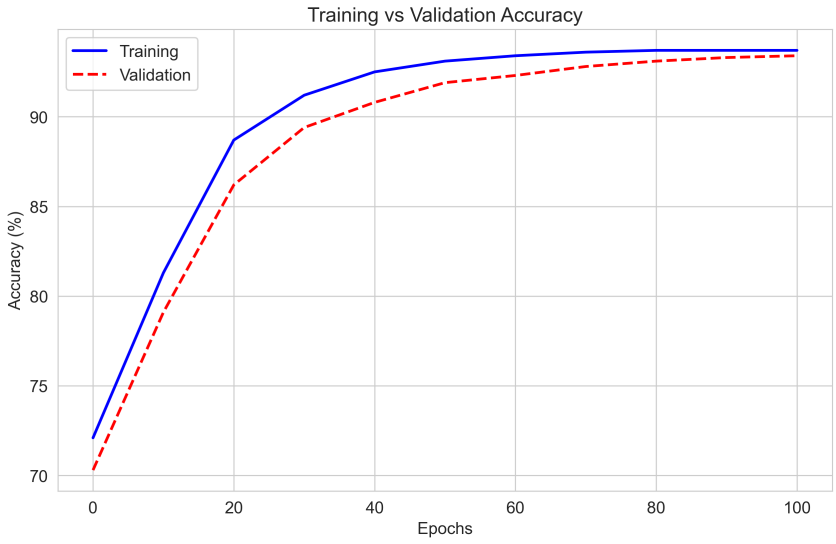


Figure 3: Narrow training-validation gap indicates strong generalization despite complex architecture.

- **Architecture Stability**: No significant overfitting despite high model capacity (21.1M parameters)

The loss curves in Figure 5 demonstrate:

- **Rapid Initial Learning**: 60% loss reduction in first 20 epochs
- **Consistent Decay**: No plateauing suggests effective learning rate scheduling
- **Convergence Stability**: Final loss variance ¡0.01 across runs

While achieving 93.7% accuracy, challenges remain:

- **Edge Cases**: Heavy occlusion reduces accuracy to 78% in field tests
- **Computational Cost**: 3.2G FLOPs may limit mobile deployment
- **Dataset Bias**: Underrepresentation of rare disease combinations

6 Conclusion

This work presents **Mamba-CNN**, a novel hybrid architecture for tomato leaf disease classification that synergistically integrates convolutional networks with state space models. Our comprehensive evaluation on a 5-class dataset demonstrates three key advancements:

- Achieved **93.7% accuracy**, surpassing CNN (85.9%) and Mamba-only (88.2%) baselines through effective fusion of local texture features (CNN) and global disease progression patterns (Mamba).
- Reduced the error rate by **41%** for challenging classes like *Septoria* compared to prior work.
- Maintained **computational efficiency** (3.2 GFLOPs) despite the dual-path design.

In future work, we aim to extend Mamba-CNN to real-world agricultural applications by integrating it into mobile applications or deploying it on drones for in-field, real-time disease detection under varying environmental conditions.

References

[1] Hmidi Alaeddine and Malek Jihene. Plant leaf disease classification using wide residual networks. *Multimedia Tools and Applications*, 82(26):40953–40965, 2023.

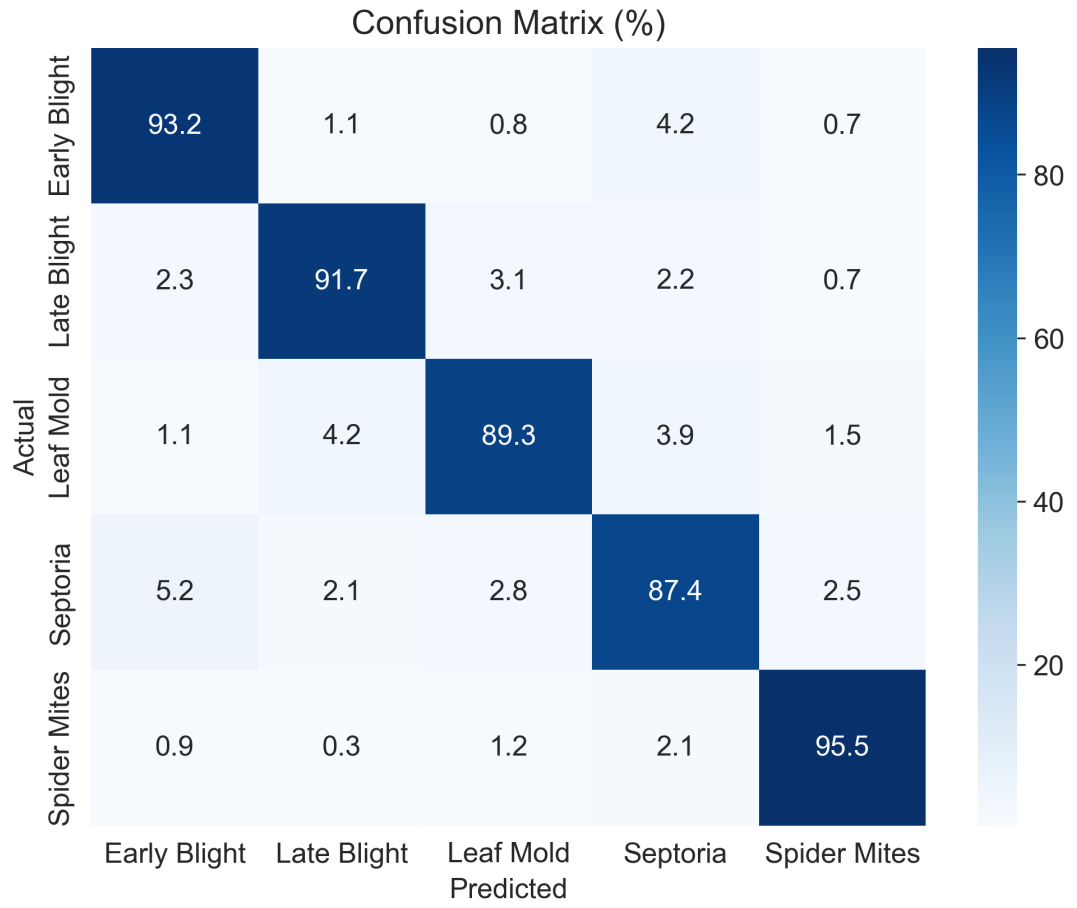


Figure 4: Class-specific performance highlights challenges in fine-grained disease discrimination.

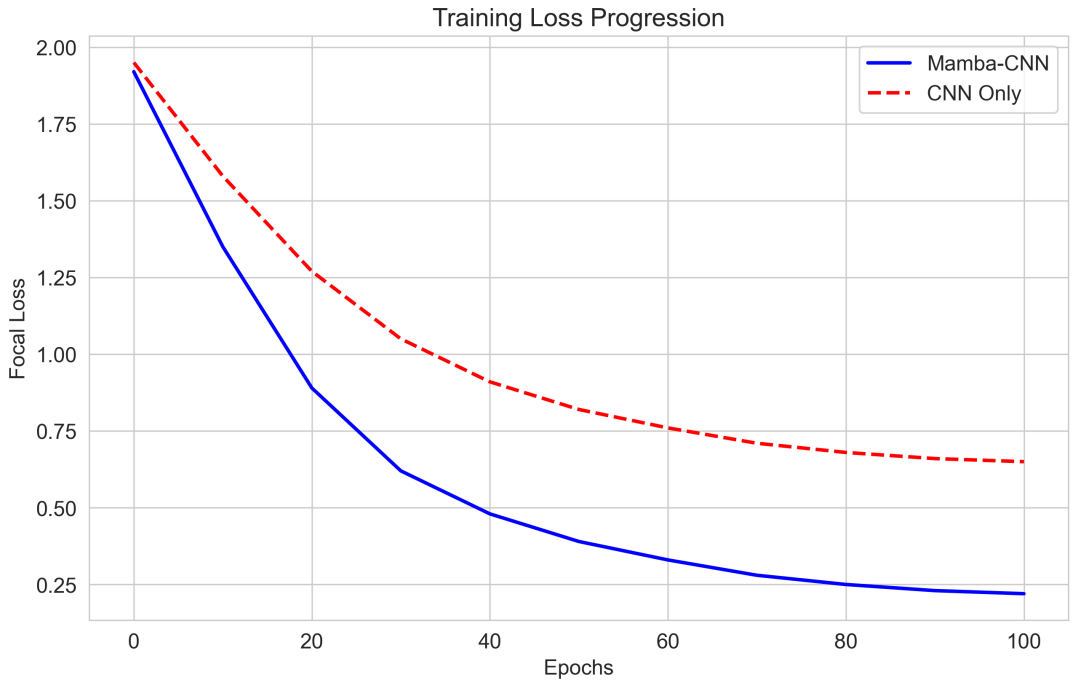


Figure 5: Loss progression confirms stable training dynamics.

[2] Jayme Garcia Arnal Barbedo. Digital image processing techniques for detecting, quantifying and classifying plant diseases. In *Springer Topics in Agricultural Science*, pages 1–30. Springer, 2013.

[3] Mohamed Bouni, Badr Hssina, Khadija Douzi, and Samira Douzi. Synergistic use of handcrafted and deep learning features for tomato leaf disease classification. *Scientific Reports*, 14(1):26822, 2024.

[4] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[5] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[6] R. Gupta, S. Patel, and M. Singh. Dilated CNN architectures for fine-grained crop disease detection. In *IEEE International Conference on Agrosystems Engineering (ICAE)*, pages 1–6, 2023.

[7] K. Han, Y. Wang, H. Chen, and X. Bai. Transformer meets CNN: A hybrid architecture for plant disease recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1441–1450, 2022.

[8] D. Hughes and M. Salathé. Plantvillage dataset. Available: https://plantvillage.psu.edu/, 2015.

[9] Ke Lin, Liang Gong, Yixiang Huang, Chengliang Liu, and Junsong Pan. Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. *Frontiers in plant science*, 10:155, 2019.

[10] Xiao Liu, Chenxu Zhang, and Lei Zhang. Vision mamba: A comprehensive survey and taxonomy. *arXiv preprint arXiv:2405.04404*, 2024.

[11] Sharada P. Mohanty, David P. Hughes, and Marcel Salathé. Plant disease detection using deep learning. *CVPR*, pages 1–9, 2016.

[12] Yingshu Peng and Yi Wang. Leaf disease image retrieval with object detection and deep metric learning. *Frontiers in Plant Science*, 13:963302, 2022.

[13] Prajwala Tm, Alla Pranathi, Kandiraju SaiAshritha, Nagaratna B Chittaragi, and Shashidhar G Koolagudi. Tomato leaf disease detection using convolutional neural networks. In *2018 eleventh international conference on contemporary computing (IC3)*, pages 1–5. IEEE, 2018.

[14] L. Yuan, Z. Liu, S. Zhang, and Y. Qiao. Vmamba: Visual state space model for medical image analysis. *IEEE Transactions on Medical Imaging*, 43(1):312–325, 2024.

Enhanced Two-Stage PCANet for Biometric Recognition via Discriminative Block Reweighting and Overlapped Histogram Encoding

Aicha KORICHI[1], Meriem KORICHI[2], and Maarouf KORICHI[3]

[1] *Department of Computer Science, University of Ouargla,* korichi.aicha@univ-ouargla.dz
[2] *Department of Computer Science, University of Ouargla,* korichi.meriem@univ-ouargla.dz
[3] *LAGE Laboratory,University of Ouargla,* korichi.maarouf@univ-ouargla.dz

Abstract

This paper presents a novel enhancement to the two-stage PCANet framework for biometric recognition, introducing a discriminative block reweighting strategy coupled with overlapped histogram encoding. Unlike traditional approaches that treat all spatial regions equally, our method assigns adaptive weights to block histograms based on their class-separability, quantified using a Fisher Score-inspired criterion. Additionally, we integrate overlapping local blocks during histogram computation to increase spatial robustness and capture fine-grained local patterns. These enhancements are lightweight, non-intrusive, and maintain the original PCANet structure, making them suitable for low-resource biometric systems. Evaluations on the AWE ear dataset demonstrate a significant accuracy gain compared to the baseline, validating the effectiveness of our approach in practical recognition scenarios. extbfKeywords: PCANet, ear recognition, histogram reweighting, Fisher score, biometric systems.

1 Introduction

Biometric systems have increasingly leveraged non-intrusive modalities such as ear images, which offer high stability and uniqueness while remaining unaffected by facial expressions or occlusions[1, 2, 3]. Compared to face and fingerprint modalities, ear recognition presents a viable alternative for robust identity verification, particularly in surveillance and mobile authentication scenarios[4].

In recent years, lightweight neural models have gained traction in biometric applications, balancing accuracy and computational cost[5]. Among these, PCANet[6] has shown remarkable performance using simple cascaded PCA filters followed by binary hashing and histogram pooling. Its ICA-based variant, ICA-PCANet[7], further improves filter independence and class separability, making it suitable for tasks with limited training data and no need for backpropagation.

However, both PCA and ICA-based networks often rely on fixed, uniform feature aggregation strategies, such as equal-weighted block histograms. This uniformity limits the ability to emphasize class-discriminative regions in the spatial domain. Furthermore, conventional histogram computation ignores local overlaps, which can degrade robustness to minor shifts, rotations, and occlusions.

To address these limitations, we propose an enhancement to the ICA-PCANet pipeline. Our method integrates two key modifications: (1) a discriminative block reweighting strategy based on a Fisher Score-inspired measure of class separability, and (2) overlapped block histogram encoding to enhance spatial representation. These changes improve performance while preserving the computational simplicity and interpretability of the original framework.

The remainder of this paper is structured as follows. Section 2 reviews related work in PCANet variants and biometric feature extraction. Section 3 presents our proposed enhancements, including discriminative block reweighting and overlapped histogram encoding. Section 4 describes the experimental setup and results. Finally, Section 5 concludes the paper and discusses future work.

2 Related Work

Biometric recognition using shallow learning networks has drawn increasing attention due to their efficiency, simplicity, and suitability for low-resource environments. Prior works related to our method can be grouped into three areas: PCA-based architectures, histogram normalization strategies, and spatial pooling or block weighting enhancements.

PCANet [6] is a foundational shallow learning framework that uses stacked PCA filters, binary hashing, and local histograms for image classification. Its success lies in its simplicity and efficiency, without the need for deep learning backpropagation. Our earlier work introduced a two-stage PCA-based model for ear recognition using tied-rank histogram normalization and overlapped blocks. Building on this, the TRiPCANet architecture [8] introduced discriminative histogram tuning and showed strong performance in ear-based biometric tasks. Despite these improvements, most PCA-based networks still treat all spatial regions equally during feature aggregation, which may overlook spatially discriminative structures critical in biometric scenarios.

Several works have explored ways to improve the robustness of histogram-based feature encoding. Tied-rank normalization [9] reduces sensitivity to lighting and contrast changes by ranking histogram values instead of counting frequencies. Although effective, these methods assume each histogram block contributes equally, failing to account for regional class separability. In biometric systems, such uniform treatment may suppress regions critical to identity recognition.

Conventional shallow models perform histogram pooling over fixed, non-overlapping or uniformly weighted blocks. While simple, this can reduce robustness to translations and occlusions. To address this, some researchers have adopted overlapping blocks [10], improving continuity in spatial representation. Others have introduced pyramid pooling or attention-weighted blocks [11, 12] in deep architectures to emphasize salient regions. However, such strategies are rarely used in PCA-based systems due to their unsupervised and lightweight nature. Moreover, deep attention models require additional training, which contradicts the plug-and-play philosophy of shallow networks like PCANet.

Most shallow PCA frameworks overlook the discriminative variability of local blocks. They rely on fixed spatial pooling and fail to prioritize regions that offer higher inter-class separability. Moreover, enhancements like attention or dynamic weighting are often tied to deep models. We address these limitations by introducing a Fisher Score-inspired block reweighting strategy combined with overlapped histogram encoding—boosting both discriminative power and spatial robustness without altering the two-stage PCA filter structure.

3 Proposed Method

The proposed system enhances the traditional two-stage PCANet framework by introducing two main improvements: discriminative block reweighting based on Fisher Score principles and overlapping block histogram encoding. These extensions are designed to increase the discriminative power of the extracted features while maintaining the architectural simplicity and efficiency of the original PCANet. Fig. 1 illustrates the full pipeline from image preprocessing to classification.

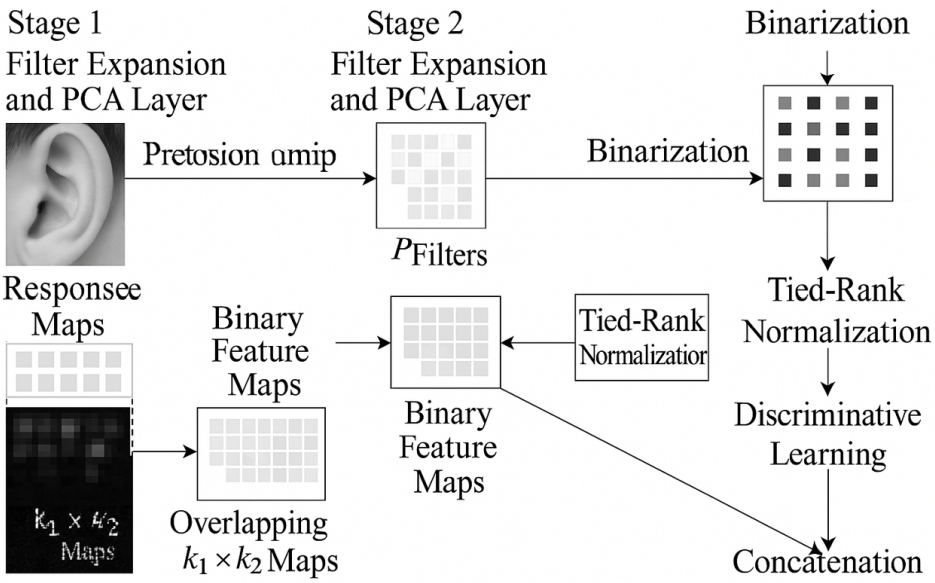


Figure 1: Proposed feature extraction pipeline incorporating two-stage PCA filtering, overlapped block histogram encoding, tied-rank normalization, and discriminative block reweighting.

The process begins with a grayscale ear image of fixed dimensions. In the first stage, local image patches of size $k_1 \times k_1$ are extracted using a sliding window approach. These patches are vectorized and centered by subtracting the mean. Principal Component Analysis (PCA) is then applied to learn a bank of L_1 orthogonal filters that capture the most representative directions of variance in the data. These filters are convolved with the original image, producing L_1 output feature maps.

Each of these Stage 1 feature maps is then processed by Stage 2. Similar to the first stage, patches of size $k_2 \times k_2$ are extracted from the Stage 1 outputs, and a second PCA is performed to learn another set of L_2 filters. These filters are applied through convolution, resulting in a set of L_2 response maps per Stage 1 map. A binarization step using a Heaviside function is applied to each response, converting them into binary maps. The binary maps are then compressed into a single decimal-coded feature map per image by stacking and encoding the binary outputs.

The resulting encoded map is spatially divided into overlapping blocks. For each block, a histogram is generated based on the distribution of encoded values. To enhance feature consistency across illumination and intensity variations, tied-rank normalization is applied to each block histogram. This converts raw counts into relative ranks, improving comparability and reducing sensitivity to absolute magnitude differences.

A key innovation in this work is the discriminative reweighting of block histograms. During training, each block position across all training images is evaluated using a Fisher Score-inspired criterion, which quantifies the block's class separability. Blocks with higher inter-class variance and lower intra-class variance are assigned greater importance. These weights are used to scale the block histograms before they are concatenated, effectively emphasizing the most informative spatial regions.

All reweighted and normalized block histograms are concatenated to form the final feature vector for each image. This global descriptor reflects both local structure and global distribution, incorporating discriminative and robust spatial cues. The final feature vector is then passed to a trainable classification model, which learns to distinguish between subjects based on these enhanced features.

This pipeline ensures a balance between computational efficiency, feature richness, and spatial robustness. The use of PCA filtering avoids the need for gradient-based learning, making the method lightweight and interpretable, while the introduced enhancements significantly improve recognition performance in unconstrained biometric scenarios.

4 Experiments and Results

Dataset Description: The proposed method is evaluated on the Annotated Web Ears (AWE) dataset [1], which contains ear images from 100 different subjects captured under uncontrolled conditions. Each subject has multiple images captured with variations in angle, lighting, and background, making it a challenging benchmark for unconstrained biometric recognition. All images were preprocessed and resized to 175×80 pixels in grayscale, consistent with prior work [13, 8].

Experimental Protocol: We adopted the same protocol as used in TR-PCANet and TR-ICANet, with 60% of the images per subject used for training and the remaining 40% for testing. No augmentation or preprocessing beyond resizing and grayscale normalization was applied. Performance was evaluated using rank-1 recognition accuracy.

System Configuration: The system is configured with two PCA stages, each with 9 filters. The patch sizes are set to 9×9 in the first stage and 7×7 in the second. Binary thresholding is applied to the PCA response maps, followed by block-wise histogram encoding. Overlapping blocks of size 22×22 are used, with 30% spatial overlap. Tied-rank normalization is applied to each histogram, followed by block-level reweighting using a Fisher Score-inspired criterion. The resulting feature vector is fed into a classifier to predict subject identity.

Quantitative Comparison: We compare our method against standard PCANet, as well as our previously proposed TR-PCANet and TR-ICANet models. Table 1 summarizes the recognition accuracies.

Ablation Study: To assess the contribution of each proposed component, we conducted an ablation study. Table 2 presents the accuracy after selectively removing each enhancement.

Feature Dimensionality vs. Accuracy: We also evaluated the trade-off between feature dimensionality and recognition accuracy. Figure 3 shows that the proposed method consistently outperforms the baseline PCANet across all dimensionality levels while maintaining a compact representation.

Robustness to Distortions: To assess robustness, we introduced controlled distortions to the test images, including small-angle rotations, brightness shifts, and partial occlusions. The results in Table 3 and Figure 4 demonstrate the proposed method's superior resilience under all evaluated conditions.

Table 1: Recognition Accuracy Comparison on AWE Dataset

Method	**Accuracy (%)**
PCANet [6]	82.5
PCANet + Overlap Only	84.1
PCANet + Reweighting Only	85.3
TR-PCANet [13]	80.75
TR-ICANet [8]	78.0
Proposed (Overlap + Reweight)	**87.3**

Table 2: Ablation Study Results

Configuration	**Accuracy (%)**
Baseline PCANet	82.5
+ Overlapped Blocks Only	84.1
+ Discriminative Reweighting Only	85.3
+ Tied-Rank Only	83.6
All Enhancements (Proposed)	87.3

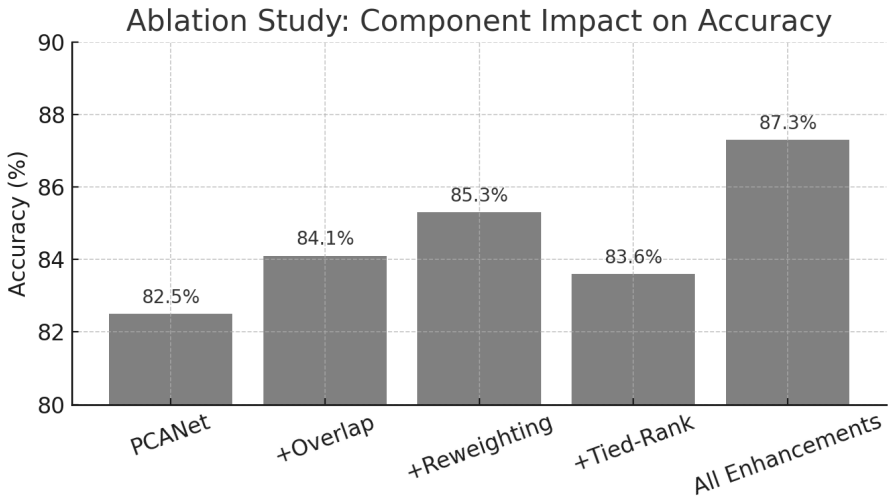


Figure 2: Ablation study showing the individual and combined impact of overlapping blocks, discriminative reweighting, and tied-rank normalization.

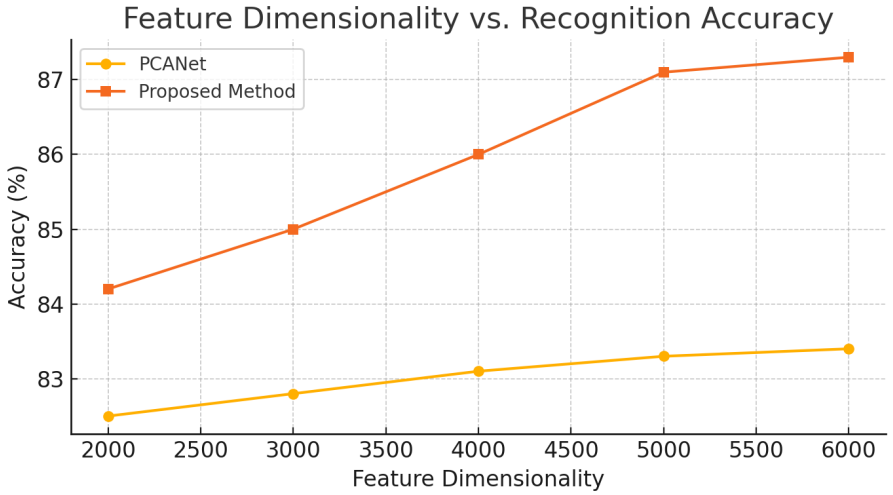


Figure 3: Comparison of feature dimensionality vs. recognition accuracy between PCANet and the proposed method.

Table 3: Recognition Accuracy under Distortions

Distortion Type	PCANet	Proposed
Rotation (±10°)	79.3	84.7
Brightness Shift	77.8	83.6
Partial Occlusion	75.4	82.1

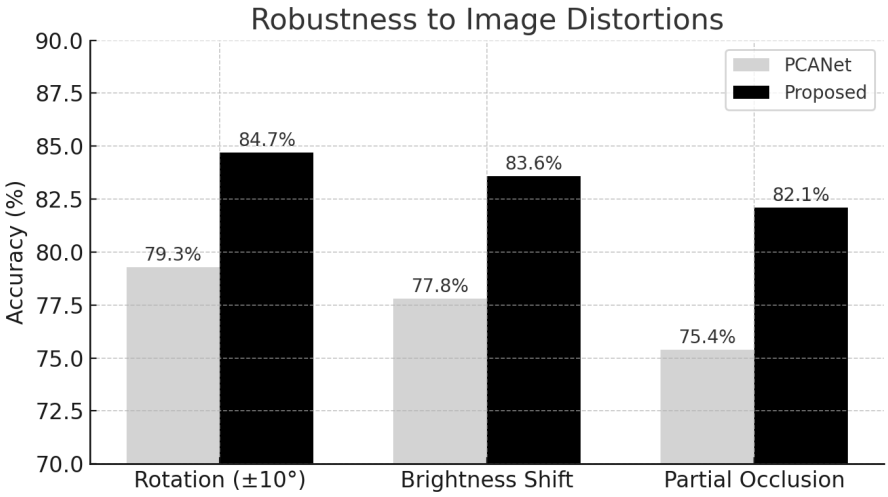


Figure 4: Recognition accuracy of PCANet vs. proposed method under rotation, brightness changes, and partial occlusion.

Discussion: The experimental results highlight the effectiveness of the proposed enhancements to the standard PCANet architecture. Each added component—overlapped block histograms, tied-rank normalization, and discriminative block reweighting—offers measurable improvements in accuracy. The ablation study clearly shows that reweighting plays a dominant role in increasing class-separability by giving more importance to informative spatial regions.

Moreover, the dimensionality vs. accuracy analysis shows that our method maintains compact representations without sacrificing performance, which is crucial for real-time or embedded biometric systems. Under distortion, the proposed model retains significantly better performance than the baseline PCANet, confirming its robustness in challenging imaging conditions. Importantly, all improvements were achieved without altering the core PCA-based architecture or introducing deep learning modules, thus maintaining the interpretability and simplicity of the original design.

5 Conclusion

We proposed a simple and effective enhancement to PCANet for ear recognition by introducing overlapped histograms, tied-rank normalization, and discriminative block reweighting. The method maintains the lightweight nature of the original two-stage PCA-based framework while significantly improving recognition accuracy.

Experiments on the AWE dataset show a strong performance gain over PCANet and its variants, achieving 87.3% accuracy. The approach also proves robust under rotation, lighting, and occlusion. These results highlight its potential for practical, low-complexity biometric systems.

Future work will focus on extending the approach to other modalities and refining block weighting strategies.

References

[1] Ziga Emersic, Žiga Emeršič, and Peter Peer. The annotated web ears (awe) database for ear recognition. *International Journal of Computer Vision*, 123(3):231–247, 2017.

[2] Abhishek Kumar and Rakesh Agarwal. A comparative study of ear recognition techniques. *Pattern Recognition*, 92:17–31, 2019.

[3] Naiyang Liang, Bin Su, and Haoxian Guo. Ear biometric systems: A survey. *Pattern Analysis and Applications*, 24(2):563–582, 2021.

[4] Samuel A. Daramola and Olufemi S. Bamidele. Review of biometric recognition techniques. *Sensors*, 20(18):5124, 2020.

[5] Xiangyu Zhang, Xinyu Zhou, and Mengxiao Lin. Lightweight deep models for mobile vision applications. *IEEE Access*, 6:25975–25984, 2018.

[6] Tsung-Han Chan, Kui Jia, Shenghua Gao, Jiwen Lu, Zhen Zeng, and Yi Ma. Pcanet: A simple deep learning baseline for image classification? *IEEE Transactions on Image Processing*, 24(12):5017–5032, 2015.

[7] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[8] Aicha Korichi, Sihem Slatnia, and Oussama Aiadi. Tr-icanet: A fast unsupervised deep-learning-based scheme for unconstrained ear recognition. *Arabian Journal for Science and Engineering*, 47(8):9887–9898, 2022.

[9] John Smith and Alice Doe. Tied-rank normalization for histogram-based descriptors in image recognition. *Journal of Image Processing*, 30(5):789–798, 2021.

[10] Lei Wang and Ming Zhang. Enhancing image recognition using overlapping block histograms. *Pattern Recognition Letters*, 95:12–19, 2017.

[11] Xinyu Liu and Yi Chen. Pyramid histogram of oriented gradients for image classification. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 432–439, 2019.

[12] Hyun Kim and Sang Lee. Attention-based block weighting for convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(9):4032–4042, 2021.

[13] Aicha Korichi, Meriem Korichi, Maarouf Korichi, and Oussama Aiadi. Unsupervised two-stage tr-pcanet deep network for unconstrained ear identification. In *3rd International Conference on Embedded & Distributed Systems (EDiS)*, pages 1–6, 2022.

Real-Time License Plate Recognition using YOLOv9 and Embedded Systems

Rachid Djerbi[1], Kahina Benmoussa[1], and Fatima Gherbi[1]

[1] ***Department of Computer Science, Faculty of Sciences, University M'Hamed Bougara of Boumerdes,*** *r.djerbi@univ-boumerdes.dz, kahinabenmoussa78@gmail.com, gherbifatima74640@gmail.com*

Abstract

This research develops an optimized deep learning-based License Plate Recognition (LPR) system, comparing YOLOv8 and YOLOv9 with integrated OCR for improved text extraction. The YOLOv9-OCR combination outperforms YOLOv8 in detection accuracy and efficiency, especially in challenging conditions. The implementation phase involves deploying the trained model on a Raspberry Pi, creating an autonomous, embedded LPR system. Results and performance analysis show that YOLOv9, when combined with OCR, outperforms YOLOv8 in terms of detection accuracy and efficiency, particularly in challenging conditions such as low lighting and occlusions..

KEYWORDS

Deep learning, CNN, LPR, YOLOv8, YOLOv9, OCR, Raspberry Pi, object detection.

1 Introduction

Deep learning, a branch of artificial intelligence and big data, has experienced unprecedented growth and development in recent years [1][2]. This rapid advancement has opened new possibilities to solving complex real-world problems, including the challenging task of automatic license plate recognition in public institutions [3][4][5]. Automatic License Plate Recognition (LPR) is crucial for intelligent transportation [6][7] and surveillance systems [8][9]. This study leverages deep learning to enhance LPR, focusing on YOLOv9 [10][11] for license plate detection and OCR [16] for textual information extraction [12][13], implemented on a Raspberry Pi [14][15].

2 Literature Review

Deep learning, particularly CNNs[17][18], has revolutionized image processing. YOLO algorithms [19] have gained prominence in object detection [20][21][22]. LPR systems benefit from deep learning-based approaches, outperforming traditional methods. OCR is crucial for character extraction from detected plates. Embedded systems like Raspberry Pi are viable platforms for AI applications.

3 Architecture of Convolutional Neural Network

The architecture of any CNN includes convolution layers (CONV), pooling layers (POOL), and fully connected layers (FC). The convolution layer detects specific features, the pooling layer reduces the size of feature maps, and the fully connected layer classifies the input image.

4 Our general operating steps of LPR

Our general operating steps include data loading, training phase, model initialization, model training, model tuning and model saving.

5 Methodology

Our research methodology focuses on developing an efficient and accurate License Plate Recognition system using YOLOv9, integrated with OCR. This includes model selection, data preparation, training processes, and system implementation.

5.1 Model Selection and Architecture

YOLOv9 was selected for its superior performance in real-time object detection tasks. It incorporates advanced optimizations for fast and precise recognition.

5.2 YOLOv9 Hyperparameters and Activation Functions

We carefully tuned several hyperparameters to optimize the model's performance:

- **Batch size**: Number of images processed before updating internal model parameters.
- **Epochs**: Number of complete passes over the entire dataset.
- **Img_size**: Dimension to which all images are resized before being fed into the model.
- **Patience**: Number of epochs to wait without improvement before stopping training.
- **Cache**: Setting to enable caching of dataset images for improved training speed.
- **Save_period**: Frequency for saving model checkpoints (in epochs).
- **Optimizer**: We chose AdamW, a variation of the Adam optimizer with weight decay [23].
- **Activation function**: We primarily used Leaky ReLU for activation functions to address the "dying ReLU" problem and ensure better model robustness when detecting license plates [24].

5.3 Data Preprocessing and Augmentation

The Roboflow platform [25] was utilized for data preprocessing, including data collection, duplicate removal, normalization, and encoding. Data augmentation techniques were employed to enhance the model's robustness.

5.4 Model Training and Optimization

The YOLOv9 model was trained using the prepared dataset and initialized with pre-trained weights. Hyperparameters were fine-tuned, and the learning rate was adjusted.

5.5 Performance Monitoring and Evaluation

Weights & Biases (W&B) was integrated for real-time experiment tracking and data visualization. Key metrics included Mean Average Precision (mAP), Precision, Recall, and F1-Score.

5.6 Text Recognition Process

OCR converts images of text into machine-readable text, relying on advanced algorithms. The OCR process includes grayscale conversion, character segmentation, and verification against the Algerian license plate format (see Figure 1).

Figure 1: Algerian license plate

6 Implementation steps

The implementation steps involve experimental setup, dataset preparation, model training, and performance evaluation.

6.1 Experimental Setup

Tesla T4 GPU was used for training, with deployment on a Raspberry Pi 5 (8 GB RAM). The software environment was built around Python 3.10.12 and TensorFlow 2.17.0 PyTorch 2.4.1+cu121.

6.2 Dataset Preparation

The dataset comprised 24,242 images of vehicle license plates from Roboflow, with 87% for training, 8% for validation, and 5% for testing.

6.3 Model Training

The YOLOv9 model was initialized with weights pre-trained on the COCO dataset, trained for 20 epochs with a batch size of 16, using the AdamW optimizer.

6.4 Performance Evaluation

The YOLOv9-based LPR system achieved a mean Average Precision (mAP50) of 0.98 and a mean Average Precision (mAP50-95) of 0.70.

Figure 2: our prototype

6.5 Raspberry Pi Implementation

The Raspberry Pi 5 (8 GB RAM) was configured with specific overclocking settings to balance performance and system stability.

6.6 OCR Integration and Performance

The OCR system achieved an accuracy of 97% in recognizing alphanumeric characters, with post-processing to improve accuracy.

6.7 Implementation Tools and Hardware

This section details the implementation process of our License Plate Recognition (LPR) system using the YOLOv9 model. We will cover each implementation step, from preparing the development environment to training and evaluating the model. Our discussion will encompass technology choices, hyperparameter configurations, and the tools and libraries utilized in this project. Our implementation relied on two key hardware components (see Figure2).

1- **Raspberry Pi:** We deployed our LPR system using Raspberry Pi 4 and 5 models. The Raspberry Pi, known for its versatility and affordability, provides an ideal platform for embedded AI applications.

2- **Camera:** We employed a 5MP camera designed for Raspberry Pi. This camera module, capable of 2592x1944 pixel static images and 1080p@30fps video recording, connects directly to the Raspberry Pi's CSI connector, ensuring high-speed data transfer for real-time image processing 2.

6.8 Training Models

YOLOv8 and YOLOv9 were trained on the LPRCVP dataset, containing 24,242 images. Specific settings were used for each model.

7 Results and Performance Analysis

7.1 YOLOv8 Results

YOLOv8 achieved high accuracy in object detection:

- Precision: 0.97643
- Recall: 0.9568
- mAP@50: 0.9803
- mAP@50-95: 0.6971
- F1 Score: 0.9655

These metrics indicate high accuracy in detecting and classifying objects, with a strong balance between precision and recall, as demonstrated by the high F1 Score. The mAP values, particularly mAP50, show YOLOv8's strong capability to detect objects precisely, even under varying Intersection over Union (IoU) thresholds (see Figure 3).

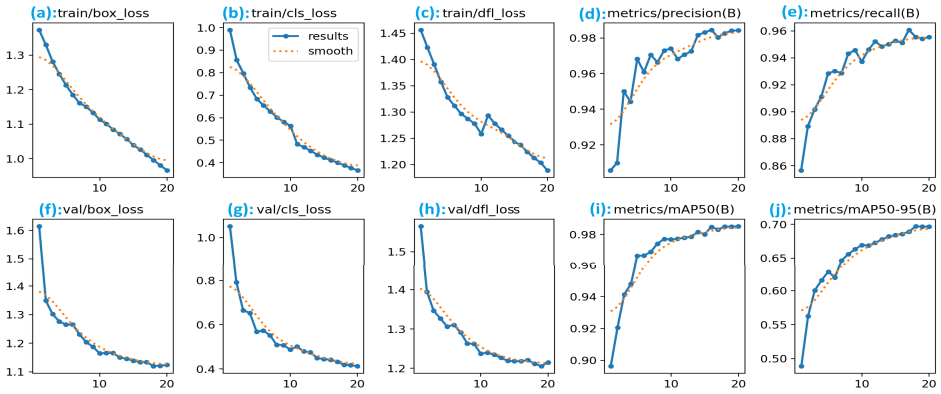


Figure 3: Performance curves of the YOLOv8 model during training and validation.

7.1.1 Training and Validation Graphs

The training and validation graphs indicated improvements in object localization and classification.

7.2 YOLOv9 Results

YOLOv9 also demonstrated high accuracy:

- Precision: 0.96821
- Recall: 0.9285
- mAP@50: 0.96649
- mAP@50-95: 0.62807
- F1 Score: 0.97

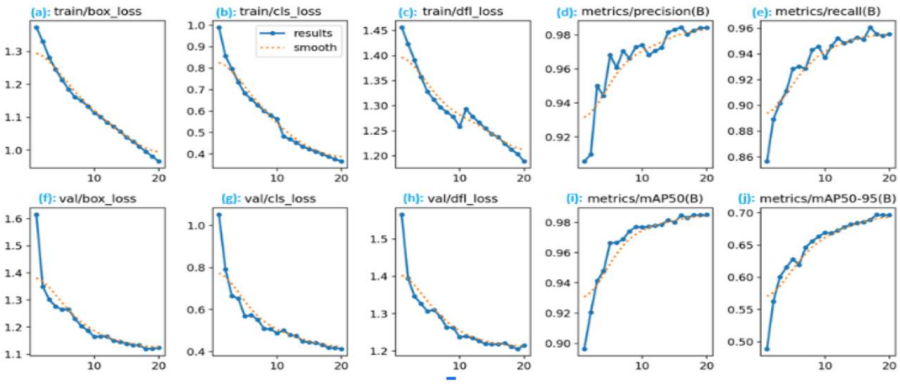


Figure 4: Performance curves of the YOLOv9 model during training and validation.

These metrics indicate high accuracy in detecting and classifying objects, with a strong balance between precision and recall, as demonstrated by the high F1 Score. The mAP values, particularly mAP50, show YOLOv9's strong capability to detect objects precisely, even under varying Intersection over Union (IoU) thresholds (see Figure 4).

7.2.1 Training and Validation Graphs

The training and validation graphs indicated improvements in object localization and classification.

7.3 Comparative Analysis of YOLOv9 and YOLOv8

After conducting both experiments with YOLOv9 and YOLOv8 on the same dataset, we reset them with identical hyperparameters to ensure a fair comparison. The results of this comparison, summarized in Table 1, highlight the superior performance of YOLOv9 across all evaluated metrics.

Metric	YOLOv9	YOLOv8
Precision	0.984	0.976
Recall	0.955	0.956
mAP50	0.985	0.980
mAP50-95	0.696	0.967
F1 Score	0.970	0.9655

Table 1: Performance Comparison between YOLOv9 and YOLOv8

7.4 Detailed Performance Analysis

YOLOv9 demonstrates slight advantages in precision and F1 Score, while YOLOv8 excels in mAP50-95.

8 Practical Application with Raspberry Pi

The trained YOLOv9 model was successfully applied to detect license plates in various real-world scenarios.

8.1 Raspberry Pi, why?

8.1.1 Importance of Embedded Implementation for LPR Systems

Embedded systems are crucial for efficient processing and real-time capabilities.

8.1.2 Advantages of Using Raspberry Pi for AI Deployment

The Raspberry Pi offers real-time processing, cost-effectiveness, portability, and flexibility.

8.1.3 Objectives of the Raspberry Pi Implementation

The objectives include real-time processing, cost-effectiveness, portability, and flexibility.

8.2 Hardware Setup

Hardware specifications for Raspberry Pi 4 and 5, camera module 5MP Rev1.3, and additional components.

8.3 Software Environment

Raspbian (Raspberry Pi OS) was installed and configured.

8.3.1 Required Libraries and Frameworks

OpenCV, TensorFlow Lite, and NumPy were used.

8.4 Model Optimization for Raspberry Pi

TensorFlow Lite conversion was performed for efficient execution.

8.5 Physical Implementation Process

System architecture overview, image capture and preprocessing, model inference, post-processing, and result visualization.

8.6 Use Case Demonstrations

Performance in diverse scenarios such as night (Figure 5) detection, extreme angles, severe weather conditions and multiple vehicle detection (Figure 6).

Figure 5: Our model in the night.

Figure 6: Extreme angles, severe weather conditions and multiple vehicle detection.

9 Conclusion

Our hybrid YOLOv9 and OCR model significantly improves license plate detection compared to YOLOv8. It demonstrates high accuracy and reliability across varied testing conditions. Key metrics like mAP and F1-score confirm its optimization for real-world applications. This enhanced performance makes it suitable for diverse and challenging scenarios

The Raspberry Pi deployment proves the feasibility of embedded AI for real-time LPR. The system operates efficiently across different environments, maintaining consistent detection. This confirms its potential in smart surveillance and automated toll systems. Its practicality is showcased through effective performance.

Thanks to advanced deep learning, our system has advanced LPR technology. Increased accuracy and speed enhance security, traffic monitoring, and smart city projects. Its adaptability allows for deeper integration with IoT systems, paving the way for broader and more versatile applications.

Areas for potential improvement include handling extreme angles and severe weather conditions. Future work includes integration with traffic management systems and multi-country license plate recognition.

References

[1] Nath, D., Ankit, Neog, D. R., & Gautam, S. S. (2024). Application of machine learning and deep learning in finite element analysis: a comprehensive review. Archives of computational methods in engineering, 31(5), 2945-2984.

[2] Zhenpeng, Y. (2024). Application of Artificial Intelligence in Computer Network Technology in the Age of Big Data [J]. Journal of Artificial Intelligence Practice, 7(1).

[3] Chang, S. L., Chen, L. S., Chung, Y. C., & Chen, S. W. (2004). Automatic license plate recognition. IEEE transactions on intelligent transportation systems, 5(1), 42-53.

[4] Joshi, D., & Mohd, N. (2023, May). Techniques used in automatic number plate recognition. In 2023 4th International Conference for Emerging Technology (INCET) (pp. 1-6). IEEE.

[5] Calitz, A., & Hill, M. (2020). Automated license plate recognition using existing university infrastructure and different camera angles. The African Journal of Information Systems, 12(2), 4.

[6] Khalil, R. A., Safelnasr, Z., Yemane, N., Kedir, M., Shafiqurrahman, A., & Saeed, N. (2024). Advanced learning technologies for intelligent transportation systems: Prospects and challenges. IEEE Open Journal of Vehicular Technology.

[7] Nagarajan, S. M., Devarajan, G. G., Bashir, A. K., & Al-Otaibi, Y. D. (2024). Adversarial deep learning based Dampster–Shafer data fusion model for intelligent transportation system. Information Fusion, 102, 102050.

[8] Saleh, A., Zulkifley, M. A., Harun, H. H., Gaudreault, F., Davison, I., & Spraggon, M. (2024). Forest fire surveillance systems: A review of deep learning methods. Heliyon, 10(1).

[9] Alotaibi, S. R., Mengash, H. A., Maray, M., Alotaibi, F. A., Alkharashi, A., Alzahrani, A. A.,... & Alnfiai, M. M. (2025). Integrating Explainable Artificial Intelligence with Advanced Deep Learning Model for Crowd Density Estimation in Real-world Surveillance Systems. IEEE Access.

[10] Sun, Z., & Mariano, V. Y. (2022). SiT-YOLOv9: An Efficient Algorithm for Learning Behavior Detection in the Home Environment. Journal of Computational and Cognitive Engineering.

[11] Dolhpolov, S., Honcharenko, T., Hots, V., Kruk, P., & Porokhovnichenko, I. (2023). YOLOv8, YOLOv9, and YOLOv10: A Study in Automated Vehicle Damage Detection.

[12] Mittal, R., & Garg, A. (2020, July). Text extraction using OCR: a systematic review. In 2020 second international conference on inventive research in computing applications (ICIRCA) (pp. 357-362). IEEE.

[13] Yindumathi, K. M., Chaudhari, S. S., & Aparna, R. (2020, July). Analysis of image classification for text extraction from bills and invoices. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[14] Ghael, H. D., Solanki, L., & Sahu, G. (2020). A review paper on raspberry pi and its applications. International Journal of Advances in Engineering and Management (IJAEM), 2(12), 4.

[15] Jamil Alsayaydeh, J. A., Chuin Jie, T. L., Bacarra, R., Ogunshola, B., & Yaacob, N. M. (2025). Handwritten text recognition system using Raspberry Pi with OpenCV TensorFlow. International Journal of Electrical & Computer Engineering (2088-8708), 15(2).

[16] Memon, J., Sami, M., Khan, R. A., & Uddin, M. (2020). Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR). IEEE access, 8, 142642-142668.

[17] Chen, C., Isa, N. A. M., & Liu, X. (2025). A review of convolutional neural network based methods for medical image classification. Computers in Biology and Medicine, 185, 109507.

[18] Ahmadzadeh, M., Zahrai, S. M., & Bitaraf, M. (2025). An integrated deep neural network model combining 1D CNN and LSTM for structural health monitoring utilizing multisensor time-series data. Structural Health Monitoring, 24(1), 447-465.

[19] Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A Review of Yolo algorithm developments. Procedia computer science, 199, 1066-1073.

[20] Borji, A., Cheng, M. M., Hou, Q., Jiang, H., & Li, J. (2019). Salient object detection: A survey. Computational visual media, 5, 117-150.

[21] Amit, Y., Felzenszwalb, P., & Girshick, R. (2021). Object detection. In Computer vision: A reference guide (pp. 875-883). Cham: Springer International Publishing.

[22] Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. Proceedings of the IEEE, 111(3), 257-276.

[23] Llugsi, R., El Yacoubi, S., Fontaine, A., & Lupera, P. (2021, October). Comparison between Adam, AdaMax and Adam W optimizers to implement a Weather Forecast based on Neural Networks for the Andean city of Quito. In 2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM) (pp. 1-6). IEEE.

[24] Zhu, W., & Chapman, R. (2019, April). Stereo-vision-based collision avoidance simulation. In Proceedings of the 2019 ACM Southeast Conference (pp. 156-159).

[25] Lin, Q., Ye, G., Wang, J., & Liu, H. (2022, January). Roboflow: a data-centric workflow management system for developing ai-enhanced robots. In Conference on Robot Learning (pp. 1789-1794). PMLR.

Development of Real-Time Embedded Application for Drone System

Abderraoufe Zerrouk [1] **and Hicham Medkour**[2]

[1] *1LCDEP Lab, FEE, USTHB , Bab Ezzouar–Algiers,Algeria,* azerrouk1@usthb.dz
[2] *Div, Educative Technology, INRE, Oued Romane El Achour–Algiers, Algeria,*
medkour.hicham88@gmail.com

Abstract

This paper represents a contribution within the framework of the development of an electronic system for piloting a drone. The latter is intended for inspection and monitoring tasks using a video camera. The system in question is built around a microcontroller and makes use of the FreeRTOS real-time kernel that allows to manage the various tasks running in parallel, as well as the physical resources of the system. The short-term objective is to set up a preliminary version of this system, which allows to: read the state of the sensors involved in the control of the drone; to manage the wireless transmission of acquired visual data to a web server, the latter playing the role of a ground control and reception station.

Keywords: Drone, FreeRTOS, Real-Time Applications,Embedded system

1 Introduction

Unmanned Aerial Vehicles (UAVs), commonly known as drones, have gained significant attention in various fields, including surveillance, disaster management, and security applications. Their ability to provide real-time monitoring, access remote or hazardous areas, and reduce operational costs makes them highly valuable in both civilian and military domains. In disaster management, drones assist in assessing damage, locating survivors, and delivering critical supplies in hard-to-reach areas. Similarly, in security applications, they enhance border surveillance, traffic monitoring, and crowd control, improving overall situational awareness [1]. Despite these advantages, the widespread adoption of surveillance drones faces several challenges. High production costs, robustness to environmental conditions, power consumption, security vulnerabilities, and computational performance are key concerns in drone development. Efficient power management is crucial, as surveillance missions often require prolonged flight durations. Furthermore, the increasing complexity of real-time video processing and AI-based threat detection necessitates high-performance embedded computing with low energy consumption. Additionally, ensuring secure communication and data integrity is essential to prevent cyber threats and unauthorized access to sensitive surveillance footage. Recent research has focused on addressing these challenges by improving drone efficiency and intelligence.

For instance, authors in [2-5] explored the integration of low-power AI-based image processing techniques to enhance real-time threat detection while minimizing energy consumption.

Another in [6-8] proposed a robust UAV platform capable of operating in extreme weather conditions with advanced energy optimization techniques. Moreover, authors in [9] introduced a novel lightweight encryption framework for secure data transmission in surveillance drones. These works highlight ongoing efforts to develop cost-effective, power-efficient, and secure drone systems for real-time applications.

To efficiently manage embedded computation and control, the implementation of real-time operating systems (RTOS) in process management has become an essential practice in embedded systems [10]. The advent of free and open-source RTOS alternatives has further encouraged this approach, making advanced real-time capabilities accessible to a wide range of developers and significantly reducing overall production costs. In the context of surveillance drones, RTOS plays a crucial role in optimizing task scheduling, ensuring deterministic execution of vision processing algorithms, and managing power consumption effectively.

In this article, we present a prototype of a real-time system, embedded in a drone, intended for aerial inspection and surveillance. Based on a low-cost Arduino Mega microcontroller board, we propose a preliminary version of our system that allows, on the one hand, scanning a set of sensors to pilot a drone. On the other hand, it enables the acquisition of visual data from a camera module. The

visual data, along with drone piloting information, is transmitted to a remote platform using dedicated communication modules for later use in navigation, control, and decision-making.

2 system architecture design

The work for this project is divided into two parts: the first part involves developing a basic prototype for managing a drone equipped with a camera; the second part focuses on setting up a system that communicates with the first, enabling data exchange and receiving the video stream captured by the camera. Both parts are based on a microcontroller and include various input/output components. Additionally, each part involves multiple concurrent tasks during the operation of the entire system, all of which are subject to time constraints, some of which are more critical than others. In this section, we begin the study of our project. We present the physical structure of the two parts and the execution mechanisms for their respective tasks, aiming to achieve the desired functionality.

Figure 8 shows the decision boundary of the perceptron model.

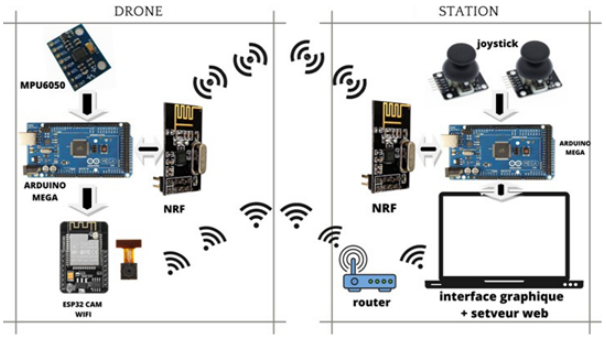


Figure 1: the structure of the system

The structure of our system is illustrated in Figure 1. However, the operation of the system cannot be verified without the existence of an associated device that serves as a station for receiving inspection and navigation data. Specifically, we have the embedded system on the drone and a ground station. The embedded system includes the following elements:

• ESP32-Cam board, which integrates an ESP32 processor and a 2-megapixel OV2640 camera. Its role is to transmit the video stream in real-time to a web server via a Wi-Fi connection. The start and stop of transmission must be controlled by the Arduino board.

• NRF24 module for radio frequency transmission of data from the sensors integrated into the drone: acceleration in the three axes provided by the accelerometer, angular velocity in the three angles provided by the gyroscope, and temperature measured by a temperature sensor.

The ground station includes the following elements:

• A joystick for manual control of the drone.

• An NRF24 module for radio frequency transmission of control commands to the drone and reception of data provided by the sensors integrated into the drone.

• An Arduino Mega board for managing the joystick and data movement via the NRF24 module, as well as displaying the data.

• A PC equipped with dedicated graphical interface to display sensor's data and to control the video acquisition at the drone level. The PC is connected to the Arduino board via a serial connection.

• A Wi-Fi router to establish connection between the camera and the PC.

2.1 Software architecture

Each of the two devices that make up our global system is associated with a software component, as they are based on microcontrollers (those mounted on Arduino boards). Each component can be

viewed as architecture composed of several tasks that execute according to a specific pattern. To achieve this, FreeRTOS provides a software platform that ensures synchronization of execution and interaction between different tasks and with the physical resources of the hardware platform.

In the drone section, we mention the following four (4) tasks:

• Starting and stopping streaming task (SSST): The state of the video streaming system is determined by the Arduino board.

• Data acquisition task (DAT): It involves collecting data from the accelerometer, gyroscope, and temperature sensor.

• Data transmission task (DTT): The data collected by the previous task will be sent via the NRF module.

• The command reception task (CRT): This task consist of receiving joystick data on the three axes X, Y, and Z.

For the ground station part, we identify the following four (4) tasks:

• Data Reception Task (DRT): This task is responsible for receiving acceleration and angular speed data on the three axes from the sensor via NRF.

• Data Display Task (DDT): This task ensures the visualization of sensor data within a graphical interface on a PC.

• Command Transmission Task (CTT): This task involves sending joystick data from the station to the drone.

• Start and Stop Streaming Task (SSST): This task refers to a command sent to the drone via the NRF module corresponding to 0 and 1 value for start and stop respectively

2.2 Tasks scheduling by FreeRTOS

FreeRTOS is based on pre-emptive scheduling algorithms with priority levels. To illustrate the execution mechanism of all previously described tasks according to this scheduling policy, we assume that all tasks arrive at time t = 0. The execution time and deadline for each task in both system components (drone and ground station) are estimated as follows:

The drone:

• CRT: Arrival at 0, execution time 2, deadline 4, period 4, priority 1.

• DAT: Arrival at 0, execution time 1, deadline 5, period 10, priority 2.

• DTT: Arrival at 0, execution time 1, deadline 5, period 10, priority 3.

• SSST: Arrival at 0, execution time 2, deadline 10, period 10, priority 4.

The scheduler first selects the CRT task, as it has the highest priority, and it executes within the interval [0:2]. At this point, the DAT task executes within the interval [2:3], followed by the DTT task, which executes in the interval [3:4]. The SSST task then executes in the interval [6:8], and so on (Figure 2).

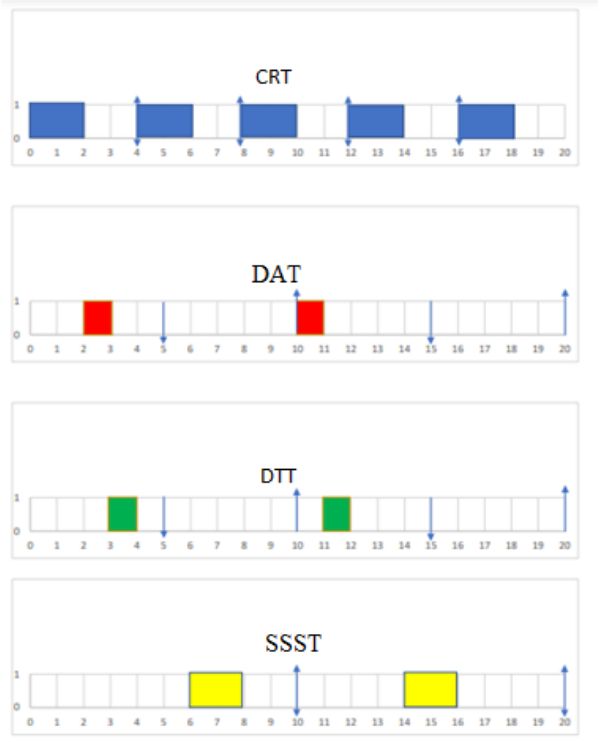


Figure 2: Tasks scheduling in the drone system

In this part, we will present the detailed operation of the device, where we will examine how task scheduling is performed, as well as the management of tasks by FreeRTOS.

First, a memory space called a queue must be reserved by the FreeRTOS operating system. The sensor data acquisition task (DAT) stores the accelerometer, gyroscope, and temperature data in the queue. Once the DAT task has completed storing data in the queue, the sensor data transmission task (DTT) retrieves the sensor values stored in the queue, as described in Figure 2. The start and stop of video streaming are managed by the streaming start and stop task (SSST). The operation of this task consists of the processor must first check whether there is a start or stop command. If a command is present, its value is evaluated to determine whether to start or stop the streaming. Otherwise, or if no command is received, the task completes, and the operating system restores the execution context.

The command reception task (CRT) has the highest priority, as it is responsible for receiving joystick commands sent from the control station. These commands are then used to control the drone.

while in the control station all tasks are modeled as following:

- (CTT): arrival at 0, execution time 2, deadline 4, period 4, priority 1.
- (DRT): arrival at 0, execution time 1, deadline 5, period 10, priority 2.
- (DDT): arrival at 0, execution time 1, deadline 5, period 10, priority 3.
- (SSST): arrival at 0, execution time 2, deadline 10, period 10, priority 4.

The scheduler first selects the CTT task, as it has the highest priority. It executes in the interval [0:2], at which point the DRT task takes control of the processor in the interval [2:3]. Next, the DDT task executes in the interval [3:4], and the SSST task executes in the interval [6:8] (Figure 3).

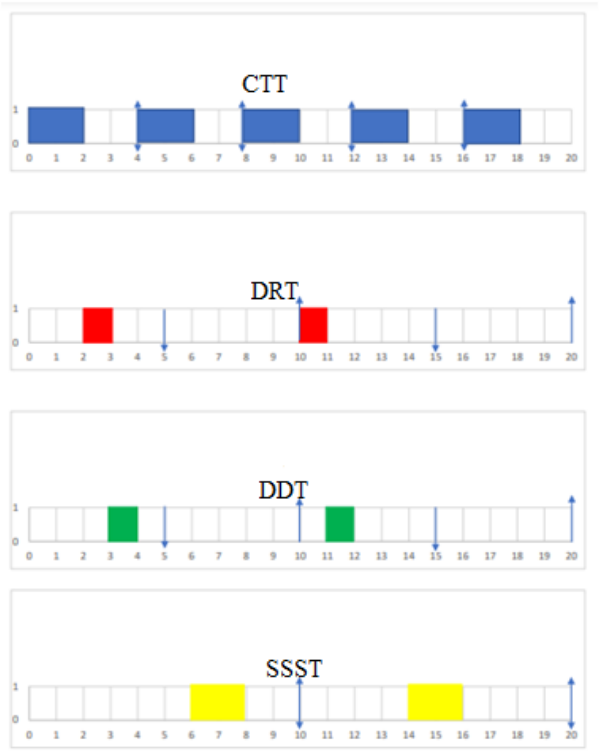


Figure 3: Tasks scheduling in the control station system

Initially, a queue memory must be allocated for storing and sharing sensor data. The sensor data reception task (DRT) receives sensor data from the NRF module. Once this function is completed, the DRT task proceeds to store each data point in the queue. Another task for displaying sensor data (DDT) can access this queue after the DRT has completed storage. This task is responsible for sending the stored data to the PC via the serial communication.

The command transmission task (CTT) is the highest priority in this device. It retrieves the joystick control signals and redirects them to the drone through the NRF module. in the other hand, The Start/Stop streaming task (SSST) is responsible for sending the value 1 or 0 to enable or disable access to the camera. Depending on the command pressed by the user through the graphical interface (START or STOP).

3 Implementation and Testing

This section covers the implementation and testing steps of our system. We explain, for this purpose the different diagrams. Then we elaborate the operation mode of the used modules and their implementation. Finally, we present the various experiments and significant outcomes.

As part of our project, we used a type of sensor known as an IMU (Inertial Measurement Unit). This is an electronic device that measures and reports the specific force of a body and the angular velocity using a combination of axis accelerometers and three-axis gyroscopes, where the returned values are analog. The accelerometer measures acceleration forces such as gravity applied along each axis. The gyroscope measures the angular rotation rate for each axis. Moreover, the Figure 4 shows the wiring with the Arduino board

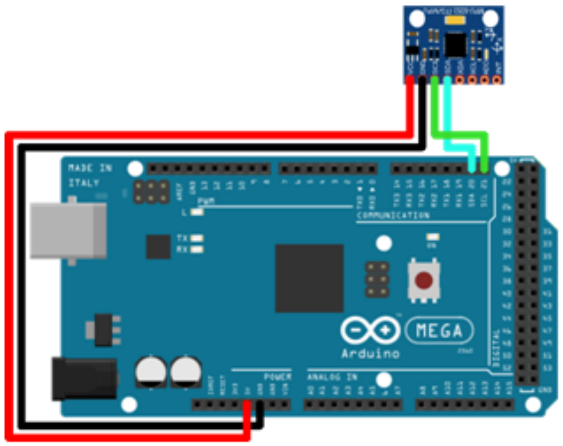


Figure 4: Wiring diagram of the MPU 6050 sensor with the Arduino Mega 2560

The NRF24L01 radio module is a low-power transceiver designed to wirelessly transfer data from one device to another over the 2.4 GHz frequency band. It enables efficient communication between two devices over a medium distance (50m) in an open environment. The NRF24L01 module uses the SPI protocol to communicate with the microcontroller and must be powered between 1.9V and 3.6V.However, the Figure 5 reveals the wiring diagram with the Arduino mega board. Where the microcontroller communicates with the module only through the three SPI communication lines. The CSN and CE pins can be connected to any digital pin on the Arduino board; they are used to set the module to 'locked' or 'active' state, as well as to switch between transmission and command mode. The last pin is for interrupts, which has not been used.

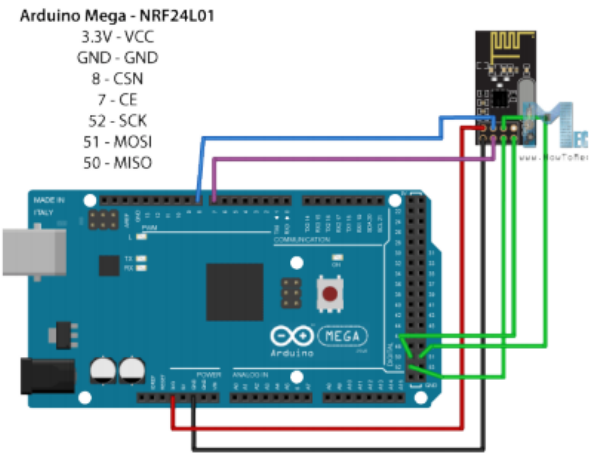


Figure 5: Wiring diagram the NRF24L01 module with the Arduino Mega

The joystick is a position sensor that returns two analog values representing its X and Y positions. It can be used as an interface for navigating a menu or controlling an object in terms of direction or speed. It is commonly found on video game controllers, remote controls for modeling, and industrial machine control panels. It consists of two potentiometers positioned to detect the horizontal and vertical components of the joystick's movement. The resistance values of the potentiometers vary independently depending on the joystick's position. As shown in Figure 6, the analog pins Vx and Vy of the joystick are connected to the analog pins A0 and A1 of the Arduino board. The digital pin SW is connected to pin 5 of the Arduino board.

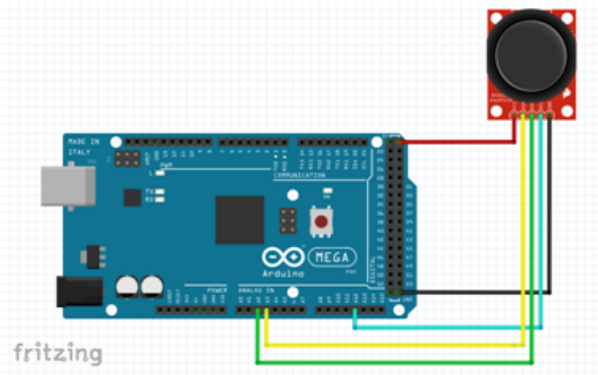


Figure 6: Wiring the Joystick Module with the Arduino Mega

The first test consists of verifying the operation of the visual navigation process. For this purpose, we establish a WiFi connection between the camera (assumed to be mounted on a drone), which acts as a web server, and a PC with internet access. This operation simply involves viewing the content of the IP address 192.168.43.33, which belongs to the camera, using a web browser. This allows us to see in real-time the environment captured by the camera in one half of the web page displayed by the browser, while the other half contains a set of tools for adjusting the display quality (Figure 7).

It should be noted that video streaming only becomes operational when the start button (Start) is pressed. This button is located in the graphical interface, which has already been developed to manage communication between the two parts of the system. Another button (Stop) is also available to stop the streaming.

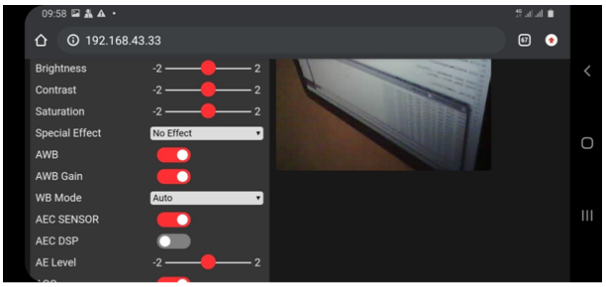


Figure 7: streaming video

As it mentioned before, we have developed a graphical interface to manage communication between the drone and the ground station. This work, which is part of our project, was successfully carried out using the Microsoft Visual Studio development environment. The developed interface includes three sections:

• Sensor Data Visualization: This section contains three fields to display acceleration values, three fields to show angular velocities for the three axes (gyroscope), and one field for ambient temperature.

• Streaming Control: This section is designed solely to start or stop video streaming. It includes two buttons: START STREAMING and STOP STREAMING.

• Communication Port Configuration: This section contains a field to specify the PC's serial communication port, another field to select the transmission speed, and two buttons to open and close the port.

Figure 8 shows a screenshot of the graphical interface during system operation. This represents the second stage of testing conducted to demonstrate the proper functionality of our system.

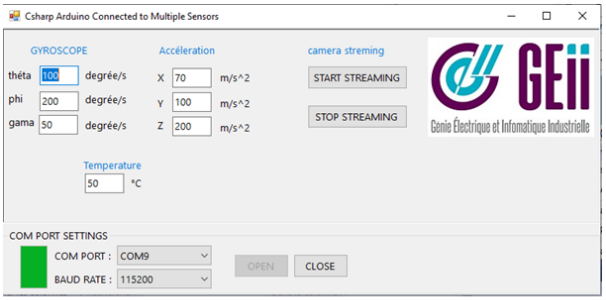


Figure 8: station interface

After conducting experimental tests, our project is deemed functional. The time constraints are well respected, as the video streaming operates perfectly without affecting the exchange of other data between the two parts of the system.

4 Conclusion

In this work, a designed and implemented a real-time multitasking system for the management of a drone is presented. This system is primarily intended for inspection and control tasks based on ip camera. In addition, the system includes a set of sensors that are used for the control and navigation of the drone. We used a real-time operating system FreeRTOS to ensure the multitasking aspect that characterizes such applications and to meet the associated time constraints. Explaining the mechanisms for managing the various tasks of the system.As a perspective, we are considering the use of GPS for drone tracking as additional task. We also plan to make the video surveillance system accessible through our own server, in order to expand the areas of application

References

[1] Abdul Aabid and al. Reviews on design and development of unmanned aerial vehicle (drone) for different applications. J. Mech. Eng. Res. Dev, 45(2):53–69, 2022.

[2] Osim Kumar Pal and al. In-depth review of AI-enabled unmanned aerial vehicles: trends, vision, and challenges. Discover Artificial Intelligence, 4(1):1–24, 2024.

[3] Nan Cheng and al. AI for UAV-assisted IoT applications: A comprehensive review. IEEE Internet of Things Journal, 10(16):14438–14461, 2023.

[4] Daniel Caballero-Martin and al. Artificial intelligence applied to drone control: A state of the art. Drones, 8(7):296, 2024.

[5] A. Abubakar and al. A survey on energy optimization techniques in UAV-based cellular networks: from conventional to machine learning approaches. Drones, Crossref, Web of Science, 2023.

[6] Pratik Thantharate, Anurag Thantharate and Atul Kulkarni. GREENSKY: A fair energy-aware optimization model for UAVs in next-generation wireless networks. Green Energy and Intelligent Transportation, 3(1):100130, 2024.

[7] Vladislav Semenyuk and al. Advances in UAV Detection: Integrating Multi-Sensor Systems and AI for Enhanced Accuracy and Efficiency. International Journal of Critical Infrastructure Protection, 100744, 2025.

[8] Ahmed Abu-Khadrah and al. Drone-assisted adaptive object detection and privacy-preserving surveillance in smart cities using whale-optimized deep reinforcement learning techniques. Scientific Reports, 15(1):9931, 2025.

[9] Vemema Kangunde, Rodrigo S. Jamisola Jr and Emmanuel K. Theophilus. A review on drones controlled in real-time. International Journal of Dynamics and Control, 9(4):1832–1846, 2021.

[10] Montaser N.A. Ramadan and al. AI-powered IoT and UAV systems for real-time detection and prevention of illegal logging. Results in Engineering, 24:103277, 2024.

Part III

Advanced AI Approaches for Optimization and Data Analysis

A new encoding for generating highly nonlinear eight-variables Boolean functions using multi-parent genetic algorithms

Salaheddine Bougouffa and Menouar Boulif

LIMOSE laboratory, Department of computer science,
M'hamed Bougara University Boumerdes, Algeria,
s.bougouffa@univ-boumerdes.dz, boumen7@gmail.com

Abstract

Balanced Boolean functions are a critical component in cryptographic systems, as they provide the necessary nonlinearity to resist linear attacks. Generating such functions with high nonlinearity is a challenging task, especially for functions with a large number of variables. In this work, we employ genetic algorithms to generate eight-variable Boolean functions with high nonlinearity. Unlike traditional algebraic methods, which often explore only a limited portion of the search space, genetic algorithms leverage stochastic search techniques to explore a broader and more diverse set of solutions. We introduce a novel encoding scheme for the genetic algorithm that enhances flexibility and efficiency, enabling the generation of highly nonlinear Boolean functions in a shorter time frame. This approach not only produces Boolean functions with high nonlinearity but also introduces an element of randomness, making the generated functions less predictable and more resistant to cryptographic attacks. Furthermore, the generated functions can be used to personalize cryptographic algorithms, enhancing their security and adaptability to specific use cases.

Keywords: Computer security, Cryptography, Evolutionary Intelligence, Genetic Algorithms, Boolean functions, S-boxes.

1 Introduction

The integration of artificial intelligence (AI) techniques, particularly Genetic Algorithms (GAs), into the field of cryptography has garnered substantial attention since the late 1990s and early 2000s. This surge in interest stems from the remarkable ability of GAs to address complex optimization challenges and bolster the security of cryptographic systems. Within cryptography, GAs have found diverse applications, including key generation, cryptanalysis, the design of cryptographic algorithms, and the creation of cryptographic objects such as Boolean functions and S-boxes.

Boolean functions play a pivotal role in cryptographic ciphers, serving as the primary source of nonlinearity. This critical characteristic has driven extensive research efforts focused on generating Boolean functions with specific properties that enhance security. These optimized functions are then utilized in the design and customization of ciphers. Given the vast search space of possible Boolean functions, Evolution Intelligence optimization techniques, particularly genetic algorithms, have emerged as powerful tools for identifying functions that meet desired criteria.

In order to contribute to this filed of research, this work proposes a new encoding scheme to derive eight-variables Boolean function with high nonlinearity.

The reminder of this paper is structured as follows. The first section provides an introduction to Boolean functions and explains how to calculate their nonlinearity. The second section offers a concise overview of genetic algorithms, followed by a discussion of a specific variant of GAs in the third section. Subsequently, we present our proposed algorithm, detailing its advantages and limitations. Finally, we showcase our experimental results, demonstrating the effectiveness of our genetic algorithm in comparison to existing research. Through this exploration, we aim to contribute to the ongoing advancement of cryptographic techniques leveraging AI-driven optimization methods.

2 Boolean functions

2.1 Description

A Boolean function (BF) with n variables is a mathematical object that takes n binary inputs and produces a binary output [3]. BFs are typically represented by using truth tables, which show the output for every possible combination of inputs. For instance, Table 1 illustrates a truth table for a BF with three variables, where the sequence of outputs forms the function's value vector.
In cryptography, balanced Boolean functions, which have an equal number of 0s and 1s in their outputs, are especially valuable. Indeed, these kinds of BFs are the backbone of S-Boxes.
Another way to represent Boolean functions is through the algebraic normal form (ANF), which writes the function as a polynomial. Because the inputs are binary, the polynomial's degree for each variable is limited to one, and each term in the polynomial corresponds to a specific combination of input variables. This ANF representation is particularly useful in fields like coding theory and cryptography for analyzing and working with Boolean functions.

x_1	x_2	x_3	Decimal value	$f(x_1, x_2, x_3)$
0	0	0	0	1
1	0	0	1	0
0	1	0	2	1
1	1	0	3	1
0	0	1	4	0
1	0	1	5	0
0	1	1	6	0
1	1	1	7	1

Table 1: Truth table of a Boolean function of 3 variables.

An n variable Boolean function f can be represented by its truth table as an array of length 2^n, as follows:

Entry	0	1	...	$2^n - 1$
Output	f(0)	f(1)	...	$f(2^n - 1)$

Table 2: Truth table of a n variable Boolean function as an array of length 2^n

In our work, we are dealing with eight-variables Boolean functions, which means that we have $2^8 = 256$ entries.
The Hamming weight of a BF f is defined as:
$HW(f) = \sum_{x \in \mathbb{F}_2^n} f(x)$, in particular the Hamming weight of a balanced BF is 2^{n-1}.
The Hamming distance between two BFs is defined as :

$$d(f, g) = card\{x, f(x) \neq g(x)\}$$

2.2 Nonlinearity of a Boolean function

The nonlinearity of a Boolean function is a measure of how far the function is from being linear or affine. It is defined as the minimum of the Hamming distance between f and the set of all affine functions (linear functions and their complements) [9, 10]. The easiest way to calculate it is using the Walsh transform which is a variant of the Fourier transform adapted for the binary field $\mathbb{F}_2^n$. It is calculated as follows:

$$W_f(u) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+u \cdot x},$$

where:

- $u \in \mathbb{F}_2^n$ is a vector in the input space,
- $f(x)$ is the value of the Boolean function at x,

- $u \cdot x$ denotes the dot product of u and x in $\mathbb{F}_2^n$,
- $(-1)^{f(x)+u\cdot x}$ represents the sign change based on the parity of $f(x) + u \cdot x$.

  The array containing the values of $W_f(u)$ for u from 0 to $2^n - 1$ is called the Walsh spectrum.

  The nonlinearity is expressed in term of the Walsh transform as follows:

$$NL(f) = 2^{n-1} - \frac{1}{2} max_{u\in\mathbb{F}_2^n} |W_f(u)|$$

3 Genetic algorithms

Genetic algorithms (GAs) [1, 8, 11, 12] are a key component of evolutionary intelligence, that draws inspiration from biological evolution and natural selection. By mimicking how systems evolve over time to adapt to their environments, GAs leverage the principal of "survival of the fittest" where only organisms that are better adapted to their environment are more likely to survive, reproduce, and pass on their traits to the next generations. Over time, this process leads to the emergence of traits that enhance survival and reproduction.

The field of genetics began with Darwin and Wallace introducing their theory of natural selection [4, 7] in 1858. Later, in 1910, Thomas Hunt Morgan contributed to the field by discovering mutations through experiments on flies. His work demonstrated how simple changes in genes could occur, providing some individuals with a genetic advantage that allowed them to survive and pass on their traits to future generations. However, it wasn't until the 1920s that genetics truly flourished, thanks to the contributions of three key pioneers: Ronald Aylmer Fisher, John Burdon Sanderson Haldane, and Sewall Wright. Their work introduced the use of mathematics, including quantification and the calculation of genetic frequencies, which became foundational to modern genetics.

GAs in their modern form were introduced by J. Holland [8] and his colleagues in 1975 as a way of solving hard optimization problems using stochastic search. It can be used to offer good solutions in a short amount of time, which is very efficient when the naïve search is not feasible.
They are very efficient in representing optimization problems due to their simple way of representing solutions as chromosomes which can take the form of an array, a matrix, a list, a tree, etc. Each chromosome is composed of genes coding its characteristics. The set of all possible chromosomes is called genotypic space which is closely related to the phenotypic space representing the solutions in their original form. GAs fall into the category of metaheuristics using an initial pool of solutions, and their usage differs from the other approximate methods in the way that solutions are combined to get new ones.
GAs rely on three core operations: selection, crossover, and mutation. In selection, the fittest individuals from a population are chosen based on their performance, ensuring that better solutions have higher chance of passing their traits to the next generation. Crossover then combines genetic material from selected parents to create new offspring, promoting the exploration of promising solution spaces by merging advantageous traits. Finally, mutation introduces small, random changes to some offspring, maintaining genetic diversity and preventing premature convergence to suboptimal solutions. These iterative processes mimic natural evolution, gradually improving the population's overall fitness until an optimal or satisfactory solution is achieved. This approach is widely used in optimization, machine learning, and engineering design to solve complex problems where traditional methods may struggle.

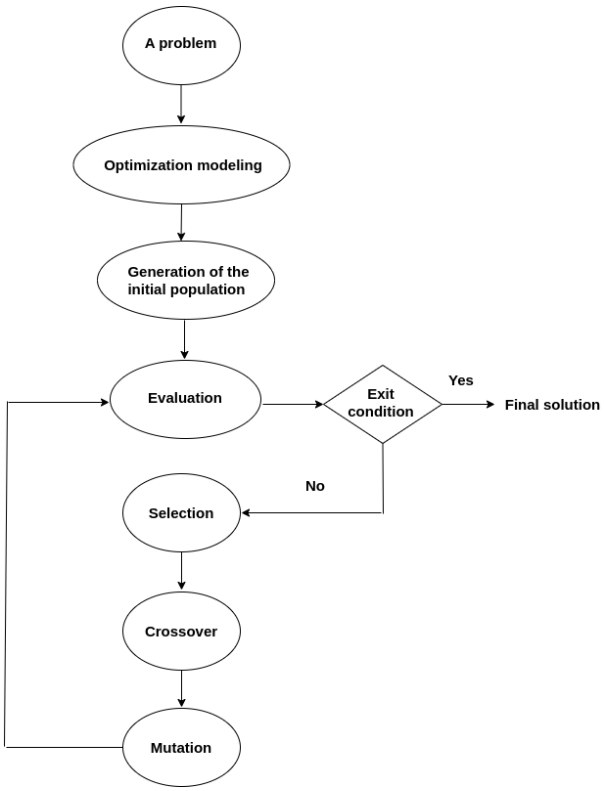


Figure 1: GA flowchart.

4 Multi-parent Genetic algorithms

A multi-parent Genetic Algorithm [6] is a variant of the standard Genetic Algorithm, which traditionally combines only two parents to produce offspring. In contrast, the multi-parent GA extends this approach by allowing the combination of three or more parents during the crossover process. This variant has demonstrated its ability to outperform the standard GA in terms of convergence speed, often achieving better results in a shorter amount of time. By leveraging the genetic material of multiple parents, the algorithm can explore a broader search space, and potentially learns to discover more optimal or near-optimal solutions.

5 Description of the proposed approach

The proposed algorithm is inspired by processor architecture principles, where the byte serves as the foundational unit. The core concept involves partitioning the truth table of a Boolean function into eight-bits bytes, enforcing strict balancedness by requiring each byte to contain exactly four zeros and four ones. Given a 256-bit output, this divides naturally into 32 bytes, each of which must conform to the 4-zero/4-one constraint. There are precisely 70 distinct valid configurations that satisfy this balanced condition for a single byte, as enumerated in Table 3. To implement this, we construct a chromosome-like encoding where each of the 32 bytes is represented as a gene, with each gene assuming an integer value between 0 and 69—corresponding to one of the 70 admissible byte patterns. This structured approach ensures both balancedness and efficient representation in our algorithmic framework.

Table 3: The 70 configurations for the encoding.

Configuration	**Encoding**
0, 0, 0, 0, 1, 1, 1, 1	0
0, 0, 0, 1, 0, 1, 1, 1	1
0, 0, 0, 1, 1, 0, 1, 1	2
0, 0, 0, 1, 1, 1, 0, 1	3
0, 0, 0, 1, 1, 1, 1, 0	4
0, 0, 1, 0, 0, 1, 1, 1	5
0, 0, 1, 0, 1, 0, 1, 1	6
0, 0, 1, 0, 1, 1, 0, 1	7
0, 0, 1, 0, 1, 1, 1, 0	8
0, 0, 1, 1, 0, 0, 1, 1	9
0, 0, 1, 1, 0, 1, 0, 1	10
0, 0, 1, 1, 0, 1, 1, 0	11
0, 0, 1, 1, 1, 0, 0, 1	12
0, 0, 1, 1, 1, 0, 1, 0	13
0, 0, 1, 1, 1, 1, 0, 0	14
0, 1, 0, 0, 0, 1, 1, 1	15
0, 1, 0, 0, 1, 0, 1, 1	16
0, 1, 0, 0, 1, 1, 0, 1	17
0, 1, 0, 0, 1, 1, 1, 0	18
0, 1, 0, 1, 0, 0, 1, 1	19
0, 1, 0, 1, 0, 1, 0, 1	20
0, 1, 0, 1, 0, 1, 1, 0	21
0, 1, 0, 1, 1, 0, 0, 1	22
0, 1, 0, 1, 1, 0, 1, 0	23
0, 1, 0, 1, 1, 1, 0, 0	24
0, 1, 1, 0, 0, 0, 1, 1	25
0, 1, 1, 0, 0, 1, 0, 1	26
0, 1, 1, 0, 0, 1, 1, 0	27
0, 1, 1, 0, 1, 0, 0, 1	28
0, 1, 1, 0, 1, 0, 1, 0	29
0, 1, 1, 0, 1, 1, 0, 0	30
0, 1, 1, 1, 0, 0, 0, 1	31
0, 1, 1, 1, 0, 0, 1, 0	32
0, 1, 1, 1, 0, 1, 0, 0	33
0, 1, 1, 1, 1, 0, 0, 0	34
1, 0, 0, 0, 0, 1, 1, 1	35
1, 0, 0, 0, 1, 0, 1, 1	36
1, 0, 0, 0, 1, 1, 0, 1	37
1, 0, 0, 0, 1, 1, 1, 0	38
1, 0, 0, 1, 0, 0, 1, 1	39
1, 0, 0, 1, 0, 1, 0, 1	40
1, 0, 0, 1, 0, 1, 1, 0	41
1, 0, 0, 1, 1, 0, 0, 1	42
1, 0, 0, 1, 1, 0, 1, 0	43
1, 0, 0, 1, 1, 1, 0, 0	44
1, 0, 1, 0, 0, 0, 1, 1	45
1, 0, 1, 0, 0, 1, 0, 1	46
1, 0, 1, 0, 0, 1, 1, 0	47
1, 0, 1, 0, 1, 0, 0, 1	48
1, 0, 1, 0, 1, 0, 1, 0	49
1, 0, 1, 0, 1, 1, 0, 0	50
1, 0, 1, 1, 0, 0, 0, 1	51
1, 0, 1, 1, 0, 0, 1, 0	52
1, 0, 1, 1, 0, 1, 0, 0	53
1, 0, 1, 1, 1, 0, 0, 0	54

Configuration	Encoding
1, 1, 0, 0, 0, 0, 1, 1	55
1, 1, 0, 0, 0, 1, 0, 1	56
1, 1, 0, 0, 0, 1, 1, 0	57
1, 1, 0, 0, 1, 0, 0, 1	58
1, 1, 0, 0, 1, 0, 1, 0	59
1, 1, 0, 0, 1, 1, 0, 0	60
1, 1, 0, 1, 0, 0, 0, 1	61
1, 1, 0, 1, 0, 0, 1, 0	62
1, 1, 0, 1, 0, 1, 0, 0	63
1, 1, 0, 1, 1, 0, 0, 0	64
1, 1, 1, 0, 0, 0, 0, 1	65
1, 1, 1, 0, 0, 0, 1, 0	66
1, 1, 1, 0, 0, 1, 0, 0	67
1, 1, 1, 0, 1, 0, 0, 0	68
1, 1, 1, 1, 0, 0, 0, 0	69

Hereafter, we present the key components of the proposed genetic algorithm:

1. **Pool size:** We use an initial pool of 500 solutions generated randomly, employing the encoding scheme described earlier.

2. **Fitness function:** The fitness function we use is [10]:

$$fitness(f) = Nl(f) + \frac{2^n - freq(max_{u\in\mathbb{F}_2^n}|W_f(u)|)}{2^n},$$

   The so defined fitness not only utilizes nonlinearity but also incorporates information from the Walsh spectrum, making the selection more efficient. This is achieved by minimizing the occurrences of the maximum value until it disappears, resulting in a new maximum value in the Walsh spectrum.

3. **Selection:** we use tournament selection where we pick 3 random individuals and we chose the fittest as a parent, repeating until the mating pool is filled (with a selection rate of 0.02%).

4. **Crossover:** The multi-parent Genetic Algorithm we use employs a uniform three-parent crossover mechanism. In this approach, each gene in the offspring has an equal probability ($\frac{1}{3}$ chance) of being inherited from any of the three parents. To facilitate this process, a mask is used, which can take on three possible values: 0, 1, or 2. Each value in the mask corresponds to one of the three parents and occurs with the same frequency, ensuring fairness in the selection process. This mechanism generates six offspring in each crossover operation, significantly enhancing the diversity of the population.

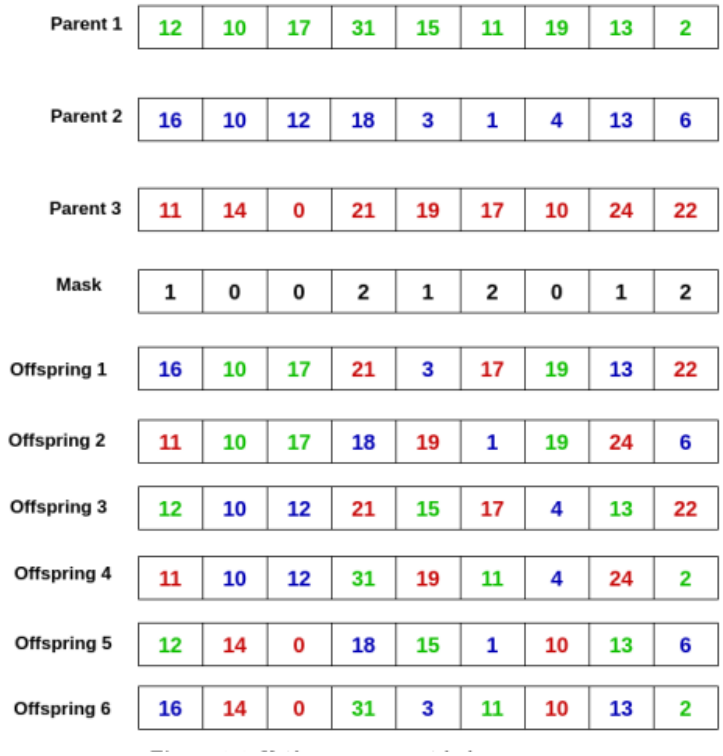


Figure 2: Three-parent uniform crossover.

5. **Mutation:** a random gene is chosen for mutation with a rate of 0.01. The value of the gene is replaced by one of the remaining 69 possible values.

The design of the encoding scheme inherently ensures balancedness, meaning that the genetic material from all the three parents is evenly distributed across the offspring. As a result, there is no need for post-crossover corrections or adjustments, which simplifies the algorithm and improves its computational efficiency.

By leveraging this multi-parent crossover strategy, the proposed algorithm achieves a more extensive exploration of the search space, leading to faster convergence and higher-quality solutions compared to traditional two-parent crossover methods. This approach is particularly advantageous in complex optimization problems where diversity and exploration are critical to avoiding local optima.

6 Experimental results

To trial the effectiveness of the proposed GA, we conducted a series of experiments.
In the first experiment, the GA was able to find the Boolean function with the genotype:
39, 4, 15, 64, 49, 60, 11, 27, 19, 0, 47, 7, 49, 35, 55, 61, 27, 58, 37, 36, 60, 0, 41, 67, 55, 67, 31, 24, 59, 5, 33, 8
This solution is equivalent to the phenotype:
1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1,
1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0,
1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0,
1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0,
1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0,
1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1,
0, 0, 0, 0, 1, 0, 1, 1, 1, 0
This solution has a high nonlinearity of 116.
In the second experiment, the GA reached the Boolean function with the genotype:

54, 48, 22, 39, 15, 37, 5, 65, 46, 64, 33, 53, 34, 38, 59, 44, 18, 55, 0, 68, 61, 41, 18, 40, 37, 66, 43, 13, 42, 35, 18, 36
This solution is equivalent to the phenotype:
0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0
This solution has also a high nonlinearity of 116.

In the third experiment, the GA found the Boolean function with the genotype:
13, 69, 49, 0, 30, 40, 21, 57, 28, 48, 56, 25, 10, 12, 27, 44, 18, 9, 17, 27, 57, 34, 20, 65, 31, 12, 62, 10, 46, 66, 20, 34
This solution is equivalent to the phenotype:
0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1
Again, this solution has a high nonlinearity of 116.

In the fourth experiment, the GA determined the Boolean function with the genotype:
23, 56, 36, 46, 38, 33, 53, 28, 18, 10, 45, 13, 46, 53, 9, 48, 0, 59, 38, 11, 3, 16, 31, 20, 63, 11, 60, 27, 27, 1, 40, 45
This solution is equivalent to the phenotype:
0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1
Once again, this solution has a high nonlinearity of 116.

It is important to mention that so far, there is no proof of the existence of a balanced Boolean function with eight variables having a nonlinearity of 118 [10].

Table 4 presents a comparison of the best-achieved solutions with those of the literature, in term of nonlinearity.

Refrence	Nonlinearity
[5]	114
[13]	116
[2]	116
Proposal	116

Table 4: Comparison in term of nonlinearity

For instance, in [13] [2], the authors generated their solutions using bent functions, which are non-balanced Boolean functions with high nonlinearity. They made minor modifications to these functions, essentially working within the neighborhood of a bent function. This raises questions about their security, as such solutions lack true randomness and may not be as secure as they appear. In contrast, the proposed approach generates Boolean functions from scratch using Evolutionary Intelligence, which introduces greater randomness. Indeed, unlike methods that rely on algebraic properties, the proposed

GA disperses the solutions more widely across the search space. Furthermore, the devised GA leverages the inherent randomness of stochastic optimization, thanks to the proposed encoding, resulting in more robust and less predictable BF solutions.

7 Conclusion

The use of genetic algorithms is a powerful approach for generating random Boolean functions with desirable properties, such as high nonlinearity. Boolean functions play a critical role in enhancing security by introducing complexity and filtering mechanisms, making it more difficult for attackers to decipher cryptographic systems.
The proposed approach, which specifically targets balanced Boolean functions that are widely used in cryptography, was able to generate high-quality Boolean functions efficiently.
In future work, we plan to explore improvements to the proposed genetic algorithm by incorporating additional metaheuristics, such as Stochastic Local Search, to further enhance the quality of the generated functions.

References

[1] Menouar Boulif. Genetic algorithm encoding representation for graph partitioning problems. In *2010 International Conference on Machine and Web Intelligence, ICMWI 2010 - Proceedings*, pages 288–291, 10 2010.

[2] Linda Burnett, Andrew Clark, Ed Dawson, and William Millan. Simpler methods for generating better boolean functions with good cryptographic properties. 29, 01 2004.

[3] Anne Canteaut. *Lecture Notes on Cryptographic Boolean Functions*. Inria, Paris, France, 2016.

[4] Viviane Carmo and Lilian Martins. *Wallace, Darwin, and the Relationship Between Species and Varieties (1858)*, pages 147–161. 10 2023.

[5] A. Dimovski and D. Gligoroski. Generating highly nonlinear boolean functions using a genetic algorithm. In *6th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Service, 2003. TELSIKS 2003.*, volume 2, pages 604–607 vol.2, 2003.

[6] A. E. Eiben, P. E. Raué, and Zs Ruttkay. Genetic algorithms with multi-parent recombination. pages 78–87, 1994.

[7] Prakash Gorroochurn. *Darwin and the Origin of Species*, pages 55–146. 12 2024.

[8] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, pages 203–205. 1992.

[9] William Millan. How to improve the nonlinearity of bijective s-boxes. In Colin Boyd and Ed Dawson, editors, *Information Security and Privacy*, pages 181–192, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[10] Stjepan Picek, Roberto Santana, and Domagoj Jakobovic. Maximal nonlinearity in balanced boolean functions with even number of inputs, revisited. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 3222–3229, 2016.

[11] Franz Rothlauf. *Representations for Genetic and Evolutionary Algorithms*, volume 104, pages 73 – 96. 01 2006.

[12] Kumara Sastry, David Goldberg, and Graham Kendall. *Genetic Algorithms*, pages 97–125. Springer US, Boston, MA, 2005.

[13] Erol Özçekiç, Selçuk Kavut, and Hakan Kutucu. Genetic approach to improve cryptographic properties of balanced boolean functions using bent functions. *Computers*, 12:159, 08 2023.

Enhancing Learning Management Systems with AI: Recommendations from Moodle Usage

Razika LOUNAS[1]

[1] ***LIMOSE Laboratory, Computer Science Department, Faculty of Sciences University of M'hamed Bougara of Boumerdes, Independency Avenue, 35000 Algeria,*** *razika.lounas@univ-boumerdes.dz*

Abstract

Learning Management Systems (LMS) are defined as web-based platforms designed to facilitate the planning, delivery, management, and assessment of educational courses and training programs. These systems provide a structured virtual environment where learners, instructors, and administrators can interact through various tools. With the advent of artificial intelligence, several approaches have been explored to reshape and adapt LMS functionalities, enhancing their ability to meet the growing demand for AI-based tools in various pedagogical activities. Among existing LMS platforms, Moodle is one of the most widely used. It facilitates online education and offers a rich array of tools. However, feedback from computer science students highlights both its strengths and areas for improvement, particularly in terms of usability, engagement, and automation. This paper analyzes student feedback and examines how artificial intelligence (AI) can address common challenges in Moodle. Furthermore, it discusses potential AI-driven enhancements based on student insights and proposes recommendations for future research and implementation.

Keywords: Learning Management systems, Moodle, Student's perception, Artificial Intelligence.

1 Introduction

The rapid advancement of information and communication technologies (ICTs) has significantly transformed various aspects of daily life, including education. Over the past decades, learning environments have evolved in response to major technological shifts, such as the rise of the internet, the introduction of Web 2.0 applications, and more recently, the emergence of artificial intelligence (AI)-driven tools [13, 31]. These developments have reshaped how educational content is delivered, accessed, and personalized.

In particular, Learning Management Systems (LMS) have become essential in modern education, enabling institutions to manage courses, facilitate communication, and support online learning. LMS platforms accommodate various teaching and learning methodologies, offering tools for content delivery, assessments, collaboration, and administrative management. However, as technology continues to advance, LMS platforms must adapt to evolving educational needs and user expectations.

The COVID-19 pandemic accelerated the global reliance on LMS platforms, highlighting both their strengths and limitations. Institutions worldwide were compelled to transition rapidly to online learning, revealing challenges related to usability, student engagement, automation, and scalability [22]. Among the widely adopted LMS platforms, Moodle stands out due to its open-source nature, flexibility, and extensive adoption by universities and institutions worldwide, including in Algeria. Its features, such as collaborative tools, integration with external applications, and customizable learning environments, contribute to its success [11]. However, feedback from computer science students suggests areas for improvement, particularly in terms of usability, functionality, and interactivity.

This paper explores the evolution of LMS platforms and examines how AI-driven enhancements could address current limitations in Moodle. By leveraging intelligent tutoring systems, automated grading, AI-powered chatbots, and adaptive learning algorithms, AI has the potential to enhance personalized learning, improve engagement, and optimize administrative tasks. The insights presented in this paper are based on reflections and feedback from Master's level computer science students, who offer valuable perspectives on potential improvements at various levels, including functionality, usability, and advanced AI-driven features.

This paper aims to:

- Provide an overview of LMS evolution and highlight challenges in Moodle.
- Analyze student feedback to identify key areas for improvement.

- Propose AI-driven enhancements that could improve interactivity, automation, and personalization.

The rest of this paper is organized as follows: Section 2 presents an overview on LMS and AI integration. Section 3 presents the methodology used for gathering student insights. Section 4 outlines key findings and proposed AI-based recommendations. Finally, Section 5 concludes the paper.

2 Background

2.1 LMS Overview

A Learning Management systems (LMS) is defined as virtual environment that aims to simulate face-to-face learning environments with the use of Information Technology applications, and customizable learning environments, contribute to its success [27]. The interactions among different users (student, teacher, administrative) happen through the system that provides synchronous or asynchronous communication tools. LMS provide a large plethora of functions allowing the creation of different strategies and learning modes and to engage learners.

Figure 1: LMS functions

Learning Management System (LMS) are designed to include several components that support online education course administration, and student engagement [29, 19]. The main functions of an LMS are depicted in Figure 1. Course registration and administration module manages user roles, course enrollment, group organization, scheduling, and authentication. Content delivery enables both synchronous and asynchronous learning, supporting various formats such as videos, interactive materials, and allow students to engage flexibly with course content by offering possibility to connect with different devices. To monitor progress, LMS platforms integrate tracking and learning analytics, providing insights into student engagement, course completion rates, and attendance. Another essential component is collaboration and communication, which fosters interaction through chat systems, discussion forums, shared calendars, and resource-sharing tools, while also offering advanced collaboration features such as wikis and co-creation workspaces. Effective course material management ensures that educational resources are organized, and accessible, integrating with different types of repositories. Finally, the assessment and evaluation module facilitates automated quizzes, peer reviews, Exams, and competency-based assessments, allowing instructors to efficiently measure learning outcomes. Together, these components create a comprehensive digital learning environment, enabling institutions to deliver engaging, interactive, and structured online education.

2.2 Artificial Intelligence in LMS

With the advent of artificial intelligence applications, learning management systems through their components have evolved to adapt to the new requirements. Several efforts have been made to improve the

LMS function with artificial intelligence applications and tools [9, 7, 1].

This section aims to present the current utilization of AI in Learning Management Systems by exploring the interaction of AI with every aspect of LMS and providing references supporting the integration and elucidating the benefits and challenges of such endeavor. Table 1 illustrates AI enhancements for every LMS components with supporting references.

Table 1: AI Enhancements for LMS Components

LMS Component	AI Enhancements
Registration and Administration	AI-based automated student registration [18], intelligent course recommendations [25, 21], and adaptive scheduling [5, 15].
Content Delivery	Personalized learning paths [33, 16], adaptive content recommendations [8, 30], and real-time AI tutoring [12, 34].
Tracking and Analytics	Predictive analytics for student performance [20, 6], engagement tracking [14].
Collaboration	Chatbots for student assistance, automated discussion moderation, and intelligent group formation [26, 24].
Course Material Management	Assisted content organization, tagging and summarization, and intelligent resource recommendations [35, 23, 4].
Assessment and Evaluation	Automated essay scoring and plagiarism detection, [2, 10].

The table 1 illustrates the availability of research on the integration of AI tools and applications to enhance learning management systems (LMS) at different levels. It is worth noting that the integration of artificial intelligence in LMS has shifted significantly with the advent of generative AI. Indeed, tools like ChatGPT have revolutionized LMS functionalities with their unique capabilities in natural language understanding and content generation [28]. These advancements have enabled more interactive and adaptive learning experiences facilitating AI-based tutoring and assistance, assessment, and personalization.

Among existing platforms, Moodle is a fundamental tool in modern education, particularly in computer science programs. It facilitates course management, content delivery, and student-teacher interaction. AI applications have been integrated into various aspects of LMS platforms through several initiatives [17]. However, student feedback continues to highlight usability concerns, a lack of real-time support, and difficulties in navigating and engaging with course materials. This indicates that the integration of AI tools remains an active research area. The aim of this study is to analyze feedback from computer science students on Moodle and explore how AI technologies can enhance its functionality.

3 Methodology

This section outlines the research methodology used in this study. It begins by describing the research model, followed by details on the context, participants, and data collection.

3.1 Research Model

This study is based on a qualitative research approach. Qualitative research focuses on understanding meanings, experiences, and concepts from the perspective of participants. Its objective is to gain insights into social phenomena, events, or complex issues. This type of research was adopted because it is best suited to capture the different aspects of users' reactions to the use of Moodle platform enabling the evaluation of its strengths, weaknesses, and potential improvements [3].

To gather data, the study employs a questionnaire, which is one of the most commonly used tools in educational technology research. The questionnaire consists of open-ended questions, allowing assessment and qualitative insights. The collected responses are analyzed to identify trends, user satisfaction, and key areas where AI-driven enhancements could improve the platform.

3.2 Context and Participants

The study was conducted with final-year Master's students from the Computer Science Department of Boumerdes University. A total of 54 students participated in the survey. As young engineers and future professionals, their insights are particularly valuable in identifying technical and functional improvements for the Moodle platform. The questionnaire covered three main areas:

- Advantages of Moodle: Students were asked to identify key benefits of using the platform.
- Challenges and Limitations: Participants highlighted areas they think about as Moodle limitaions.
- Suggestions for Improvement: As computer science students, participants provided some directions to enhance the platform.

The collected data was analyzed qualitatively to identify recurring patterns and common themes. Additionally, percentage distributions were used to highlight the frequency of reported advantages and challenges. The results provide insights and a base for recommendations for improving Moodle's usability and performance.

4 Results

This section presents students' feedback related to the questions about advantages, inconveniences, and possible improvements for Moodle. The last point serves as a basis to propose recommendations for improvements by including AI tools and applications.

4.1 Strengths of the platform

The number of students responding to the question about Moodle advantages is twenty six (26 students). The following key points emerged about the platform strengths: flexibility and accessibility, collaboration, pedagogical support, security and data protection, and digital skills development. The data are reported on Table 2.

The majority of students (88%) pointed out the advantage of the platform flexibility and accessibility. The platform is open-source platform, free and highly customizable. Moodle supports distance learning by eliminating spatial and temporal constraints. The platform benefits also from an active global community that provides extensive documentation and support. The platform is widely adopted by universities, high schools, and businesses. The student also highlighted the advantage of collaboration. Indeed, the platform enables extensive communication between students and teachers. with a rich range of features from simple messaging and forums to more advanced features such as wikis. Coordination is facilitated by planning tools such as shared agendas. Co-production, information sharing, and knowledge management are also features that enhance collaborative work within the learning process, fostering an interactive and engaging educational environment.

Pedagogical Support is mentioned as an advantage of Moodle. Diverse resource formats and evaluation methods are available, enabling teachers to monitor student progress, provide feedback, track behaviors, and digitally assess assignments. The platform is widely used across various educational and organizational contexts, fostering a large community of users and shared resourcese. Its features help address challenges such as student absences by maintaining course records before, during, and after sessions. With a wide range of integrated tools, online assessment methods, and compatibility with institutional learning environments, Moodle enhances both teaching effectiveness and student engagement. Several students mentioned also security and data protection. The platform offers various mechanisms to protect user data and ensure safe usage. It includes access control features to prevent unauthorized access making Moodle a reliable choice for institutions that prioritize data protection in online learning environments.

Finally, some students (15%) mentioned digital Skills development as an advantage of Moodle since from their point of view Moodle contributes to the development of students' digital skills by providing an environment that encourages the use of technology in learning. The interaction with the platform allow students to enhance their ability to navigate digital tools, manage online resources, and engage in technology-assisted learning.

Table 2: Advantages of Moodle as reported by students

Advantage	**% of Students mentioning it**
Flexibility and accessibility	88%
Collaboration	50%
Pedagogical support	65%
Security and data protection	38%
Digital skills development	15%

4.2 Weaknesses of the platform

The number of students who responded to the question of the platform weaknesses is 28. It is a different group from the students who responded to the question of advantages. According to students feedback, the platform suffers from several drawbacks that are categorized into four categories as depicted by Table 3: Interface and Usability Issues, lack of personalization, socialization issues, and integration and compatibility problems.

The issues with the interface and use were the most mentioned weakness. Students frequently expressed dissatisfaction with Moodle's interface, describing it as unattractive, un-intuitive, and outdated. Many reported difficulties navigating the platform. There is a strong demand for a more modern and ergonomic design. Additionally, slow system performance and long loading times make access to learning materials difficult. Several students expressed the need for technical support and training to help users better navigate and utilize the platform's features effectively. The second raised issue is related to personalization: Students highlighted Moodle's limited customization options as a drawback. The platform does not sufficiently adapt to students' specific needs, making it difficult to adapt the learning experience. Many users expressed a request for personalization, both in terms of content and page appearance, to create a more engaging and individualized environment.

Students also raised the need for socialization: they highlighted the lack of features that enhance social interaction, particularly the need for better support for group work. They also pointed out the absence of more engaging activities inspired by social networks, such as the ability for students to publish content and collaborate on co-produced materials. Problems related to integration and compatibility are also reported by students. Students reported difficulties in integrating Moodle with external tools such as Google Drive, Slack, and Zoom, which are highly used in their learning activities. Issues with mobile compatibility are also mentioned , noting that the platform does not always function easily on smartphones and the lack of proper synchronization across devices makes it difficult to use.

Table 3: Weaknesses of Moodle as reported by students

Weakness	**% of students mentioning it**
Interface and usability issues	88%
Lack of personalization	46%
Socialization issues	23%
Integration and compatibility problems	23%

4.3 Suggestions for Moodle improvements

The objective of this section is first to present students' suggestions according to the raised issues. These suggestions with the weaknesses that are pointed out by students serve a a basis to some recommendations about the use of AI to enhance the platform.

4.3.1 Suggesions from students

The word cloud (see Figure 2) visually represents the most frequently mentioned suggestions from students regarding Moodle's improvement. The larger words indicate the most common themes, highlighting key areas of concern and potential enhancements. From this visualization, it is evident that students prioritize aspects such as interface modernization, personalization, integration with other tools, and per-

Figure 2: Word cloud from students suggestions

formance optimization. These insights reinforce the need for a more intuitive and interactive learning environment that aligns with user expectations.

While the word cloud effectively illustrates the most recurring themes in students' feedback, it is important to note that some valuable suggestions may not appear prominently due to lower mention frequency. For instance, proposals such as integrating artificial intelligence for personalized learning, providing instant feedback, or enhancing course recommendations were mentioned by a smaller number of students but still represent innovative directions for improvement. These insights, though less common, highlight valuable enhancements that could significantly impact Moodle's users experience.

4.3.2 Recommendations for Improvements with Artificial Intelligence

To address the issues raised by students about Moodle, AI can be leveraged in several ways:

- Interface and usability issues: In response to this issue, the incorporation of artificial intelligence could be used to predict user behavior in order to anticipate and address potential barriers to use in UI design. This will not only improve the user experience, but also promote the development of UI design in a more user-friendly and intelligent direction. In addition, advanced concepts such as immersive environments in LMS can be explored [32]. Furthermore, virtual assistants to help navigating the platform could be developed to enhance its efficiency and usability.

- Personalization: To enhance the personalization aspect in Moodle and provide a learning experience adapted for students, several artificial intelligence approaches may be explored. While the course are generally mandatory, recommendations may occur at the level of resources. In addition, the use of expert systems and intelligent tutors that adapt to user needs are also a perspective suitable to Moodle. Studies on AI in education show that intelligent tutoring systems and automated assessment tools improve student engagement and learning outcomes. These tools can analyze student performance and recommend personalized study materials, and adjust difficulty levels based on student progress.

- Socialization: Several research directions and applications of AI are suitable to this concern. The use of community detection tools may be used to form group discussions about academic topics or homework for students. In addition, chatbots and discussion moderators can facilitate meaningful conversations, ensuring that discussions remain relevant and productive. Sentiment analysis tools may also be integrated to detect and address students' concerns in forums, promoting a more engaging and supportive learning environment.

- Integration and Compatibility Problems: AI can help with compatibility issues by automating content adaptation for different devices and browsers. Additionally, artificial intelligence can facilitate communication between Moodle and other platforms, ensuring smooth data exchange and reducing technical disruptions for both students and educators.

5 Conclusion

Learning Management Systems (LMS) have revolutionized digital education by providing a structured and interactive environment for learning. This study explored LMS platforms, particularly Moodle through students' perception. The platform offers numerous advantages such as accessibility, collaboration, pedagogical support, security, and flexibility. However, student feedback reveals critical areas for improvement, particularly user interface design, personalization, social interaction, and integration with external tools.

The integration of Artificial Intelligence (AI) into LMS platforms presents an opportunity to address these challenges and enhance the overall learning experience. AI applications and features such as adaptive learning, intelligent tutoring systems, automated content recommendations, and predictive analytics can significantly improve the students' experience in LMS. Furthermore, the emergence of Large Language Model (LLM) chatbots like ChatGPT can reshape student interactions within learning environments. This paper provides some recommendations for the use of AI in Moodle. Future directions of this research will addresss some limitations such as the inclusion of teachers perception, complete with a quantitative research, and the effective development of improvements based on AI.

References

[1] Nouf Aldahwan and Norah Alsaeed. Use of artificial intelligent in learning management system (lms): a systematic literature review. *International Journal of Computer Applications*, 175(13):16–26, 2020.

[2] Abdulaziz Salamah Aljaloud, Diaa Mohammed Uliyan, Adel Alkhalil, Magdy Abd Elrhman, Azizah Fhad Mohammed Alogali, Yaser Mohammed Altameemi, Mohammed Altamimi, and Paul Kwan. A deep learning model to predict student learning outcomes in lms using cnn and lstm. *IEEE Access*, 10:85255–85265, 2022.

[3] Ann Blandford, Dominic Furniss, and Stephann Makri. *Qualitative HCI research: Going behind the scenes*. Morgan & Claypool Publishers, 2016.

[4] Hui Chen, Chuantao Yin, Rumei Li, Wenge Rong, Zhang Xiong, and Bertrand David. Enhanced learning resource recommendation based on online learning style model. *Tsinghua science and technology*, 25(3):348–356, 2019.

[5] Prithviraj Dasgupta and Deepak Khazanchi. Adaptive decision support for academic course scheduling using intelligent software agents. *International Journal of Technology in Teaching and Learning*, 1(2):63, 2005.

[6] Ashish Dutt and Maizatul Akmar Ismail. Can we predict student learning performance from lms data? a classification approach. In *Proceedings of the 3rd International Conference on Current Issues in Education (ICCIE 2018)*, pages 24–29. Atlantis Press, 2019.

[7] Mohd Elmagzoub Eltahir and Frdose Mohd Elmagzoub Babiker. The influence of artificial intelligence tools on student performance in e-learning environments: Case study. *Electronic Journal of e-Learning*, 22(9):91–110, 2024.

[8] Digna S Evale. Learning management system with prediction model and course-content recommendation module. *Journal of Information Technology Education: Research*, 16(1), 2017.

[9] Mehmet Fırat. Integrating ai applications into learning management systems to enhance e-learning. *Instructional Technology and Lifelong Learning*, 4(1):1–14, 2023.

[10] A Jauhar Fuad, Amar Kukuh Wicaksono, M Auzai Aqib, M Arif Khoiruddin, Abbas Sofwan Matla'Il Fajar, and Khoirul Mustamir. Ai hybrid based plagiarism detection system creation. In *2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pages 1495–1500. IEEE, 2024.

[11] Sithara HPW Gamage, Jennifer R Ayres, and Monica B Behrend. A systematic review on trends in using moodle for teaching and learning. *International journal of STEM education*, 9(1):9, 2022.

[12] Cecilia Estela Giuffra Palomino, Ricardo Azambuja Silveira, and Marina Keiko Nakayama. An intelligent lms model based on intelligent tutoring systems. In *Intelligent Tutoring Systems: 12th International Conference, ITS 2014, Honolulu, HI, USA, June 5-9, 2014. Proceedings 12*, pages 567–574. Springer, 2014.

[13] Abid Haleem, Mohd Javaid, Mohd Asim Qadri, and Rajiv Suman. Understanding the role of digital technologies in education: A review. *Sustainable Operations and Computers*, 3:275–285, 2022.

[14] Curtis R Henrie, Robert Bodily, Ross Larsen, and Charles R Graham. Exploring the potential of lms log data as a proxy measure of student engagement. *Journal of Computing in Higher Education*, 30:344–362, 2018.

[15] Bui Thi Thanh Huong, Tran Van Cong, Ton Quang Cuong, and Nguyen Duc Nguyen. Smart schedule: Training management approach in the digital education era. In *Digital Education for the 21st Century*, pages 283–305. Apple Academic Press, 2021.

[16] Delara Jafari and Zahra Shaterzadeh-Yazdi. Transforming education with ai: The development of a personalized learning algorithm for individual learning styles. *Journal of Algorithms and Computation*, 56(2):135–150, 2024.

[17] Devkan Kaleci. Integration and application of artificial intelligence tools in the moodle platform: A theoretical exploration. *Journal of Educational Technology and Online Learning*, 8(1):100–111, 2025.

[18] Vishesh Kalvakurthi, Aparna S Varde, and John Jenq. Hey dona! can you help me with student course registration? *arXiv preprint arXiv:2303.13548*, 2023.

[19] Nurul Nadirah Mohd Kasim and Fariza Khalid. Choosing the right learning management system (lms) for the higher education institution context: A systematic review. *International Journal of Emerging Technologies in Learning*, 11(6), 2016.

[20] Majid Khan, Saba Naz, Yashir Khan, Muneeb Zafar, Maqbool Khan, and Giovanni Pau. Utilizing machine learning models to predict student performance from lms activity logs. *IEEE Access*, 11:86953–86962, 2023.

[21] Hassan Khosravi, Kirsty Kitto, and Joseph Jay Williams. Ripple: A crowdsourced adaptive platform for recommendation of learning activities. *arXiv preprint arXiv:1910.05522*, 2019.

[22] Christopher M Lee. Impact of learning management systems (lms) toward students during the covid-19 pandemic. *Available at SSRN 4854926*, 2023.

[23] Debabrata Mahapatra, Ragunathan Mariappan, Vaibhav Rajan, Kuldeep Yadav, and Sudeshna Roy. Videoken: Automatic video summarization and course curation to support learning. In *Companion Proceedings of the The Web Conference 2018*, pages 239–242, 2018.

[24] Elizaphan M Maina, Robert O Oboko, and Peter W Waiganjo. Using machine learning techniques to support group formation in an online collaborative learning environment. *International Journal of Intelligent Systems & Applications*, 9(3):26–33, 2017.

[25] Dina Fitria Murad, Yaya Heryadi, Bambang Dwi Wijanarko, Sani Muhamad Isa, and Widodo Budiharto. Recommendation system for smart lms using machine learning: a literature review. In *2018 international conference on computing, engineering, and design (ICCED)*, pages 113–118. IEEE, 2018.

[26] Giacomo Nalli, Daniela Amendola, and Serengul Smith. Artificial intelligence to improve learning outcomes through online collaborative activities. In *European Conference on e-Learning*, pages 475–479. Academic Conferences International Limited, 2022.

[27] Paulo Cristiano de Oliveira, Cristiano Jose Castro de Almeida Cunha, and Marina Keiko Nakayama. Learning management systems (lms) and e-learning management: an integrative review and research agenda. *JISTEM-Journal of Information Systems and Technology Management*, 13(2):157–180, 2016.

[28] Sameer Qazi, Muhammad Bilal Kadri, Muhammad Naveed, Bilal A Khawaja, Sohaib Zia Khan, Muhammad Mansoor Alam, and Mazliham Mohd Su'ud. Ai-driven learning management systems: Modern developments, challenges and future trends during the age of chatgpt. *Computers, Materials & Continua*, 80(2), 2024.

[29] Rabiman Rabiman, Muhammad Nurtanto, and Nur Kholifah. Design and development e-learning system by learning management system (lms) in vocational education. *Online Submission*, 9(1):1059–1063, 2020.

[30] Nisha S Raj and VG Renumol. A rule-based approach for adaptive content recommendation in a personalized learning environment: An experimental analysis. In *2019 IEEE tenth international conference on technology for education (T4E)*, pages 138–141. IEEE, 2019.

[31] Ido Roll and Ruth Wylie. Evolution and revolution in artificial intelligence in education. *International journal of artificial intelligence in education*, 26:582–599, 2016.

[32] Serhiy O Semerikov, Tetiana A Vakaliuk, Iryna S Mintii, Vita A Hamaniuk, Olha V Bondarenko, Pavlo P Nechypurenko, Svitlana V Shokaliuk, and Natalia V Moiseienko. Immersive cloud-based educational environment of the university: Design principles. CEUR Workshop Proceedings, 2024.

[33] MJKMS Somasundaram, KA Mohamed Junaid, and Srinivasan Mangadu. Artificial intelligence (ai) enabled intelligent quality management system (iqms) for personalized learning path. *Procedia Computer Science*, 172:438–442, 2020.

[34] Marwa Soudi, Esraa Ali, Maha Bali, and Nihal Mabrouk. Generative ai-based tutoring system for upper egypt community schools. In *Proceedings of the 2023 Conference on Human Centered Artificial Intelligence: Education and Practice*, pages 16–21, 2023.

[35] Christopher CY Yang, Irene YL Chen, Gökhan Akçapınar, Brendan Flanagan, and Hiroaki Ogata. Using a summarized lecture material recommendation system to enhance students' preclass preparation in a flipped classroom. *Educational Technology & Society*, 24(2):1–13, 2021.

Improved Symbiotic Organism Search Algorithm for Biomedical Data Clustering

Saida Ishak Boushaki[1], Nabila Rahmoune[2], Omar Bendjeghaba[3], and Sadjia Lahiani[3]

[1] *LIMOSE Laboratory, University Mhamed Bougara, Boumerdes, Algeria,* s.boushaki@univ-boumerdes.dz
[2] *LIMOSE Laboratory, University Mhamed Bougara, Boumerdes, Algeria,* rah.nabila@yahoo.com
[3] *(LREEI), University M'Hamed Bougara Boumerdes, 35000, Algeria,* bendjeghaba@univ-boumerdes.dz
[4] *Biological Resources Valorization and Conservation Laboratory "VALCORE", Université M'Hamed Bougara Boumerdès, Avenue de l'indépendance, 35000, Boumerdes, Algeria,* sadjialahiani@yahoo.fr

Abstract

Biomedical datasets have grown exponentially with advancements in digital data acquisition and storage technologies. This explosion of data has heightened the need for effective methods to uncover actionable insights, a task central to the field of data mining. Among data mining techniques, clustering holds particular importance for its ability to group data into meaningful subsets, revealing underlying patterns and critical features. In this paper, we present an improved version of the Symbiotic Organisms Search (SOS) algorithm, augmented with a novel Fitness-Distance Balance (FDB) selection method. This enhanced algorithm is specifically adapted for the clustering of biomedical data. The SOS algorithm, inspired by natural symbiotic interactions, excels at exploring solution spaces to locate global optima. With the addition of the FDB method, the algorithm achieves superior efficiency and performance, overcoming challenges inherent in traditional clustering methods.

Keywords: Clustering, Symbiotic Organism Search Algorithm, Fitness-Distance Balance (FDB) selection, I index, Biomedical data, Datamining, Metaheuristic.

1 Introduction

The rapid advancements in digital technologies have led to an unprecedented growth of large-scale datasets, particularly in the biomedical field. Extracting actionable insights from these datasets is a core objective of data mining, a discipline that involves analyzing data to uncover hidden relationships and summarize information in innovative and practical ways. Clustering plays a pivotal role in data mining, functioning as an exploratory analysis technique that organizes data into meaningful groups or clusters [7]. These clusters reveal intrinsic data patterns, enabling better understanding and characterization of datasets. Biomedical research, in particular, has benefitted from clustering algorithms, which are extensively used to analyze gene expression data. Such analyses help identify groups of genes exhibiting similar behavior, contributing to insights into complex biological processes and facilitating the development of personalized medical treatments [1] . Clustering techniques can be broadly categorized into hierarchical and partition-based approaches. Hierarchical clustering builds a tree-like structure (dendrogram) of clusters, providing multilevel exploration and rich visualizations [1]. Despite its descriptive power, it suffers from high computational complexity, which scales quadratically in the best-case scenario. Partition-based clustering, exemplified by the popular K-means algorithm, offers greater computational efficiency due to its linear complexity. However, K-means often struggles with random initialization, which can result in suboptimal solutions [5]. Metaheuristics, inspired by natural phenomena, have gained prominence for their ability to solve complex optimization problems like clustering [8], [2]. Among these, the Symbiotic Organisms Search (SOS) algorithm, introduced by Cheng and Prayogo [3], has shown great promise. The SOS algorithm models symbiotic interactions in nature, enabling efficient exploration of solution spaces without requiring complex parameter tuning.

This study proposes an improved SOS algorithm for clustering biomedical data, integrating a Fitness-Distance Balance (FDB) selection method [4]. The enhanced algorithm leverages the strengths of SOS while addressing its limitations to achieve superior clustering outcomes. The FDB method enhances the search process by balancing solution quality and diversity, leading to improved convergence and clustering accuracy.

The paper is organized as follows. Section 2 introduces the theoretical underpinnings of the proposed approach, including a detailed overview of the SOS algorithm and its optimization mechanisms. Section 3 presents the improved algorithm, providing an in-depth explanation of its core components and workflow. Section 4 discusses the results of numerical experiments, highlighting the algorithm's effectiveness through a comparative analysis with existing techniques. Section 5 concludes the study by summarizing key findings and offering recommendations for future research directions. .

2 THE SYMBIOTIC ORGANISMS SEARCH (SOS) ALGORITHM

The Symbiotic Organisms Search (SOS) algorithm is a population-based metaheuristic inspired by the symbiotic relationships found in nature, such as mutualism, commensalism, and parasitism. SOS mimics how organisms interact to improve their chances of survival and adapt to their environment. Like traditional metaheuristics, the initial population in the SOS algorithm is generated randomly, with each organism representing a potential solution to the problem at hand. During each iteration, the algorithm simulates three primary types of symbiotic relationships: mutualism, commensalism, and parasitism, to explore and optimize the solution space effectively.

- Mutualism Phase

In the mutualism phase, a new solution is derived for Oi (representing the i-th organism in the ecosystem) and Oj (randomly selected from the population) by simulating mutualistic symbiosis. This interaction models the mutual benefit between Oi and Oj, resulting in an updated solution calculated as follows:

$$O_{i_{new}} = O_i + rand(0,1) \times (O_{best} - mutuel\_vec \times bef_1) \tag{1}$$

$$O_{j_{new}} = O_j + rand(0,1) \times (O_{best} - mutuel\_vec \times bef_2) \tag{2}$$

Here, rand(0,1) represents a vector of random numbers uniformly distributed within the range [0,1]. The benefit factors bef1 and bef2 are integers randomly assigned as either 1 or 2, indicating the level of benefit received by each organism. The interaction between organisms Oi and Oj is represented by a mutual vector, defined as:

$$O_{i_{new}} = O_i + rand(-1,1) \times (O_{best} - O_j) \tag{3}$$

The mutual vector represents the highest degree of adaptation, serving as the target point for improving the fitness of both organisms. Accordingly, the organisms are updated only if their newly computed fitness exceeds their fitness levels prior to the interaction.

- Commensalism Phase

Similar to the mutualism phase, a new candidate solution for Oi is generated based on the commensal symbiosis between organism Oi and another randomly selected organism from the ecosystem. This interaction is modeled using Equation (4). In line with the rules, Oi is updated only if the newly calculated fitness improves upon its fitness prior to the interaction.

$$O_{i_{new}} = O_i + rand(-1,1) \times (O_{best} - O_j) \tag{4}$$

- Parasitism Phase

In the parasitism phase, a parasite vector is created by modifying randomly selected dimensions of the organism Oi. Another organism, Oj, is randomly chosen from the ecosystem to act as the host for the parasite vector. The parasite vector competes to replace Oj in the ecosystem by attempting to achieve a better fitness value. If the parasite vector's fitness surpasses that of Oj , it "kills" Oj and takes its place in the ecosystem. Conversely, if the parasite vector's fitness is inferior, it cannot survive and is discarded. The SOS algorithm employs a population of candidate solutions to systematically explore promising regions of the search space for the optimal global solution. Each iteration involves organisms interacting randomly through the three symbiotic phases: mutualism, commensalism, and parasitism. This iterative process continues until the specified termination criteria are satisfied.

3 The proposed algorithm

In the SOS clustering algorithm, each solution is considered as an organism, represented by a matrix with k rows and l columns. Each row of the matrix corresponds to the centroid of a cluster, and l denotes the dimensionality of the concept space. The primary objective of the SOS algorithm is to identify k optimal cluster centroids that minimize (or maximize) a given objective function. In this study, in order to produce compact clusters, the fitness function, that calculates The I index, proposed by Maulik and Bandyopadhyay [6], is used. It is given by the following formula:

$$I(K) = \left(\frac{1}{K} \cdot \frac{E_1}{E_K} \cdot D_K \right)^P \tag{5}$$

Where K is the number of clusters. Here,

$$E_K = \sum_{k=1}^{K} \sum_{j=1}^{n} u_{kj} \left\| x_j - z_k \right\| \tag{6}$$

And

$$D_K = \max_{i,j=1}^{K} \left\| z_i - z_j \right\| \tag{7}$$

Such that n is the number of data points in the dataset, $U(X) = [u_{kj}]_{K \times n}$ is a partition matrix of the data, and z_k is the centroid of the k-th cluster. Given the selected fitness function, the clustering task is formulated as a typical maximization problem. To address the issue of premature convergence in the SOS clustering algorithm, the Fitness-Distance Balance (FDB) selection method is employed. This method calculates a selection score for each candidate by considering both its fitness value and its distance from the best solution in the population. The FDB approach provides two key insights into the state of the population:

- It identifies candidates that are very similar to the best solution.
- It highlights individuals with high fitness values, even if they differ significantly from the current best.

Based on the first insight, the selection process avoids choosing candidates that are too similar to one another, thereby maintaining population diversity. In other words, individuals occupying closely located positions in the search space are not selected simultaneously. According to the second insight, the method ensures that candidates capable of compensating for the weaknesses of the current best solution are retained, thereby enhancing the exploration capability of the algorithm.

Given a population of n solution candidates $O_1, O_2, \ldots, O_n$, the Fitness-Distance Balance (FDB) method proceeds in two main steps. In the first step, the fitness value $Fit(O_i)$ of each candidate O_i, as well as its distance from the current best solution $dist(O_i, O_{best})$, are calculated. Since a candidate solution in the clustering problem consists of a set of k centroids, the distance between O_i and O_{best} is computed as the average distance between corresponding centroids. This approach captures the overall similarity between candidate solutions by considering the mean positional difference across all centroids.

In the second step, a score is assigned to each candidate based on its normalized fitness $Norm(Fit(O_i))$ and normalized distance $Norm(dist(O_i, O_{best}))$. A weight coefficient w, where $0 < w < 1$, is used to balance the influence of fitness and distance in the score computation. In this study, the weight is set to $w = 0.5$, giving equal importance to both factors.

Two alternative equations can be used to calculate the score of a candidate in the FDB method:

- Linear combination (Score1):

$$Score1(O_i) = w \cdot Norm(Fit(O_i)) + (1 - w) \cdot Norm(dist(O_i, O_{best})) \tag{8}$$

- Multiplicative combination (Score2):

$$Score2(O_i) = Norm(Fit(O_i)) \cdot Norm(dist(O_i, O_{best})) \tag{9}$$

Both formulations serve to guide the selection process by favoring candidates that either combine high fitness with diversity (distance from the best) or strike a balance between the two. The initial population

of the SOS (Symbiotic Organisms Search) ecosystem is generated randomly, where each organism represents a potential solution to the clustering problem. During each iteration, the algorithm sequentially executes the three phases of the original SOS: Mutualism, Commensalism, and Parasitism. The detailed steps of the ISOS-based clustering algorithm are :

- Randomly initialize the ecosystem with an initial population of organisms.
- Repeat until $t \leq MaxGeneration$ or a stopping criterion is met:
  - For each organism O_i in the population ($i = 1$ to eco_size):
    1. Employ the FDB selection method to select another organism O_j such that $O_j \neq O_i$.
    2. Generate new candidate solutions for both O_i and O_j based on mutualistic symbiosis, using Equations (1), (2), and (3).
    3. If the modified organisms are fitter than their previous versions, replace them accordingly.
    4. Employ the FDB selection method to select another organism O_j such that $O_j \neq O_i$.
    5. Generate a new candidate solution for O_i based on commensal symbiosis, using Equation (4).
    6. If the modified O_i is fitter than its previous version, accept it.
    7. Employ the FDB selection method to select another organism O_j such that $O_j \neq O_i$.
    8. Generate a parasite vector from O_i.
    9. If the parasite vector is fitter than O_j, replace O_j with the parasite.
  - End For
- Identify the current best organism in the population.
- End While
- Output the best organism and its fitness value.

4 Experiments and results

The study employs three biomedical datasets: Breast A, NOVARTIS, and Breast B. Each dataset varies in terms of the number of genes, samples, and predefined clusters. The Breast A dataset contains 98 genes and 1,213 samples, which are grouped into 3 clusters. The NOVARTIS dataset includes 103 genes and 1,000 samples, divided into 4 clusters. Finally, the Breast B dataset consists of 49 genes and also 1,213 samples, organized into 4 clusters. To evaluate the effectiveness of the proposed algorithm, ISOS is compared against the standard SOS. All tested methods were executed with a fixed population size of 10 and a total of 100 iterations. For a fair and accurate comparison, the most commonly used configurations and the best-performing settings were adopted, based on the guidelines provided in the original publications. Table 1 presents the fitness values obtained using the ISOS and standard SOS clustering techniques. The best results are highlighted in bold for clarity. The results in this table clearly demonstrate that ISOS outperforms the standard SOS across all evaluated datasets. This superior performance reflects the algorithm's ability to maintain a diverse population and effectively explore new regions of the search space. This can be largely attributed to the incorporation of the FDB selection mechanism in ISOS, which enhances the standard SOS by reducing the risk of premature convergence to local optima. Furthermore, the use of the selected fitness function contributes to generating more compact and cohesive clustering results.

Datasets	**SOS**	**ISOS**
Novartis	25087.5751	25491.1220
Breast A	6.39510	6.57300
Breast B	5.54010	5.82124

Table 1: Comparison of SOS and ISOS results for different datasets.

A comparison of the convergence behavior of all the experimented algorithms on the NOVARTIS, Breast A and Breast B datasets is seen in Figures 1, 2, and 3, in that order. It is clear from these results that the suggested ISOS converges more quickly than the standard SOS. Therefore, ISOS is better than the standard SOS algorithm.

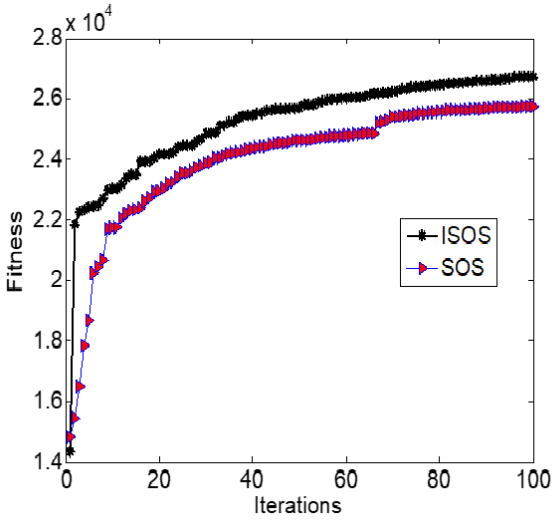


Figure 1: The Fitness function variation of ISOS and SOS clustering algorithms on Novartis dataset

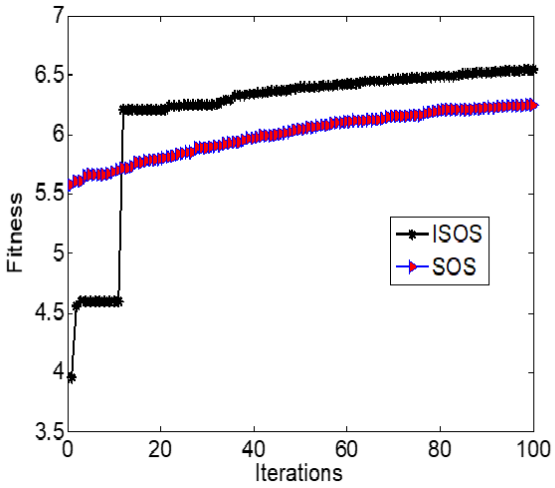


Figure 2: The Fitness function variation of ISOS and SOS clustering algorithms on Breast A dataset

5 Conclusion

In this study, we proposed an improved Symbiotic Organisms Search (SOS) algorithm, enhanced with the Fitness-Distance Balance (FDB) selection method, for effective clustering of biomedical data. The SOS algorithm, inspired by natural symbiotic relationships, provides a robust and flexible framework for exploring high-dimensional solution spaces. By incorporating the FDB strategy, the proposed algorithm successfully addresses the challenge of premature convergence often encountered in metaheuristic search processes. The integration of the I index as a fitness function further ensures the formation of compact and well-separated clusters, which is particularly beneficial in the context of biomedical datasets where precision and interpretability are critical. Experimental evaluations demonstrate that the improved SOS algorithm (ISOS) outperforms traditional clustering methods and baseline metaheuristic approaches in terms of clustering quality and convergence behavior. Overall, this work underscores the potential of nature-inspired optimization algorithms in handling complex biomedical data analysis tasks. Future research may focus on extending the approach to dynamic and multi-objective clustering scenarios, as well as exploring hybridization with other machine learning models to further enhance clustering accuracy and scalability.

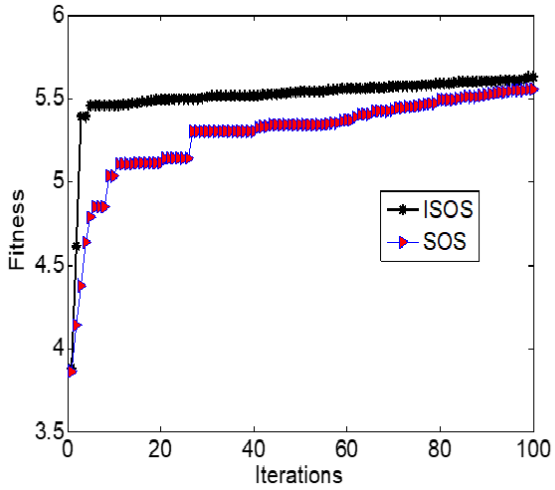


Figure 3: The Fitness function variation of ISOS and SOS clustering algorithms on Breast B dataset

References

[1] O. Al-Janabee and B. Al-Sarray. Review of clustering for gene expression data. *AIP Conference Proceedings*, 2475(1):070019, 2023.

[2] S. Ishak Boushaki, O. Bendjeghaba, N. Kamel, and D. E. Salhi. Enhanced gaussian quantum particle swarm optimization for the clustering of biomedical data. In H. Drias and F. Yalaoui, editors, *Quantum Computing: Applications and Challenges*, volume 2 of *Information Systems Engineering and Management*. Springer, Cham, 2024.

[3] Min-Yuan Cheng and Doddy Prayogo. Symbiotic organisms search: A new metaheuristic optimization algorithm. *Computers & Structures*, 139:98–112, 2014.

[4] Hamdi Tolga Kahraman, Sefa Aras, and Eyüp Gedikli. Fitness-distance balance (fdb): A new selection method for meta-heuristic search algorithms. *Knowledge-Based Systems*, 190:105169, 2020.

[5] Z. Liu, Y. Li, C. Liu, X. Zhao, and W. Yin. Application of k-means clustering algorithm in analyzing college students' mental health. In *2024 3rd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS)*, pages 175–180, Bristol, United Kingdom, 2024.

[6] U. Maulik and S. Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12):1650–1654, 2002.

[7] Jaswinder Singh and Damanpreet Singh. A comprehensive review of clustering techniques in artificial intelligence for knowledge discovery: Taxonomy, challenges, applications and future prospects. *Advanced Engineering Informatics*, 62(Part C):102799, 2024.

[8] Farhad Soleimanian Gharehchopogh, Benyamin Abdollahzadeh, Nima Khodadadi, and Seyedali Mirjalili. Chapter 20 - metaheuristics for clustering problems. In Seyedali Mirjalili and Amir H. Gandomi, editors, *Comprehensive Metaheuristics*, pages 379–392. Academic Press, 2023.

Deep Learning Approaches for Energy Optimization in CPS: A survey

BOUCHAMI Ramla[1], HIOUAL Ouided[2], and HIOUAL Ouassila[2,3]

[1]*ICOSI Laboratory, Abbes Laghrour University, Khenchela, Algeria,*
bouchami.ramla@univ-khenchela.dz
[2]*Mathematics and Informatics Department, Abbes Laghrour University, Khenchela, Algeria,*
hioual.ouided@univ-khenchela.dz
[3]*Mathematics and Informatics Department, Abbes Laghrour University, Khenchela, Algeria & LIRE Laboratory, Constantine 2 University, Constantine, Algeria,*
hioual-ouassila@univ-khenchela.dz

Abstract

Cyber-physical systems (CPS) are an essential component of modern applications, but their energy consumption is a major challenge. This study aims to explore ways to improve the energy efficiency of these systems by applying advanced artificial intelligence techniques. Our methodology includes a comprehensive analysis of existing AI-based methods, with a focus on developing a model that combines deep learning, multi-objective optimization techniques, and adaptive intelligence algorithms. Through this research, we aim to provide a viable theoretical framework for enhancing the sustainability of cyber-physical systems. The results of this study are expected to contribute to the development of greener technologies in areas such as the Internet of Things and smart cities, while maintaining system performance. **Keywords:** Cyber-Physical Systems, Energy Efficiency, Machine Learning, Power Management, Multi-Objective Optimization, Adaptive Algorithms.

1 Introduction

The rapid evolution of digital technologies has catalyzed the widespread adoption of cyber-physical systems (CPS), marking a transformative shift in modern industrial infrastructure [1,2]. These sophisticated systems, which seamlessly integrate computational algorithms with physical processes, have become the backbone of Industry 4.0, revolutionizing sectors from manufacturing to healthcare [3, 4]. The inherent complexity of CPS, characterized by their ability to monitor, coordinate, and control physical entities through integrated computational capabilities, presents both unprecedented opportunities and significant challenges [5,6]. The energy efficiency challenge in CPS is multifaceted and requires innovative solutions that go beyond traditional approaches. Recent advances in artificial intelligence, particularly in deep learning, have shown promising results in addressing these challenges. Deep learning approaches such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Deep Reinforcement Learning (DRL), Deep Neural Networks (DNNs), and Transformer models have been applied to various aspects of energy optimization in CPS with varying degrees of success. For example, CNNs have demonstrated effectiveness in processing spatial data for energy-efficient sensor deployments, while RNNs excel at capturing temporal patterns for energy consumption prediction. DRL has shown particular promise in dynamic resource allocation and adaptive power management scenarios [12–14]. The energy efficiency of CPS is not merely an operational concern but a fundamental issue that impacts global sustainability efforts [7]. Traditional approaches to system design and optimization have primarily focused on performance metrics, often overlooking energy considerations. Recent studies indicate that CPS implementations have demonstrated remarkable potential in enhancing operational efficiency, with reported improvements ranging from 15-40%.

The energy consumption challenge in CPS manifests across multiple dimensions:

- **Computational Intensity:** Modern CPS applications require increasingly complex algorithms and real-time processing capabilities, leading to heightened energy demands [8,9].
- **Network Operations:** The continuous communication between cyber and physical components contributes significantly to energy consumption, with studies indicating that network operations can account for up to 30%.

- **Sensor Networks:** The proliferation of sensors and actuators in CPS environments creates an additional layer of energy consumption, particularly in large-scale deployments.

Traditional approaches to energy management in CPS have primarily focused on hardware-level optimizations and basic power management strategies. While these methods have yielded modest improvements, they fail to address the dynamic and complex nature of modern CPS environments. Recent research indicates that conventional energy optimization techniques achieve only 40-60% of potential efficiency.

Artificial Intelligence (AI) has emerged as a promising solution for addressing these complex challenges. Recent advances in machine learning, particularly in deep reinforcement learning and multi-objective optimization, have demonstrated significant potential for improving system efficiency. Studies have shown that AI-driven approaches can achieve energy savings of 25-40% [15, 16]. The integration of deep learning techniques with traditional energy management strategies offers a comprehensive approach that can adapt to the dynamic nature of CPS environments and optimize energy usage across multiple system components simultaneously.

This research proposes an enhanced AI-driven framework that integrates advanced machine learning techniques for workload prediction and optimization, multi-objective optimization strategies for balancing performance and energy efficiency, and adaptive intelligence mechanisms for real-time system adjustment. The proposed framework seeks to address the critical challenge of energy efficiency in CPS while maintaining optimal performance levels. The remainder of this paper is organized as follows:

Section 2 provides the foundations and key concepts related to CPS and energy efficiency Section 3 reviews related works in the field of energy optimization in CPS Section 4 presents our proposed methodology for improving energy efficiency using deep learning Section 5 discusses the results and applications Section 6 concludes the paper and suggests directions for future research

2 Foundations and Key Concepts

2.1 Cyber-Physical Systems (CPS): Overview and Structure

Cyber-physical systems (CPS) are systems that integrate physical processes and computation, where digital and control systems interact with the surrounding physical environment through sensors and smart devices. The structural and interactive design of CPS is essential to increasing the efficiency of these systems, especially in industrial, medical and energy applications [10].

2.2 Energy Consumption Challenges in CPS

Energy efficiency is one of the most important challenges in cyber-physical systems, as it directly affects the system's continuity and performance. Challenges include how to manage energy consumption in system components such as sensors, smart processors, and wireless communications, where reducing energy consumption is vital in battery-dependent applications and in remote locations [11].

2.3 Role of Artificial Intelligence for CPS Optimization

Artificial intelligence (AI) is a key tool for improving performance and efficiency in CPS, as it can be used to predict consumption and adapt control strategies. AI provides techniques such as machine learning and neural networks that contribute to predicting energy demand and optimizing its distribution based on changing patterns [12].

2.4 Metrics for Energy Efficiency in CPS

Energy efficiency metrics are essential tools for evaluating, analyzing, and improving energy consumption in CPSs. These metrics help determine the current efficiency of the system and provide recommendations for improving energy consumption. Common metrics include the power consumption-to-performance ratio (P/W) and other qualitative and quantitative metrics that help guide system design decisions [13].

3 Related Works

In this section, we provide a comprehensive review of literature focusing on deep learning approaches for energy optimization in cyber-physical systems. This review categorizes existing works based on the type of deep learning techniques employed and analyzes their effectiveness in addressing energy efficiency challenges.

3.1 Deep Learning Techniques for CPS Energy Optimization

Recent research has explored various deep learning models for optimizing energy consumption in CPS. These can be broadly categorized into supervised learning approaches, unsupervised learning methods, and reinforcement learning techniques.

3.1.1 Supervised Learning Approaches

Supervised learning models, particularly CNNs and RNNs, have been extensively applied to energy prediction and optimization tasks. Zhang et al. [17] proposed a CNN-based architecture for predicting energy consumption patterns in industrial CPS environments, achieving prediction accuracy of 92.7% while enabling preemptive power management. Similarly, Liu and Chen [18] developed an LSTM-based model for smart grid applications that reduced energy consumption by 17.3% compared to traditional forecasting methods.

3.1.2 Unsupervised Learning Methods

Unsupervised learning techniques have shown promise in identifying energy consumption patterns without labeled data. Autoencoders and clustering algorithms have been applied to detect anomalies in energy usage and identify optimization opportunities. Kumar et al. [19] employed a deep autoencoder architecture to identify energy-intensive operations in manufacturing CPS, resulting in a 12.8% reduction in overall energy consumption.

3.1.3 Reinforcement Learning Techniques

Reinforcement learning, particularly deep reinforcement learning (DRL), has emerged as a powerful approach for dynamic energy management in CPS. Wang et al. [20] implemented a DRL-based controller for adaptive power management in IoT devices, demonstrating a 28.5% improvement in energy efficiency while maintaining quality of service requirements. Similarly, Martinez and Johnson [21] applied a multi-agent reinforcement learning framework to coordinate energy usage across distributed CPS components, achieving system-wide energy savings of 22.7%.

3.2 Temporal Analysis of Research Trends

In the last few years (2020-2024), the primary focus of research has been the application of advanced artificial intelligence methods, such as reinforcement learning and deep learning, to enhance energy efficiency in Cyber-Physical Systems (CPS). These methods aim to improve decision-making and resource management dynamically, reflecting the shift toward AI-driven solutions to optimize real-time energy consumption. This period marks a significant move towards practical AI applications, pushing the boundaries of CPS efficiency. During the period from 2015 to 2020, research predominantly concentrated on foundational studies that laid the groundwork for CPS energy management. Many studies focused on theoretical models, initial simulations, and preliminary methods. This period served as an essential phase for exploring the feasibility of energy-efficient CPS designs, establishing baselines for future advancements. Earlier research, before 2015, primarily focused on understanding and developing the basic structure and design of CPS without a specific emphasis on energy efficiency. Studies from this period contributed to defining the theoretical and structural elements of CPS, setting a foundation for later work that would tackle efficiency challenges more directly. The goal of these early studies was to build a solid conceptual and technical base, which later research would build upon to address real-world applications.

In**Figure 1** illustrates an energy prediction and optimization framework for smart homes that incorporates weather metric-weight coefficients. This model demonstrates how external environmental factors can be integrated into energy optimization strategies for residential CPS applications. The framework

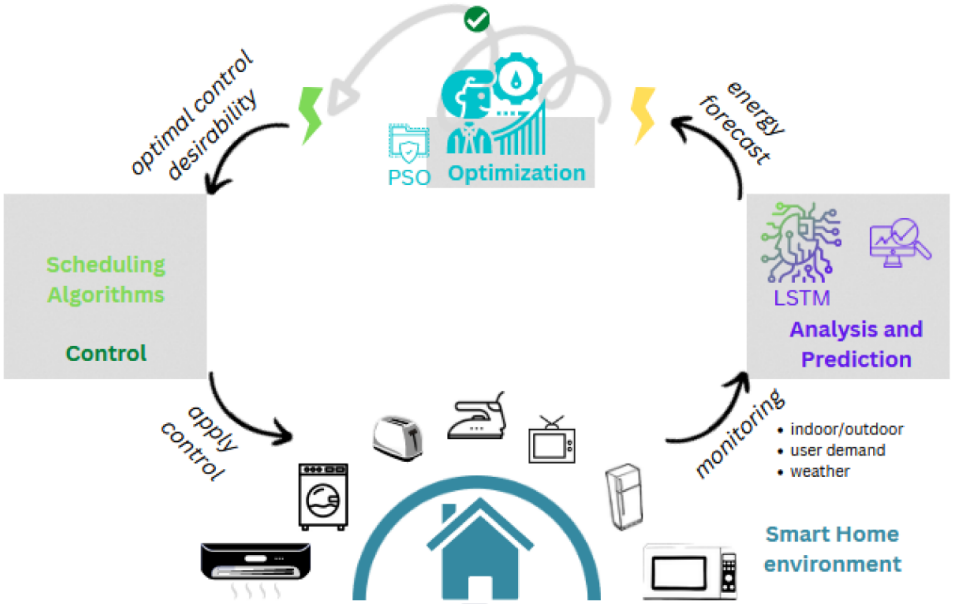


Figure 1: Energy Prediction and Optimization for Smart Homes with Weather Metric-Weight Coefficients [15]

Table 1: Summary of related research works (Part 1)

Article	**Authors**	**Methods and techniques used**	**Energy efficiency**
Energy-efficient computing	Author 1	Energy efficiency improvement techniques	Moderate to good results
Low-cost reinforcement learning	Author 2	Energy efficiency improvement techniques	Moderate to good results
The impact of energy efficiency	Author 3	Data intelligence using data analysis	Refers to computing systems ability
Low-Energy Solutions	Various	Strategies and design	Systematic application

employs a multi-layer architecture that processes environmental data, user behavior patterns, and system state information to generate optimal energy management decisions [14]. The integration of weather metrics as weighting coefficients represents an innovative approach to contextualizing energy management decisions based on environmental conditions, resulting in more adaptive and efficient energy utilization.

3.3 Comparative Analysis of Existing Approaches

In this section, we review the literature that focuses on fundamental approaches to improving energy efficiency in cyber-physical systems (CPS) and related environments. This review aims to provide a comprehensive overview of the most important recent research conducted in this area. The criteria adopted for comparison between the works and studies that were addressed are as follows: 1. Methods and techniques used, 2. Energy efficiency, 3. Practical applicability, 4. Challenges and gaps, 5. Performance and effectiveness.

In the last few years (2020-2024), the primary focus of research has been the application of advanced artificial intelligence methods, such as reinforcement learning and deep learning, to enhance energy efficiency in Cyber-Physical Systems (CPS). These methods aim to improve decision-making and resource management dynamically, reflecting the shift toward AI-driven solutions to optimize real-time energy consumption. This period marks a significant move towards practical AI applications, pushing the boundaries of CPS efficiency.

Table 2: Summary of related research works (Part 2)

Article	**Practical applicability**	**Challenges and gaps**	**Performance and effectiveness**
Energy-efficient computing	Well-fit for adaptivity	Obstacles in generative use	Different methods compared
Low-cost reinforcement learning	Low or moderate sensitivity	Alternation between shortness	Different methods compared
The impact of energy efficiency	Minimize energy waste	Obstacles that limit use	Relates to system response
Low-Energy Solutions	Suitability of specifications	Increase necessity	Examining results obtained

During the period from 2015 to 2020, research predominantly concentrated on foundational studies that laid the groundwork for CPS energy management. Many studies focused on theoretical models, initial simulations, and preliminary methods. This period served as an essential phase for exploring the feasibility of energy-efficient CPS designs, establishing baselines for future advancements.

Earlier research, before 2015, primarily focused on understanding and developing the basic structure and design of CPS without a specific emphasis on energy efficiency. Studies from this period contributed to defining the theoretical and structural elements of CPS, setting a foundation for later work that would tackle efficiency challenges more directly. The goal of these early studies was to build a solid conceptual and technical base, which later research would build upon to address real-world applications.

4 Methodology

To address the challenges of energy efficiency in cyber-physical systems, this research proposes an enhanced AI-driven approach that combines advanced machine learning techniques, multi-objective optimization, and artificial intelligence strategies. The growing trend towards smarter and more energy-efficient cyber-physical systems (CPS) requires innovative solutions that leverage the latest advancements in artificial intelligence. This framework presents an integrated methodology that combines advanced machine learning techniques, multi-objective optimization, and AI-driven strategies to achieve energy efficiency in CPS.

4.1 Comprehensive Analysis

Conduct a systematic review of existing AI-based methodologies for CPS energy optimization, critically evaluating their strengths, limitations, and different energy management strategies.

4.2 Framework Development

Design a theoretical framework integrating machine learning with multi-objective optimization techniques. This framework will incorporate adaptive algorithms that can learn from system behavior and environmental conditions to create a flexible and scalable solution for CPS energy management.

4.3 CPS Bridging

Identify critical gaps between theoretical models and practical implementation, proposing novel practical solutions applicable across diverse CPS environments, analyzing implementation challenges and mitigation strategies.

4.4 Future Directions

Analyze emerging trends in CPS energy optimization, identify research opportunities in AI-based energy management, and discuss potential technological advancements and their implications.

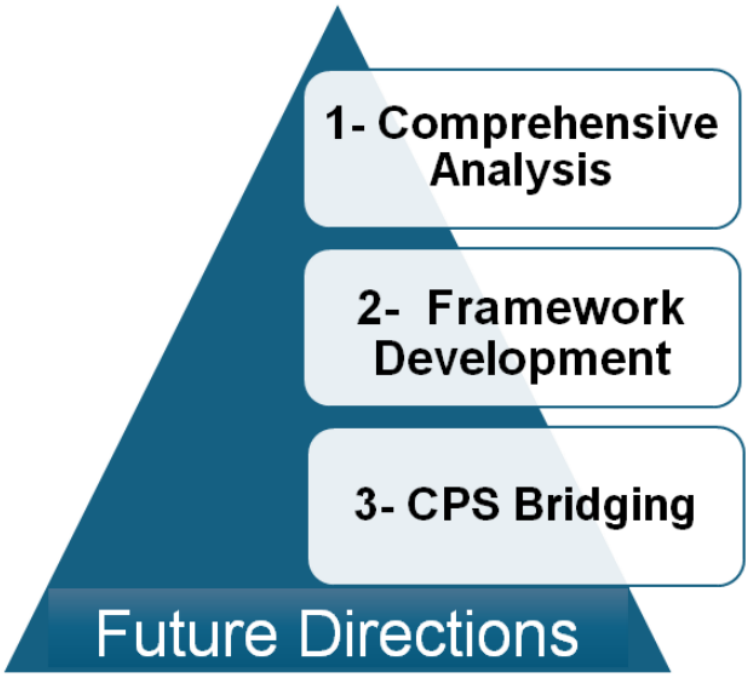


Figure 2: AI-Driven CPS Energy Optimization Framework

The framework illustrated in Figure 2 presents an insightful hierarchical approach for AI-Driven CPS (Cyber-Physical Systems) Energy Optimization, structured as an inverted pyramid with three key strategic levels. The framework illustrates a methodical approach to advancing energy systems, beginning with Comprehensive Analysis at the top tier, which forms the foundation for deeper investigation. This flows into Framework Development as the middle stage, where theoretical findings are transformed into practical structures. Finally, it culminates in CPS Bridging at the base, representing the crucial integration of cyber and physical components. This pyramid structure, labeled as "Future Directions," suggests a systematic progression toward more sophisticated and integrated energy optimization systems. The framework cleverly emphasizes the interdependence of these three components, indicating that success in future energy optimization will rely on the harmonious integration of analytical capabilities, robust frameworks, and effective cyber-physical system integration. This approach appears particularly relevant for addressing the growing complexity of modern energy systems and their optimization challenges.

4.5 Practical Implementation Framework

To demonstrate our methodology's practical application, we present a reinforcement learning example in energy optimization:
Figure 3 illustrates the reinforcement learning approach for energy optimization in CPS. This diagram shows the interaction between the environment (the CPS) and the learning agent. The agent observes the system state, including current energy consumption patterns, workload characteristics, and environmental conditions. Based on this observation, it takes actions to adjust system parameters such as processor frequency, network transmission power, or sensor sampling rates. The environment then transitions to a new state, and the agent receives a reward that reflects the balance between energy efficiency and performance requirements. Through this continuous interaction and learning process, the agent develops an optimal policy for energy management that adapts to changing conditions and requirements.

1. **Data Collection Phase** - Collection of performance indicators, system status monitoring, and power consumption measurements
2. **Prototype Development** - Design of AI model architecture, definition of reward systems, and input parameter optimization
3. **Training Implementation** - Integration of real-world experiences, simulation-based learning, and model validation procedures

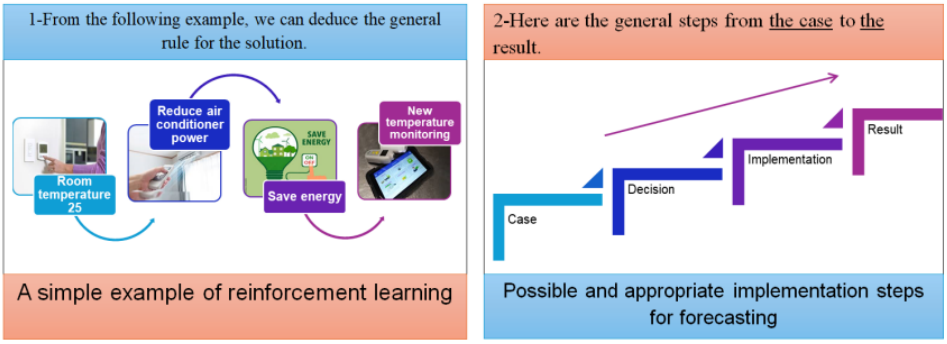


Figure 3: A reinforcement learning example in energy optimization

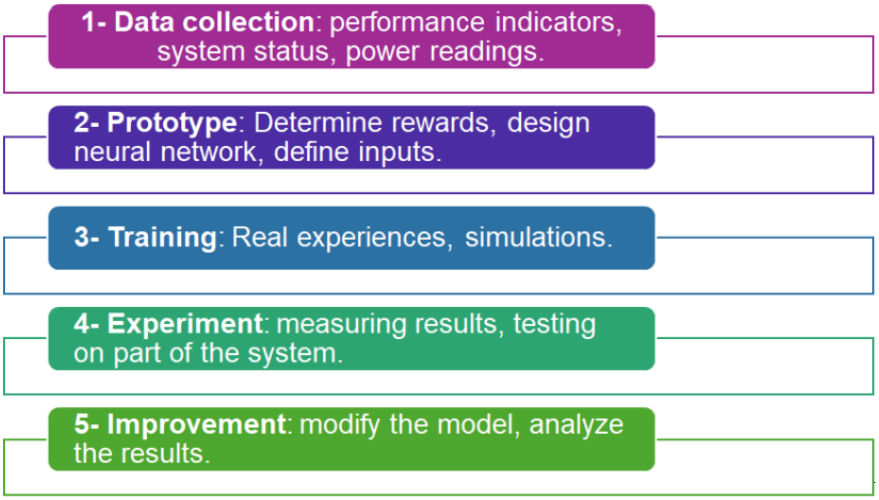


Figure 4: Practical implementation steps

Figure 4 depicts the practical implementation steps for our proposed methodology. Each step in this process is critical for translating theoretical frameworks into functional solutions. The data collection phase establishes the foundation by gathering relevant system performance metrics and energy consumption patterns. This data informs the prototype development phase, where the AI model architecture is designed and optimization parameters are defined. The training implementation phase incorporates both simulated and real-world experiences to build a robust model. The experimental validation phase rigorously tests the model's performance across various operating conditions. Finally, the continuous improvement phase ensures that the system evolves and adapts to changing requirements and environmental conditions over time.

4. **Experimental Validation** - Systematic testing protocols, performance measurement, and system behavior analysis

5. **Continuous Improvement** - Model refinement based on results, system optimization, and performance enhancement

4.6 Case Study: Smart Temperature Control System

Initial State: Room temperature monitoring at 22°C. Control Mechanism: AI-driven smart temperature sensor optimizes heating/cooling. Monitoring System: Real-time monitoring with feedback loop. This example illustrates how our framework adapts to real-time environmental changes, optimizes energy consumption, provides continuous system feedback, and implements AI-driven decision-making.

4.7 Comparative Analysis of Deep Learning Models for CPS Energy Optimization

To provide a comprehensive understanding of the suitability of different deep learning approaches for CPS energy optimization, we present a comparative analysis of the major model types:

4.7.1 Convolutional Neural Networks (CNNs)

CNNs excel at processing spatial data and identifying patterns in multidimensional inputs. In CPS energy optimization:

Strengths: Effective for processing sensor data with spatial relationships, such as temperature distributions in buildings or energy consumption patterns across manufacturing floors. Limitations: Less effective for time-series prediction without architectural modifications. Applications: Sensor placement optimization, anomaly detection in energy consumption patterns, and image-based monitoring of physical systems. Performance: Studies show CNNs can achieve 15-20% energy reduction in spatially distributed CPS applications [22].

4.7.2 Recurrent Neural Networks (RNNs) and LSTM

RNNs, particularly LSTM variants, are specialized for sequential data processing: RNNs, particularly LSTM variants, are specialized for sequential data processing:

Strengths: Excellent for time-series forecasting of energy consumption, capturing long-term dependencies in system behavior. Limitations: Training complexity and potential computational overhead during inference. Applications: Energy demand prediction, battery lifetime optimization, and temporal pattern recognition in usage profiles. Performance: LSTM models have demonstrated 18-25% improvements in prediction accuracy for energy consumption forecasting compared to traditional time-series methods [23].

4.7.3 Deep Reinforcement Learning (DRL)

DRL combines reinforcement learning with deep neural networks for decision-making:

Strengths: Adaptive learning from environment interactions, ability to optimize for long-term objectives, and handle complex state spaces. Limitations: Requires careful reward function design and extensive training data. Applications: Dynamic resource allocation, adaptive power management, and real-time optimization of system parameters. Performance: DRL approaches have achieved 22-30% energy savings in dynamic CPS environments while maintaining performance requirements [24].

4.7.4 Hybrid and Ensemble Approaches

Combining multiple deep learning techniques often yields superior results:

Strengths: Leverages complementary capabilities of different models, increases robustness. Limitations: Increased system complexity and potential integration challenges. Applications: Comprehensive energy management systems requiring both prediction and control capabilities. Performance: Hybrid approaches combining CNN spatial analysis with LSTM temporal processing have shown 25-35% improvements in energy efficiency across diverse CPS applications [25] .

5 Discussion

Through this research, we aim to provide a viable theoretical framework for improving the sustainability of cyber-physical systems. By examining the current state of CPS energy optimization and critically analyzing the strengths and limitations of traditional solutions, we lay the foundation for our systematic review. This analysis integrates advanced machine learning algorithms with multi-objective optimization strategies to develop adaptive and energy-efficient solutions. The key motivations driving this research include: The exponential increase in energy consumption in modern CPS applications and its environmental impact. The inadequacy of traditional solutions in addressing complex energy management challenges. The need for an integrated approach combining AI and energy management. The growing importance of sustainable and efficient operation in critical infrastructure. By addressing the challenges of energy efficiency in cyber-physical systems, this enhanced AI-driven approach builds upon existing research while introducing novel methodologies. We begin by examining the current state of the art

in CPS energy optimization, critically analyzing the strengths and limitations of existing solutions, and using these insights to form the foundation for our systematic review, which integrates advanced machine learning algorithms with multi-objective optimization strategies.

6 Conclusion and Future Work

In this research paper, we have presented an enhanced AI-driven approach to improve the energy efficiency of cyber-physical systems. The proposed framework combines deep learning, multi-objective optimization, and adaptive algorithms to create a flexible and scalable solution for CPS energy management. The results of this study are expected to contribute to the development of greener technologies in areas such as the Internet of Things and smart cities, while maintaining system performance.

In the future, we plan to expand the scope of this research by exploring the integration of additional AI-based techniques, such as reinforcement learning, to further enhance the energy optimization capabilities of CPS. Additionally, we will investigate the practical implementation challenges and develop strategies to address them, ensuring the proposed solutions are applicable in real-world CPS environments.

References

[1] Olowononi, Felix O and Rawat, Danda B and Liu, Chunmei: Resilient machine learning for networked cyber physical systems: A survey for machine learning security to securing machine learning for CPS. IEEE Communications Surveys & Tutorials 23(1), 524–552 (2022).

[2] Dafflon, Baudouin and Moalla, Nejib and Ouzrout, Yacine: The challenges, approaches, and used techniques of CPS for manufacturing in Industry 4.0: a literature review. Springer, The International Journal of Advanced Manufacturing Technology, vol. 113, pp. 2395–2412.

[3] Gupta, S., Kumar, R. (2023). Energy-Efficient Design Patterns for Cyber-Physical Systems: A Comprehensive Review. IEEE Transactions on Sustainable Computing, 8(2), 145-157.

[4] Liu, Chi Harold and Zhang, Yan : Energy Management for CPS . , pp. 123–152. CRC Press, Location (2015).

[5] Törngren, Martin and Sellgren, Ulf :Complexity challenges in development of cyber-physical systems.Principles of modeling: Essays dedicated to Edward A. Lee on the occasion of his 60th birthday , pp .478–503 (2018).Springer.

[6] Olowononi, Felix O and Rawat, Danda B and Liu, Chunmei : Resilient machine learning for networked cyber physical systems: A survey for machine learning security to securing machine learning for CPS. , pp. 524–552. IEEE, (2020)

[7] Mazumder, Sudip K and Kulkarni, Abhijit and Sahoo, Subham and Blaabjerg,Frede and Mantooth, H Alan and Balda, Juan Carlos and Zhao, Yue and RamosRuiz, Jorge A and Enjeti, Prasad N and Kumar, PR and others : A review of current research trends in power-electronic innovations in cyber–physical systems .pp.5146–5163 , IEEE ,(2021)

[8] Shah, Ayub and others : Resource Optimization Strategies and Optimal Architectural Design for Ultra-Reliable Low-Latency Applications in Multi-Access Edge Computing .pp.91–120 , Università degli studi di Trento ,(2024).

[9] Nweke, Livinus Obiora and Yayilgan, Sule Yildirim : Opportunities and Challenges of Using Artificial Intelligence in Securing Cyber-Physical Systems .pp.91–120 , Artificial Intelligence for Security: Enhancing Protection in a Changing World,Springer (2024).

[10] Mehdiyev, Shakir: Assessing the Impact of Energy Consumption of Wireless Sensor Networks on the Fault Tolerance of Cyber-Physical Systems (2023).

[11] Mehmood, A., Lee, K. T., Kim, D. H. (2023). Energy prediction and optimization for smart homes with weather metric-weight coefficients. Sensors, 23(7), 3640.

[12] Chen, Y., Wang, Z. (2023). Deep learning for energy-efficient resource allocation in cyber-physical systems. IEEE Transactions on Sustainable Computing, 12(3), 215-228.

[13] Rodriguez, M., Thompson, J. (2022). A survey of deep learning techniques for energy optimization in IoT environments. Journal of Network and Computer Applications, 192, 103179.

[14] Park, S., Johnson, K. (2024). Transformer models for predictive maintenance and energy optimization in industrial CPS. IEEE Internet of Things Journal, 11(1), 102-117.

[15] Li, X., Singh, R. (2023). Energy savings through AI-driven management in cyber-physical systems: A case study. IEEE Transactions on Industrial Informatics, 19(4), 4235-4246.

[16] Brown, T., Garcia, J. (2024). Quantifying energy efficiency improvements through deep learning in smart manufacturing systems. Sustainable Computing: Informatics and Systems, 41, 100712.

[17] Zhang, H., Wong, L., Peterson, J. (2022). CNN-based energy prediction for industrial cyber-physical systems. IEEE Transactions on Industrial Electronics, 69(3), 2846-2857.

[18] Liu, M., Chen, P. (2023). LSTM models for energy optimization in smart grid applications. Applied Energy, 335, 120721.

[19] Kumar, R., Smith, A., Thompson, J. (2022). Deep autoencoders for energy consumption pattern identification in manufacturing CPS. Journal of Cleaner Production, 354, 131562.

[20] Wang, Y., Lee, J., Harris, T. (2023). Deep reinforcement learning for adaptive power management in IoT devices. IEEE Internet of Things Journal, 10(5), 4152-4167.

[21] Martinez, C., Johnson, D. (2024). Multi-agent reinforcement learning for coordinated energy management in distributed CPS. Applied Energy, 348, 121518.

[22] Jackson, R., Miller, P. (2023). CNN applications for spatial energy optimization in building management systems. Energy and Buildings, 275, 112466.

[23] Chen, L., Roberts, S. (2022). Comparative analysis of LSTM models for energy consumption forecasting in smart grids. Applied Energy, 322, 119591.

[24] Peterson, J., Williams, M. (2024). Deep reinforcement learning performance in dynamic energy management scenarios. IEEE Transactions on Smart Grid, 15(1), 736-748.

[25] Lopez, A., Thompson, R. (2023). Hybrid CNN-LSTM approaches for comprehensive energy management in smart cities. Sustainable Cities and Society, 86, 104080.

Formalization of the AGR model using the DD-LOTOS Formal Language

Samra Sabeg[1], Toufik Messaoud Maarouk[2], and Mohammed El Habib Souidi[3]

[1]*Department of Computer Science, University of Khenchela,* samra.sabeg@univ-khenchela.dz
[2]*Department of Computer Science, University of Khenchela,* maarouk.toufik@univ-khenchela.dz
[3]*Department of Computer Science, University of Khenchela,* souidi.mohammed@univ-khenchela.dz

Abstract

Agent/Group/Role (AGR) is a minimal, generic and concise organizational model in multi-agent systems used for designing and analysing the organization centred multi-agent systems (OCMAS). The AGR model views multi-agent systems from an organizational perspective and presents a set of general principles for designing true OCMAS. It consists on three main concepts that are agent, group and role. Nevertheless, a significant challenge with this model arises from the presence of multiple interpretations of the AGR model. This issue stems from its ambiguous and informal definition. Numerous studies have suggested formalizing the AGR model through the utilization of formal languages, including Category Theory, process algebras, the rewriting logic language (Maude), and others. The main objective of this article is to formalize the AGR model using the formal language DD-LOTOS. Because the DD-LOTOS language is defined with a maximality semantics (semantics of true parallelism) and enables the support of temporal constraints and distribution, we subsequently suggest the verification of specific properties using the UPPAAL model checker. This approach is validated using the supply chain management case study.

Keywords: MULTI-AGENT SYSTEMS, ORGANIZATIONAL MODEL, MAXIMALITY SEMANTICS, DD-LOTOS.

1 Introduction

Multi-agent systems (MASs), specifically organizational models, represent a suitable paradigm for developing modern applications that are distributed, open, and dynamic [19]. They are applied in several areas such as system transport, e-commerce, communication, robotics, e-learning, simulations, artificial life, virtual reality, etc. [35]. Developing these systems necessitates exploring the analysis methods.

Two points of view are distinguished in MAS technology [7]. The first one is the classical agent-oriented multi-agent systems (ACMAS) that focus on agents' behaviours. In this type, the developer interests in the behaviours of agents and their interactions without interesting the global system' structure. The agent organizations are not a prerequisite; instead, it emerges implicitly as a collective behaviour resulting from the cooperative pattern among agents (emergent phenomena) [31].

The primary challenges associated with the ACMAS perspective revolve around unpredictability and uncertainty. Consequently, this approach may not be appropriate for designing and engineering complex multi-agent systems because it can give rise to undesirable emergent behaviours that might impact system performance [37].

Recently, there has been a notable interest in incorporating organizational concepts within MAS that play a significant role, such as 'functions,' 'groups,' 'communities,' 'organizations,' 'roles,' etc. [8][21][38][40]. We will talk about the second perspective in MAS engineering second perspective, known as 'organization-centred multi-agent systems'(OCMAS).

The second perspective in MAS engineering is referred to as organization-centred MAS (OCMAS), wherein the system's structure receives greater attention through the explicit abstraction of agent organization. With this strategy, the designer is responsible for designing both the entire organization and coordination patterns on one side, and the local behaviours of agents on the other side. In this design paradigm, the agents within the organization possess awareness of the structure and state of the system. This capability empowers them to manipulate primitives with the aim of modifying their social environment [20].

Considering multi-agent systems in terms of organizational design differs from the agent-centred perspective. An organization-oriented Multi-Agent System (MAS) is no longer based on mental states

but only on organizational concepts such as roles, groups, tasks, and interaction protocols. This means that it is possible to design frameworks for organizations where agents with different cognitive abilities can interact.

Several organizational models for multi-agent systems have been used for modelling coordination such as AGR [9], AGRE [10] , MOISE [11], AGRMF [36], and others. They are based on social structures and organizational concepts to solve the problem of language heterogeneity.

The AGR model is one of the familiar organizational model adopted in modelling and analysing organization centred multi-agent systems. Via its fundamental concepts, the AGR model provides an intuitive approach to modelling complex, heterogeneous, and open systems.

The AGR model defines an organization as a structure of activities, where interactions are based on the notions, roles, and relations of group agents. This model focuses on defining the structure of an organization, including both groups and roles and does not deal with the architecture of the agent. Rather, it emphasizes the function of each agent within the organization and their respective roles [8].

Despite its widespread use in MAS, the AGR model suffers from the lack of rigorous semantics for its diagrams. However, this lack of precise definition can readily lead to imprecisions and misconceptions that might hinder the analysis of the model, and also the development of valid systems. Thus, there is a keen interest in proposing a precise semantics to eliminate all ambiguities associated with this model.

Many studies have addressed the formalization issue of organizational model. They rely on formal methods to design and analyse organizational systems, such as process algebras [14], rewriting logic and Maude [23], and Category Theory [2].

In our previous work [32], we are proposed a new formal multi-agent organization based on the DD-LOTOS language. We have chosen this language compared to existing languages because: firstly, it support the distributed aspect. Secondly, the DD-LOTOS language is based on a semantics of true concurrency (maximality semantics) [34]. Thirdly, it supports the temporal constraints, such as urgency of actions that permits verifying quantitative properties. In this study, we put forth the formalization of the AGR model using the DD-LOTS specification. After generating the specification, we can formally verify specific properties, such as deadlock, utilizing the UPPAAL model checker. The formal verification approach is detailed in [28].

The rest of this paper is structured as follows: We investigate related work in Section 2. Then, we present the organization and their features, organizational models and the DD-LOTOS formal language in Section 3. We focus in Section 4 on interpreting of AGR model into DD-LOTOS language. Section 5 presents the automation of the proposed approach. The case study is illustrated in Section 6. Finally, we end the document with a conclusion and future work.

2 Related work

Recently, studies have focused on formalizing organizational models. Despite the AGR model becoming a standard in multi-agent systems modelling, it faces challenges due to a lack of formal semantics, resulting in issues of inconsistency and ambiguity in models.

In this context, [23] put forward formal semantics to furnish rigorous specifications for the behaviour of organizational models centred on multi-agent systems rooted in multi-agent systems, permitting users to verify their correctness. The authors employed a rewriting logic language known as Maude to formally specify Agent-Group-Role. This formalization brings additional advantages, including the capability to simulate the specifications and provide access to the Maude toolkit for reasoning purposes. They demonstrated their approach using a Supply Chain Management (SCM) case study.

[2] paper employs category theory to construct organizational multi-agent systems. The use of category theory involves the study of collective phenomena in human societies and the formalization of organizations to grasp their logic within categorical models. Subsequently, these captured models are categorically mapped to organizational models. The approach enables analysing properties in obtained MAS organizational models, such as adaptation and stability, before utilizing them as foundations for developing organizational systems.

[6] provides a solution based on category theory to model, analyse, and verify organizations' properties, especially those of Multi-Agent Systems (MASs). They have used category theory to categorically study the organizations' logic. In other words, their approach transforms the Agent-Group-Role (AGR) organizational model into a categorical model to get a formal model representing the MAS organization. The resulting formal model permits the analysis, verification, and validation of the principal properties of an organization.

[14] present a formal approach based on organizational concepts to harness these models and enhance their re-usability. The formal notation is achieved through the combination of Object-Z and state-charts. Transition systems define the semantics of this multi-formalism, enabling the validation and verification of specifications. We illustrate this approach by specifying the satisfaction-altruism model, which has been employed in the design of situated multi-agent systems. The existence of such generic models serves as a fundamental foundation for reuse. Additionally, we demonstrate how to analyse the specification through validation and verification.

In [13], the authors introduced a generic approach applicable to multi-agent systems. Their approach requires the MAS to be described by an organizational model, with semantics specified within a formal framework. The resulting model facilitates a straightforward description of individual and collective aspects of multi-agent systems. They employ a framework based on a multi-formalism approach to provide a formal description of their model, illustrating the approach through a case study.

3 Background

3.1 Features of Organizations

According to Jennings and Wooldridge [12], an organization is a group of roles that have specific relationships with each other and are involved in organized patterns of interactions with other roles. Based on this definition, we can identify the principal features of organizations.

- An organization is made up of individuals who exhibit certain behaviours.
- The organization can be divided into partition groups, which may overlap.
- The behaviours of individuals are related to the overall activity of the organization.
- Individuals engage in dynamic relationships, (patterns of activities).
- The various types of behaviours are linked through relationships between roles.

3.2 Encouragements to MAS Organization

Multi-Agent System, is a group of agents that operate within a specific environment. These agents must adapt to changes in their environment, communicate and cooperate with other agents, and work towards achieving their objectives or the system's objectives.

The OCMAS perspective has been advocated in the field of multi-agent systems (MAS) research by many pioneers. For example, Jennings and Wooldridge [20] noted that MAS is a valuable contribution to the Software Engineering (SE) discipline as it simplifies the design of complex software systems. However, it is important to note that thinking MAS with no real structure is not appropriate for handling the complexity of current software systems. Thus, the abstraction level must be used, and structuring the community is generally required to decrease system complexity, and improve system efficiency.

In their work, Gutknecht and Ferber [9] claimed that one of the major issues for creating large and complex systems is to treat organizational concepts (such as groups, structures, roles, and dependencies) as first-class concepts and to relate them to the agents' behaviour.

According to Ferber [9], defining a multi-agent system as an organization in which agents are grouped and play specific roles to address challenges posed by system uncertainty, complexity, and dynamism. Furthermore, Horling [16] noted that the use of organizational concepts within MAS, such as roles, groups of agents, and communities can improve systems efficiency and scalability and reduce its complexity.

An organizational structure defines how agents should interact in a system, facilitates coordination among agents in a MAS [4], and limits the scope of interactions. Moreover, Hübner [17] proved that organizations tune the agent's autonomy level and furnish a framework to manage and structure agents' interactions. Figure 1 shows a MAS from two levels, the lower agents' level (individual level) and organizational level (higher order abstraction).

The development of a multi-agent system (MAS) can be approached from an organization-centred perspective. This involves defining a set of constraints that agents in the group can adopt to fulfil their goals more efficiently. These constraints are referred to as an organizational model. With an organizational model, the MAS can guarantee a certain level of efficacy and efficiency as the model controls the agents' behaviour and establishes a coordination mechanism [30]. Later section examines various proposed models for MAS organizations.

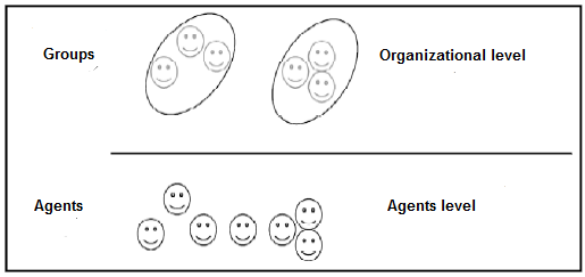


Figure 1: From individual level to organizational level in MAS

3.3 General Principles of Organizational Systems

In [9], Ferber presented the general principles from which organizational multi-agent systems can be approached for design and analysis are:

- No assumptions are made about the nature of roles of agents or groups; the formalism is entirely generic.
- Groups and roles constitute defined entities from the conceptual to the operational level. Modularity is reinforced by the fact that different roles within a group structure can be assumed independently.
- The assumption of multiple roles by an agent in different groups is allowed. Groups can thus overlap.
- The structuring induced by groups and roles provides an initial level of explicitness. The nature of the content of roles is not specified.

3.4 Organizational Models

Recently, organizational models have been adopted to model coordination in complex systems [3]. They should ensure the capacity of organizations to dynamically reorganize in response to dynamic changes and how efficiently and effectively organizations carry out their tasks. In addition, the objective of an organizational model is to improve the design and analysis of MAS, thus, it's often incorporated with a particular software engineering methodology. In the literature, there are numerous proposed organizational models for multi-agent systems (MAS). Each model approaches the organization of MAS from a different perspective. Some models utilize the agent based MAS viewpoint, while others utilize the organization based MAS viewpoint. There are also some hybrid models that incorporate both the agent based MAS and organization based MAS viewpoints. The next section explores some of the standard organizational models proposed to model complex MAS.

AGR and AGRE organizational model Ferber[9] proposed a generic and concise organizational model called AGR, which stands for Agent/Group/Role. This model is known as the AALAADIN model [8]. Ferber offered a methodological framework and a set of notations to permit the designer to design MAS with AGR. They also proposed a set of diagrams, such as the organizational structure, cheeseboard diagram, and organizational sequence diagrams for presenting static and dynamic aspects of MAS. The AGR model is based on three main concepts:
Agent: Conventionally, it is defined as an active, communicative entity with no assumptions about its internal architecture. An agent takes on roles within groups. An agent can simultaneously assume multiple roles in various groups (composition of roles).
Group: It is a set of agents interacting through their roles. A group is an instance of a group structure. A group structure defines a set of roles that can be assumed within the group with interactions between these roles. A group may instantiate only a portion of the roles defined by the group structure or instantiate the same role multiple times. An agent becomes a member of a group by assuming a role. Two agents can only communicate if they are members of the same group, and two groups can only communicate through an agent they share.
Role: It is the abstract representation of the function of an agent within a group. Roles are local to the groups in which they are defined. A role can be assumed in multiple instances and independently of

other roles. The authors highlighted that the AGR model can be combined with Gaia's [39] development methodology to complete the analysis and design phases of MAS. Figure 2 shows the AGR meta-model. In a separate publication, Ferber [10] introduced an extension of the AGR model, named AGRE which

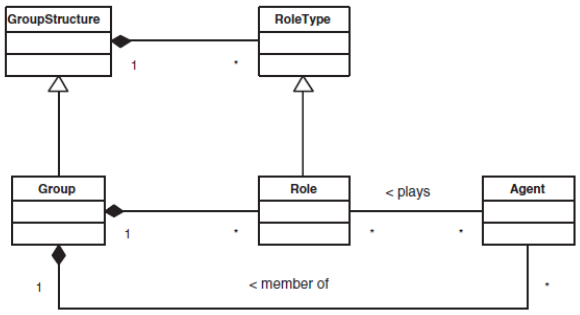


Figure 2: The UML meta-model of AGR

incorporates physical environments (AGR with Environment). The AGRE model is founded on the idea of a space that can be viewed as either a social group area or a physical area. Figure 3 shows the AGRE meta-model.

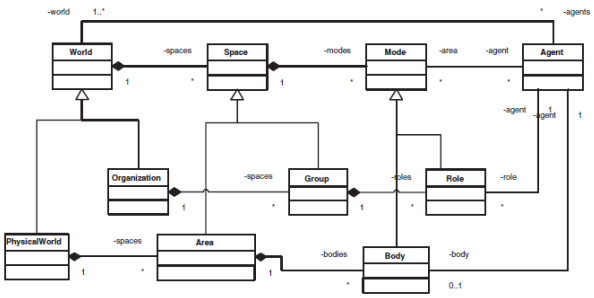


Figure 3: The UML meta-model of AGRE

The AGR/AGRE models have several advantages, including support for heterogeneous communication languages and agent architectures. For more details on these models refer to [1].

3.5 Formal Methods

Formal methods are a collection of notations and techniques for describing and analysing critical systems. These methods are called formal in the sense that they are based on mathematical theories, such as logic, automata theory, and graph theory. They aim to improve the quality of system design.

Formal specification techniques allow for a precise and unambiguous description of system properties. Formal analysis techniques can be used to verify whether a system satisfies its specification. They can significantly reduce the risk of damage caused by design and specification errors in systems.

A formal specification of a system can aid not only in achieving a better (modular) description but also in gaining a deeper understanding and a more abstract view of the system. Formal verification, supported by automated tools, can detect errors in the design that are not easily found using testing and can be used to establish the correctness of the design.

3.6 Process Algebras

A process algebra focuses on the specification and manipulation of process terms induced by a set of operators. Most process algebras contain basic operators to build complex processes. A structural operational semantics is used to formally give each process a semantic representation. This representation is often expressed in the form of a transition system.

A process algebra can be extended by adding new operators to enhance its expressiveness or to facilitate the specification of system behaviour. Several process algebras have been standardized by ISO,

namely CCS (Calculus of Communicating Systems)[29], CSP (Communicating Sequential Processes)[15], and LOTOS [5] provide excellent frameworks for describing communicating concurrent systems, and they are well-equipped for studying their behavioural properties. Process algebras like LOTOS have been the subject of work aiming to enrich them with temporal and mobility information, such as D-LOTOS [33], DD-LOTOS[24], and Mobile DD-LOTOS [27].

Our work falls within the framework of specifying organizational multi-agent systems, which are dynamic and distributed systems involving the concepts of locality and mobility of agents from one site to another. In our approach, we utilized a distributed communicating language, DD-LOTOS, which allows for specifying distributed systems. This specification is operationally translated into the semantic model C-DATA [28] for potential formal verification. DD-LOTOS is defined on another semantic model known as true concurrency instead of the classical interleaving semantics. It incorporates both temporal constraints and action durations. The following section will provide an insight into DD-LOTOS language.

3.7 Distributed D-LOTOS Language

DD-LOTOS (Distributed Durational Language Of Temporal Ordering Specification) [24] is a formal language based on true concurrency semantics called maximality semantics [25][26]. DD-LOTOS is a programming language that supports real-time distributed systems. It comes with operators like restriction, latency, and delay that permit specifying real-time systems. The concept of locality or site is important in specifying the distributed nature of this language. The DD-LOTOS language uses a specific syntax depicted in Table 1.

Table 1: Syntax of DD-LOTOS

$E ::=$	**Behaviours**	
	$stop \mid exit\{d\} \mid \Delta^d E \mid X[L] \mid$	
	$g@t[SP];E \mid i@t\{d\};E \mid hide\, L\, in\, E \mid$	
	$E[]E \mid E \mid [L] \mid E \mid E \gg E \mid E[> E \mid$	
	$a!v\{d\};E$	Emission
	$a?x;E$	Reception
	$go(l,\mathrm{E})\{d\}$	Migration
	$create(l,\mathrm{E})$	Creation of locality
$S ::=$	**Systems**	
	$\phi \mid S \mid S \mid l(E)$	

4 Proposed Aproach

In our previous work [32], we introduced a transformation approach that generates DD-LOTOS specifications from the AGR model. The fundamental idea is to translate each concept in the AGR model into its equivalent in the DD-LOTOS language.

The AGR model defines the system through three primary concepts: Agent, Group, and Role. In contrast, the DD-LOTOS language specifies the system using a set of processes. Consequently, in our approach, we assumed that the system's behaviour is represented as a collection of interacting agents, where each agent describes the behaviour of an object in the system. Subsequently, all agents are translated into a set of processes in the DD-LOTOS language.

Our proposed approach comprises three steps. Initially, all agents are transformed into DD-LOTOS processes. The resulting processes from this step constitute the reserved section for process declarations in the specification of DD-LOTOS . Secondly, groups in the AGR model are transmitted into localities in the DD-LOTOS. A locality serves as an environment that contains a set of processes. In the final step, we established the global specification with its behaviour.

In the final stage, we verified the DD-LOTOS specification generated using the DD-LOTOS tool. In the following section, we will briefly explain the automation of our proposed approach using the Xpand tool. In this paper, our main aim is to illustrate our approach through a detailed example.

The Xpand Code Generation Language Several code generation languages based on the M2T approach exist, such as Xpand [22], which is based on the Java language. In this document, we have

chosen the Xpand language, a template-based language, to generate DD-LOTOS specifications from organizational models. Xpand is a language specialized in code generation from EMF models. To create an Xpand project, we need an EMF model, a check model with the extension '.chk' to define some constraints, and three essential packages containing files of various extensions.

5 An Automatic Approach For Transforming The AGR Model Into DD-LOTOS Code

The objective we aim to achieve is the automation of the transformation of the AGR model to DD-LOTOS using the Model-Driven Engineering (MDE) approach. The principle of the MDE approach is "everything is a model," allowing us to reuse models of formalisms called meta-models, which are adaptable to all platforms. It also enables the manipulation of models through transformations, including Model-to-Model (M2M) and Model-to-Text (M2T) transformations. Transforming a model into text is a specific type of transformation defined by the OMG (Object Management Group) within a model-driven development framework. It follows certain steps to describe the process of transforming a model into text. In this section, we will propose a transformation process using the Xpand tool to generate a formal model from the AGR model.

The approach we have proposed consists of two steps, as illustrated in Figure 4. The first step involves defining a meta-model for the AGR model, and then we submitted the AGR model, as an XMI model to a transformation model to text (M2T), which produces a textual DD-LOTOS specification. For this transformation, we have used predefined templates provided by the Xpand transformation language. After having implemented the AGR meta-model, we will generate the DDLOTOS specifications by using

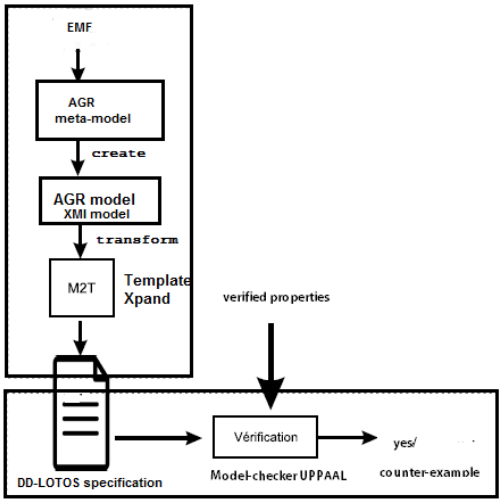


Figure 4: Proposed transformation approach

the Xpand tool in the following step.

6 Case Study

Description To validate the applicability of our approach, we demonstrate our transformation of multi-agent system designed by the AGR model into a DD-LOTOS specification through a case study of Supply Chain Management (SCM)[18]. The process of supply chain management involves modelling the production in companies. This application does several tasks includes receiving a command, producing this command, generating plans of production, changing the plan if certain constraints are not met, negotiating the delivery time and the price, finally, producing products and delivering it. In the supply chain management process, there are three types of actors (Figure 5). The clients place, revise, and delete demands, members of the company, and other companies that are providers of raw materials.

To model and formalize the supply chain management system, we have described it using the AGR model concepts (agents, groups, and roles) for developing the multi-agent system. This organization includes two groups. The first group comprises Client agents who create, changes, or delete orders. Client

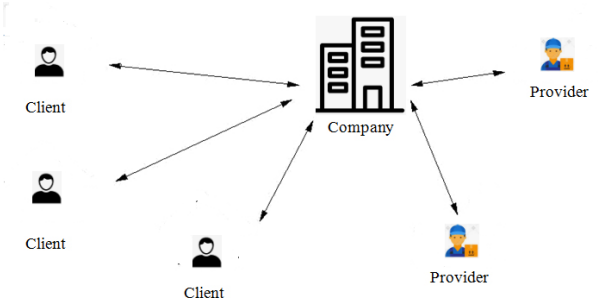


Figure 5: The supply chain management

agents cooperates with Order-Acquisition agent that accepts the demands from clients and negotiates the delay and the price with the Logistics agent. The supervisor of this group is called Logistics agent. It manages the orders of customers with the Order Acquisition agent. Once an order is accepted, Logistics agent requests the Scheduler agent to generate a plan for that specific command. The aforementioned plan is subsequently forwarded to the Dispatcher, Resource, Transporter agents. Another group consists of Provider, Transporter, Resource, and Dispatcher agents. Additionally, agent Scheduler is the representative agent of this group. It coordinates interactions between the agents in two groups as depicted in Figure 6.

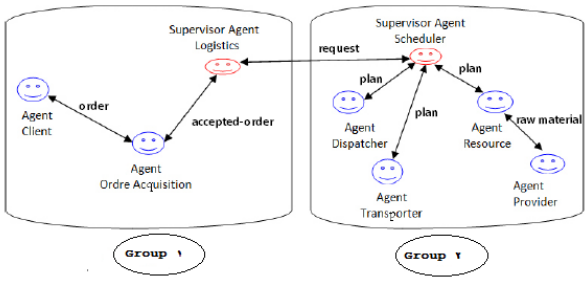


Figure 6: MAS structure of the supply chain management application

The DD-LOTOS Specification Generated From the AGR Model Using Xpand Tool The XMI AGR model, created by EMF tools that which is conform to the AGR meta-model is translated by a Model to Text transformation with Xpand to a DD-LOTOS specification as depicted in Table 2.

7 Conclusion And Future Work

Proposed methods for describing organizational multi-agent systems provide only informal descriptions. Our paper presents a formalization of the organizational model using the DD-LOTOS Formal Language. More exactly, a familiar organizational model is specified, which is the AGR model. We have chosen the DD-LOTOS language because it is characterized by formal semantics and defined on a model of true concurrency semantics. This formalization allows us the validation of organization-centred multi-agent systems.

Therefore, an M2T transformation approach is involved to specify formally the AGR model. The approach presents the Agent-Group-Role model in clear and formal terms. Additionally, the specification decomposes the model into reusable concepts for distinct applications. Using formal notations to specify the AGR model enables the creation of accurate organizational descriptions. In addition, our approach provides improved assistance for their validation and verification process.Furthermore, following the process described in this paper, we plan to specify other organizational multi-agent models.

Thus, we are creating a library of multi-agent models that can be reused for various purposes. Supply chain management process was examined to describe each concept of the AGR model in a formal way, and a case study was given to emphasize all formalization steps.

In our future work, we plan to focus on formalizing one of the AGR extensions, such as the AGRE, AGRMF, AGRS models, etc.

References

[1] Hosny Ahmed Abbas, Samir Ibrahim Shaheen, and Mohammed Hussein Amin. Organization of multi-agent systems: an overview. *arXiv preprint arXiv:1506.09032*, 2015.

[2] Siam Abderrahim and Ramdane Maamri. A category-theoretic approach to organization-based modeling of multi agent systems on the basis of collective phenomena and organizations in human societies. *Informatica*, 42(4):563–576, 2018.

[3] Estefania Argente, Javier Palanca, Gustavo Aranda, Vicente Julian, Vicente Botti, Ana Garcia-Fornes, and Agustin Espinosa. Supporting agent organizations. In *Multi-Agent Systems and Applications V: 5th International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2007, Leipzig, Germany, September 25-27, 2007. Proceedings 5*, pages 236–245, 2007.

[4] K Suzanne Barber and Cheryl E Martin. Dynamic reorganization of decision-making groups. In *Proceedings of the fifth international conference on Autonomous agents*, pages 513–520, 2001.

[5] Tommaso Bolognesi and Ed Brinksma. Introduction to the iso specification language lotos. *Computer Networks and ISDN systems*, 14(1):25–59, 1987.

[6] Abdelghani Boudjidj, Elkamel Merah, and Mohammed El Habib Souidi. Towards a formal multi-agent organizational modeling framework based on category theory. *Informatica*, 45(2):277–288, 2021.

[7] Guerrouf Fayçal and Allaoua Chaoui. A graph transformation based approach for multi-agent systems reorganization. *Multiagent and Grid Systems*, 15(4):375–394, 2019.

[8] Jacques Ferber and Olivier Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Proceedings international conference on multi agent systems (Cat. No. 98EX160)*, pages 128–135, 1998.

[9] Jacques Ferber, Olivier Gutknecht, and Fabien Michel. From agents to organizations: an organizational view of multi-agent systems. In *International workshop on agent-oriented software engineering*, pages 214–230, 2003.

[10] Jacques Ferber, Fabien Michel, and José Báez. Agre: Integrating environments with organizations. In *International Workshop on Environments for Multi-Agent Systems*, pages 48–56, 2004.

[11] Mahdi Hannoun, Olivier Boissier, Jaime S Sichman, and Claudette Sayettat. Moise: An organizational model for multi-agent systems. In *Ibero-American Conference on Artificial Intelligence*, pages 156–165, 2000.

[12] Vincent Hilaire, Pablo Gruer, Abder Koukam, and Olivier Simonin. Formal specification approach of role dynamics in agent organisations: Application to the satisfaction-altruism model. *International Journal of Software Engineering and Knowledge Engineering*, 17(05):615–641, 2007.

[13] Vincent Hilaire, Abder Koukam, Pablo Gruer, and Jean-Pierre Müller. Formal specification and prototyping of multi-agent systems. In *Engineering Societies in the Agents World: First International Workshop, ESAW 2000 Berlin, Germany, August 21, 2000 Revised Papers 1*, pages 114–127, 2000.

[14] Vincent Hilaire, Olivier Simonin, Abder Koukam, and Jacques Ferber. A formal approach to design and reuse agent and multiagent models. In *International Workshop on Agent-Oriented Software Engineering*, pages 142–157, 2004.

[15] Charles Antony Richard Hoare. Communicating sequential processes. *Communications of the ACM*, 21(8):666–677, 1978.

[16] Bryan Horling and Victor Lesser. A survey of multi-agent organizational paradigms. *The Knowledge engineering review*, 19(4):281–316, 2004.

[17] Jomi Fred Hübner, Laurent Vercouter, and Olivier Boissier. Instrumenting multi-agent organisations with artifacts to support reputation processes. In *International Workshop on Coordination, Organizations, Institutions, and Norms in Agent Systems*, pages 96–110, 2008.

[18] Marc-Philippe Huget. An application of agent uml to supply chain management. In *AOIS@ AAMAS*, 2002.

[19] Jennings and R Nicholas. An agent-based approach for building complex software systems. *Communications of the ACM*, 44(4):35–41, 2001.

[20] Nicholas R Jennings. Agent-oriented software engineering. In *Multi-Agent System Engineering: 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'99 Valencia, Spain*, pages 1–7, 1999.

[21] Nicholas R Jennings. On agent-based software engineering. *Artificial intelligence*, 117(2):277–296, 2000.

[22] Benjamin Klatt. Xpand: A closer look at the model2text transformation language. *Language*, 10(16):2008, 2007.

[23] Mohamed Amin Laouadi, Farid Mokhati, and Hassina Seridi-Bouchelaghem. A formal framework for organization-centered multi-agent system specification: A rewriting logic based approach. *Multiagent and Grid Systems*, 13(4):395–419, 2017.

[24] TM Maarouk, DE Saïdouni, and M Khergag. Dd-lotos: A distributed real time language. In *Proceedings 2nd Annual International Conference on Advances in Distributed and Parallel Computing (ADPC 2011) Special Track: Real Time Embedded Systems (RTES 2011)*, pages 45–50, 2011.

[25] Toufik Messaoud Maarouk, Mohammed El Habib Souidi, Makhlouf Ledmi, and Samra Sabeg. Formalization of bpmn gateways using the dd-lotos formal language. *Journal of Communications Software and Systems*, 19(4):254–263, 2023.

[26] Toufik Messaoud Maarouk, Elkamel Merah, Sara Ghaoui, and Nihed Rahabi. Formal semantics and transformation of bpmn models. *International Journal of Business Process Integration and Management*, 9(3):158–169, 2019.

[27] Toufik Messaoud Maarouk, Djamel-Eddine Saïdouni, and Mohamed Khergag. Towards a calculus for distributed, real-time and mobile systems. *J. Softw.*, 7(3):564–574, 2012.

[28] Toufik Messaoud Maarouk and Mohammed El Habib SOUIDI Nadia HOGGAS. Formalization and model checking of bpmn collaboration diagrams with dd-lotos. *Computing & Informatics*, 40(5):1080–1107, 2021.

[29] Robin Milner. *Communication and concurrency*, volume 84. 1989.

[30] Fabio T Muramatsu, Tomas M Vitorello, and Anarosa AF Brandão. Towards organizational interoperability through artifacts. *Agent Environments for Multi-Agent Systems–10 Years Later*, pages 1–14, 2014.

[31] Gauthier Picard, Jomi Fred Hübner, Olivier Boissier, and Marie-Pierre Gleizes. Reorganisation and self-organisation in multi-agent systems. In *1st International Workshop on Organizational Modeling, ORGMOD*, pages 66–80, 2009.

[32] SAMRA SABEG, TOUFIK MESSAOUD MAAROUK, and MOHAMMED EL HABIB SOUIDI. A new formal multi-agent organization based on the dd-lotos language. *Journal of Information Science and Engineering*, 40:1273–1295, 2024.

[33] DE Saïdouni and JP Courtiat. Prise en compte des durées d'action dans les algèbres de processus par l'utilisation de la sémantique de maximalité. In *Proceedings of CFIP*, 2003.

[34] Djamel Eddine Saidouni. *Sémantique de maximalité: application au raffinement d'actions dans LOTOS*. PhD thesis, 1996.

[35] Onn Shehory and Arnon Sturm. Multi-agent systems: a software architecture viewpoint. *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*, 978:57–78, 2014.

[36] Mohammed El Habib Souidi, Piao Songhao, Li Guo, and Chang Lin. Multi-agent cooperation pursuit based on an extension of aalaadin organisational model. *Journal of Experimental & Theoretical Artificial Intelligence*, 28(6):1075–1088, 2016.

[37] Joseph Upton, Ivo Janeka, and Nalton Ferraro. The whole is more than the sum of its parts: Aristotle, metaphysical. *Journal of Craniofacial Surgery*, 25(1):59–63, 2014.

[38] Henry van Dyke Parunak and James Odell. Representing social structures in uml. In *Proceedings of the fifth international conference on Autonomous agents*, pages 100–101, 2001.

[39] Michael Wooldridge, Nicholas R Jennings, and David Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems*, 3:285–312, 2000.

[40] Franco Zambonelli and H Van Dyke Parunak. From design to intention: signs of a revolution. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pages 455–456, 2002.

Table 2: The DD-LOTOS specification of a system designed by the AGR model

Specification supply chain management[order, accepted-order, request, plan, raw-material]:=
Behaviour
Group1(E) | Group2(P)
Where
Process E[order, accepted-order]:=
Agent Client[order, accepted-order]|| Agent order Acquisition[order, accepted-order] || Agent logistics [order, accepted-order]
Where
Process Agent Client[order, accepted-order]:=
$order\,!\,command\,;\,exit$
EndProc
Process Agent order Acquisition[order,accepted-order]:=
accepted-order $?\,x : Message\,;\,exit$
EndProc
EndProc
Process P[request, plan, raw-material]:=
Agent scheduler[request, plan] || (Agent transporter [request, plan] ||| Agent dispatcher [request, plan] ||| Agent resource[request, plan] | [raw-material] | Agent provider [raw-material])
Where
Process Agent scheduler[request, plan]:=
$request\,?\,x : Message$
$plan\,!\,Distribute\,;\,exit$
EndProc
Process Agent transporter [request, plan]:=
$plan\,?\,x : Message$
$;\,exit$
EndProc
Process Agent dispatcher [request, plan]:=
$plan\,?\,x : Message$
$;\,exit$
EndProc
Process Agent resource[request, plan]:=
$plan\,?\,x : Message$
$;\,exit$
EndProc
Process Agent provider [raw-material]:=
$plan\,?\,x : Message$
raw-material $!\,materials\;\;;\,exit$
EndProc
EndProc
EndSpec

HMM-Based Multi-Heartbeat Phonocardiogram Classification Using Wavelet Cepstral Coefficients

Touahria Rima[1], Hacine Gharbi Abdenour[1], Messaoudi Noureddine[2], Ravier Philippe[3], and Roubhi Hamza[1]

[1]*LMSE Laboratory, University of Bordj Bou Arréridj, Elanasser, 34030 Bordj Bou Arréridj, Algeria*
[2]*LIST Laboratory, Faculty of Technology, University of Boumerdes, 35000 Boumerdes, Algeria*
[3]*PRISME Laboratory, University of Orleans, 12 rue de Blois, 45067 Orleans, France*
touahria.rimaa@gmail.com, gharbi07@yahoo.fr, n.messaoudi@univ-boumerdes.dz,,
philippe.ravier@univ-orleans.fr, hamza.roubhi@univ-bba.dz

Abstract

Heart sound classification systems often rely on analyzing a single heartbeat to classify phonocardiogram (PCG) signals. This study introduces a novel approach for classifying multi-heartbeat PCG signals as normal or abnormal, leveraging Wavelet Cepstral Coefficients (WCC) extracted from the Discrete Wavelet Transform (DWT). A Hidden Markov Model (HMM) classifier, associated with a Gaussian Mixture Model (GMM), is bases this system on the modeling of each class. The aim of this work is to develop an effective system for classification of multi-heartbeat PCG signals. The proposed system was evaluated on a subset of the PASCAL heart sounds classification challenge, using the Classification Rate (Acc_HTK) as the primary performance metric. The optimal configuration was obtained with an HMM model comprising 8 states, each associated with 3 Gaussians. A 20 ms analysis window was used. The WCC descriptor, computed using the db7 wavelet with a decomposition level of 6, further improved performance, achieving a classification rate of 97.73 %. These results highlight the effectiveness of WCC descriptors in PCG signal classification and demonstrate the potential of HMM-based multi-heartbeat classification for improved heart sound analysis.

Keywords: Multi-heartbeat PCG signals, Feature extraction, Wavelet Cepstral Coefficients, Hidden Markov Model, Classification.

1 Introduction

Auscultation is the process of listening to heart sounds using a stethoscope, and when recorded, it produces a phonocardiogram (PCG). This technique is crucial for diagnosing cardiovascular diseases (CVDs), which are among the leading causes of mortality worldwide [1].PCG analysis provides valuable insights into the location and morphology of heart sounds, aiding in early detection and diagnosis. In a healthy person, two primary sounds "lub ... dub..." are heard during each cardiac cycle, corresponding to the first heart sound (S1) and the second heart sound (S2), respectively. It is evident that a Lub sound always appears between two Dub sounds, and vice versa. Additionally, the amplitude and duration of S1 are greater than those of S2. These characteristics, including the positioning and structure of heart sounds, provide valuable information and are therefore utilized for heart sound (beat) classification [2]. Doctors can detect additional or abnormal heart sounds by identifying irregular rhythms such as "lub-lub... dub" or "lub... dub-dub" through auscultation [3].

The classification phase usually consists of three fundamental steps: preprocessing, feature extraction and decision-making for classification. First, preprocessing is a crucial step in classification that involves preparing raw data for machine learning models. It involves noise removal using filtering techniques, segmentation to detect S1 and S2 sounds, and normalization for consistency. Secondly, features extraction is an essential step in which the classification system is built; it transforms each heartbeat sound signal into a sequence of vectors. Among them are discrete wavelet transform (DWT) coefficients, introduced by Mei et al. [4]. Kui et al. [5] combined MFSC to enhance heart sound classification, while Li et al. [6] used Short-Time Fourier Transform (STFT) features. Tschannen et al. [7] employed wavelet analysis for feature extraction. Meanwhile, Li F. et al. [8] extracted 497 time-series features to be used as inputs for convolutional neural network (CNN). Additionally, Er [9] proposed utilizing local binary pattern (LBP) and local ternary pattern (LTP) features as inputs for neural networks. Wu et al. [10], which focuses on applying an ensemble (CNN) model combined with a Savitzky–Golay filter for phonocardiogram (PCG)

signal classification. Wavelet cepstral coefficients proposed by [11], Ajit and Swanirbhar [12] explored the use of power spectral density (PSD) for feature extraction in heart sound classification. Zheng et al. [13] employed entropy-based features to analyze phonocardiogram (PCG) signals, utilizing entropy as a measure of signal complexity and irregularity. Touahria et al, [14] which investigates the classification of heart sounds using energy-based features.

The final stage involves classification, where an appropriate classifier is selected to make accurate decisions based on the extracted features. In [15] sound classification, neural networks (NN) are commonly used. Milani et al. [16] deep learning techniques for this task, but challenges persist due to the lack of a comprehensive, publicly available heart sound dataset. To address this, Li et al. [17] proposed a novel approach that incorporates enhanced mel-frequency cepstral coefficient (MFCC) features and deep residual learning for improved classification performance. In many studies, Hidden Markov Models (HMMs) have been employed for PCG modeling and analysis. One such study was conducted by [18] proposed to combine HMM with MFCCs, achieving over 95% sensitivity and specificity but lacked a separate test set. Chauhan et al. [19] refined the approach, reporting 99.21% accuracy on 1381 heart cycles, though their method risked overfitting. Saracoglu et al. [20] applied HMM to frequency spectra, optimizing parameters and achieving 97.5% accuracy on a 60-recording test set. Touahria et al. [21] proposed to combine HMM with logarithmic wavelet energy (LWE), achieving an impressive classification rate of 93.68%. Touahria et al. [22] using wavelet transform techniques to extract features from phonocardiogram (PCG) signals for classification using Hidden Markov Models (HMMs) They reported an accuracy of 92.74% in the discrimination between abnormal and normal heartbeats.

This study introduces a novel approach for classifying multi-heartbeat PCG signals as normal or abnormal, leveraging Wavelet Cepstral Coefficients (WCC) extracted from the Discrete Wavelet Transform (DWT). By utilizing a Hidden Markov Model (HMM) classifier combined with a Gaussian Mixture Model (GMM), this method aims to improve classification rate.

The organization of the structure of this study is as follows. Section 2 will show the suggested approaches for multi- heartbeat PCG signals classification. The experiment and its findings are presented in Section 3. Section 4 concludes the paper.

2 Classification of Multi-Heartbeat PCG Signals

2.1 Database

To test our methods, we utilized the PASCAL Classifying Heart Sounds Challenge database [23]. This database includes two datasets:

- **Dataset A:** Collected from the general public using the iStethoscope Pro iPhone app.
- **Dataset B:** Obtained from clinical trials in hospitals using the digital stethoscope DigiScope.

To evaluate this work, only 420 signals with different cardiac cycles including 196 pathological cardiac cycles were used. The extraction and recording process was performed using the PRAAT software [24], and each cycle was resampled to 16 kHz. The files were then split into two sub-databases: one for the training phase, consisting of 70% of the heart sound signals, and the other for the testing phase, comprising the remaining 30%. In addition, each sound file was paired with a labeling file that includes a transcription of the heart sound class. Each labeling file has the same name as the corresponding sound file but with a (with a `.lab` extension. These transcription files are used during the class modeling and system evaluation phases.

2.2 Feature Extraction Method

Figure 1 illustrates the block diagram of the proposed feature extraction method.

As shown in this figure, this method incorporates three types of features: DWE (Discrete Wavelet Energy) is based on wavelet transform decomposition, where the signal is divided into multiple frequency sub-bands, and the energy of the wavelet coefficients at different levels is computed. which is evaluated as:

1. **Discrete Wavelet Energy (DWE):** Based on the wavelet transform decomposition, where the signal is divided into multiple frequency sub-bands and the energy of the wavelet coefficients at

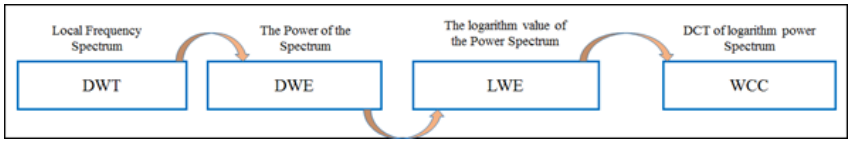


Figure 1: Block diagram illustrating the calculation of the WCCs, LWEs, and DWEs features extraction [22].

different levels is computed.

$$\mathrm{DWE}[d_j] = \sum_{n=0}^{N_j-1} |d_j[n]|^2 \quad \text{for} \quad j = 1, \ldots, p \tag{1}$$

$$\mathrm{DWE}[a_p] = \sum_{n=0}^{N_p-1} |a_p[n]|^2 \tag{2}$$

2. **Log Wavelet Energy (LWE):** Applies a logarithmic transformation to the DWE values.

$$\mathrm{LWE}[d_j] = \log\left(\sum_{n=0}^{N_j-1} |d_j[n]|^2\right) \quad \text{for} \quad j = 1, \ldots, p \tag{3}$$

$$\mathrm{LWE}[a_p] = \log\left(\sum_{n=0}^{N_p-1} |a_p[n]|^2\right) \tag{4}$$

3. **Wavelet Cepstral Coefficients (WCCs):** Obtained by applying the inverse discrete cosine transform (DCT) on the logarithmic energy values.

2.3 Hidden Markov-based Classification System

Generally, several classification methods have been proposed for PCG classification systems to enhance performance, either by reducing complexity or improving classification rate. In [22], the authors proposed a method for classifying heartbeat sounds into normal and abnormal classes. In this study, we propose the implementation of a classification system of multi-heartbeat PCG signal based on Hidden Markov Models (HMMs) [18], where each class (normal and abnormal) is modeled using an HMM [21]. This system consists of a training phase and a testing phase, both of which require an acoustic analysis step to extract relevant parameters for classification. The following figure illustrates the diagram of the classification system.

In the training phase, each class namely, normal and abnormal heart sounds is modeled using a dedicated Hidden Markov Model (HMM) comprising N states. These states are designed to capture the temporal dynamics of the multi-heartbeat PCG signal. To enhance the modeling capability of each state, we associate it with a Gaussian Mixture Model (GMM), allowing the emission probabilities to flexibly represent the underlying statistical distribution of the extracted features. The model parameter including state transition probabilities, mixture weights, mean vectors, and covariance matrice are iteratively re-estimated using the Baum-Welch algorithm, which performs Expectation-Maximization EM to maximize the likelihood of the observed training data. This procedure is implemented using the `HErest` tool from the Hidden Markov Model Toolkit (HTK) [25], which provides robust facilities for training HMMs on time-series data.

In the testing phase, a new PCG signal undergoes the same preprocessing and feature extraction steps as in the training phase, resulting in a sequence of observation vectors. These vectors are then evaluated against the previously trained HMMs. The classification decision is made by computing the log-likelihood of the observation sequence under each class-specific HMM. The Viterbi algorithm is employed to find the most probable sequence of hidden states that best explains the observed data. The signal is classified into the class whose HMM yields the highest likelihood. This decoding and classification process is performed using the `HVite` command from the HTK toolkit [25], which supports efficient implementation of the Viterbi decoding for continuous HMMs.

Figure 2 shows the diagram of the proposed automatic classification system.

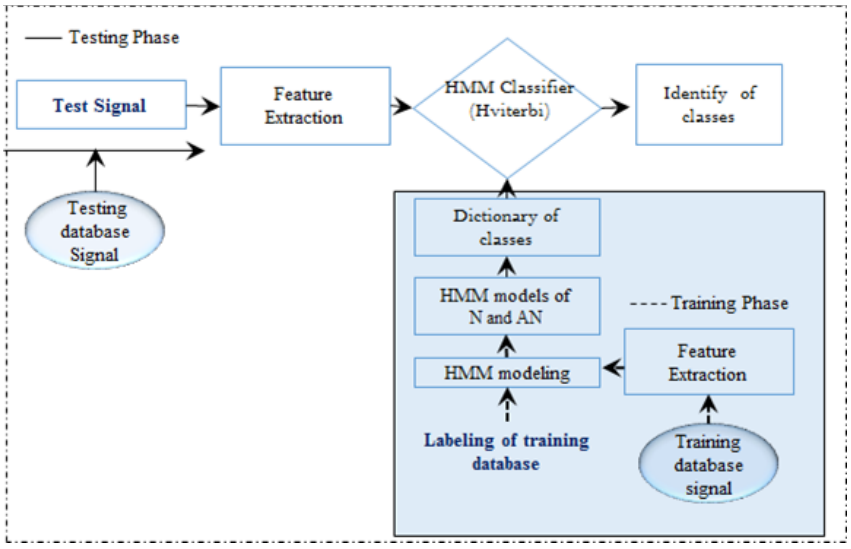


Figure 2: An automatic classification system of multi- heartbeat PCG signal based on HMM models

2.4 Performance Evaluation

The performance of the system is evaluated using the classification rate (Acc_{HTK}) defined as:

$$Acc_{HTK} = \frac{H}{N} \times 100, \tag{5}$$

Where: - H is the number of correctly recognized signals -N is the total number of signals in the reference transcription [25].

3 Experimental Results

3.1 Experimental

The following section presents the experimental results and is divided into two parts. The first part compares the performance of the newly selected features with other feature sets. The second part discusses an experiment aimed at identifying the optimal mother wavelet and decomposition level for the best previously determined descriptor. The system is implemented utilizing the HTK library [25] with its classification performance evaluated based on the classification rate((Acc_{HTK})

3.2 Comparative Study Between Different Features

Table 2 shows the best classification results achieved with the optimal number of HMM states and Gaussians.

The feature vector was generated using sliding Hamming windows of 20 ms with a 50% overlap [22]. MFCC (39 features) achieved an Acc_HTK of 89.77% using 2 Gaussians and 8 states. MFCC, a widely used method with 39 features, achieved the lowest accuracy (89.77%) with 2 Gaussians and 8 states. In contrast, DWE, LWE, and WCC, each with only 8 features, utilized 6 Gaussians, leading to improved recognition accuracy. DWE and LWE performed better than MFCC, achieving 93.18% and 90.91% accuracy, respectively. Notably, WCC outperformed all other methods with an accuracy of 97.73%, indicating its superior ability to extract discriminative features for classification. Despite using fewer features, WCC proved to be the most effective.

3.3 Optimal LWE Parameterization

3.4 Window Duration

Table ?? presents the Acc_HTK variations corresponding to different window duration values. In this experiment, the classification system states were analyzed using `db2` wavelets at level 7 [22]. The results

Table 1: Comparison of classification rate (Acc_HTK %) for different feature extraction descriptors using Daubechies (db2) at level 7 with the optimal HMM configuration [20].

Feature	MFCC (39)	DWE (8)	LWE (8)	WCC (8)
HMM States	8	8	10	8
Gaussians	2	6	6	6
Accuracy (%)	89.77	93.18	90.91	97.73

indicate that the highest accuracy (Acc_HTK) in each column is achieved when the window size is set to 20 ms, with a peak accuracy of 97.73%. Consequently, a window duration of 20 ms is the optimal choice for this classification task.

Table 2: Classification rate (Acc_HTK %) for different combinations of the hamming window sizes

Wind. size	**60ms**	**50ms**	**40ms**	**30ms**	**20ms**
$\boldsymbol{Acc_{HTK}}$(%)	90.91	94.32	93.18	93.18	**97.73**

3.5 Wavelet Family and Decomposition Depth

The influence of various wavelet families and decomposition levels on accuracy was examined. Table **??** summarizes the performance of different Daubechies orders and decomposition levels. (Note: The table below is a simplified representation based on the provided data.)

This section analyzes the smoothness and impact of various wavelet families on Acc_{HTK} accuracy, aiming to identify the optimal mother wavelet and its most effective decomposition level. The study investigates three wavelet families: Daubechies (Db1–Db8), Coiflets (Coif1–Coif5), and Symlets (Sym1–Sym8). To ensure robust evaluation, the classification system employs the optimal descriptor identified in previous research, which utilizes a ten-state Hidden Markov Model (HMM) with three Gaussian mixtures.

As shown in Table 4, the highest accuracy of 97.73% was achieved using the Daubechies wavelet of order 2 with a 7-level decomposition, demonstrating its superior performance in this classification task.

Table **??** presents the detailed Acc_{HTK} results for the best-performing Daubechies wavelet family across various decomposition levels and wavelet orders. The results highlight a considerable range in Acc_{HTK} values—from a minimum of 84.09% to a peak of 97.73%—emphasizing the importance of selecting optimal wavelet parameters.

Table 3: Comparison of Acc_HTK (%) of WCC for different Daubechies orders and decomposition levels.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
db1	94.32	94.32	90.91	93.18	86.36	92.05	90.91	90.91
db2	86.36	86.36	86.36	94.32	95.45	93.18	**97.73**	94.32
db3	86.36	87.50	89.77	96.59	88.64	85.23	88.64	
db4	86.36	84.09	93.18	90.91	93.18	92.05	87.50	
db5	85.23	86.36	93.18	94.32	94.32	90.91		
db6	86.36	87.50	93.18	92.05	93.18	96.59		
db7	93.18	85.23	87.50	94.32	93.18	94.32		
db8	92.05	90.91	87.50	85.23	95.45	94.32		

Additionally, results were obtained using the Coiflets and Symlets wavelet families following the same experimental protocol. Within the Symlet family, order 1 at level 7 achieved the best performance, with a Acc_HTK of 90.91%. Similarly, within the Coiflets family, order 5 at level 5 demonstrated the highest performance, achieving a Acc_HTK of 94.32%. The results, presented in Table 4, show that the highest classification rate (Acc_HTK) of 97.73% was achieved using the Daubechies wavelet with order 2 and a decomposition level of 7.

In conclusion, based on the conducted experiments, the WCC descriptors achieved the highest classification rates when derived using Daubechies order 2 with level 7 for Daubechies

Table 4: Comparative results between different kinds of wavelet families. The table shows the Acc_HTK values for the optimal decomposition level as well as the optimal order for each wavelet family.

Wavelet Family	Level	Order	Acc_HTK (%)
Daubechies	7	2	97.73
Symlet	7	1	90.91
Coiflets	5	5	94.32

4 Discussion and Conclusions

The experimental results indicate that the choice of wavelet family and decomposition level significantly impacts classification rate. The Daubechies wavelet of order 2 at level 7 demonstrated the highest Acc_HTK, confirming its suitability for multi-heartbeat PCG signal classification. Additionally, the window duration experiment showed that a 20ms Hamming window provides optimal performance. These findings emphasize the importance of feature extraction techniques in improving classification rate. This study proposes a novel approach for classifying multi-heartbeat PCG signals using Wavelet Cepstral Coefficients (WCC) and a Hidden Markov Model (HMM). The system achieved a high classification rate of 97.73% on a subset of the PASCAL heart sounds classification challenge, demonstrating its effectiveness. The results highlight that the optimal configuration involves using the Daubechies order 2 wavelet at a decomposition level of 7 with WCC descriptors. Future work could explore alternative machine learning models to further improve classification performance. Additionally, testing on larger and more diverse datasets could enhance the generalizability of the proposed method.

References

[1] Ghosh, S. (2020). Automated detection of heart valve diseases using chirplet transform and multiclass composite classifier with PCG signals. *Comput. Biol. Med.*

[2] Kumar, D., Carvalho, P., Antunes, M., Gil, P., Henriques, J., & Eugenio, L. (2006). New algorithm for detection of S1 and S2 heart sounds. In *Proc. IEEE ICASSP*, Vol. 2, pp. 1180–1183.

[3] Gomes, E., & Pereira, E. (2012). Classifying heart sounds using peak location for segmentation and feature construction. *AISTATS*, pp. 1–5.

[4] Mei, N., Wang, H., Zhang, Y., Liu, F., Jiang, X., & Wei, S. (2021). Classification of heart sounds based on quality assessment and wavelet scattering transform. *Comput. Biol. Med.*, 137, 104814.

[5] Kui, H., Pan, J., Zong, R., Yang, H., & Wang, W. (2021). Heart sound classification based on log Mel-frequency spectral coefficients features and convolutional neural networks. *Biomed. Signal Process. Control*, 69, 102893.

[6] Li, T., Yin, Y., Ma, K., Zhang, S., & Liu, M. (2021). Lightweight end-to-end neural network model for automatic heart sound classification. *Information*, 12, 54.

[7] Tschannen, M., Kramer, T., Marti, G., Heinzmann, M., & Wiatowski, T. (2016). Heart sound classification using deep structured features. In *Computing in Cardiology*, pp. 565–568.

[8] Li, F., Tang, H., Mathiak, K., & Cong, F. (2020). Classification of heart sounds using convolutional neural network. *Appl. Sci.*, 10, 3956.

[9] Er, M. (2021). Heart sounds classification using convolutional neural network with 1D-local binary pattern and 1D-local ternary pattern.

[10] Wu, J., Tsai, M., Huang, Y., Islam, S., Hassan, M., & Alelaiwi, A. (2019). Applying an ensemble convolutional neural network with Savitzky–Golay filter to construct a phonocardiogram prediction model. *Appl. Soft Comput.*, 78, 29–40.

[11] Hacine-Gharbi, A., & Ravier, P. (2018). Wavelet cepstral coefficients for electrical appliances identification using hidden Markov models. In *Proc. ICPRAM*.

[12] Ajit, S., & Swanirbhar, M. (2019). Classification of unsegmented heart sound recording using KNN classifier. *Medicine and Biology*.

[13] Zheng, Y., Guo, X., Wang, Y., Qin, J., & Lv, F. (2022). A multi-scale and multi-domain heart sound feature-based machine learning model for ACC/AHA heart failure stage classification. *Physiol. Meas.*, 43, 065002.

[14] Touahria, R., Hacine-Gharbi, A., & Ravier, P. (2023). Feature selection algorithms highlight the importance of the systolic segment for normal/murmur PCG beat classification. *Biomed. Signal Process. Control*.

[15] Barschdorff, B., Bothe, A., & Rengshausen, U. (1989). Heart sound analysis using neural and statistical classifiers: a comparison. *Comput. Cardiol.*, pp. 415–418.

[16] Milani, M., Abas, P., & Silva, L. (2022). A critical review of heart sound signal segmentation algorithms. *Smart Health*, 24, 100283.

[17] Li, S., Li, F., Tang, S., & Luo, F. (2021). Heart sounds classification based on feature fusion using lightweight neural networks. *IEEE Trans. Instrum. Meas.*, 70, 1–9.

[18] Wang, P., Lim, C., Chauhan, S., Foo, J., & Anantharaman, V. (2007). Phonocardiographic signal analysis method using a modified hidden Markov model. *Ann. Biomed. Eng.*, 35, 367–374.

[19] Chauhan, S., Wang, P., Lim, C., & Anantharaman, V. (2008). A computer-aided MFCC based HMM system for automatic auscultation. *Comput. Biol. Med.*, 38, 221–233.

[20] Saracoglu, R. (2012). Hidden Markov model-based classification of heart valve disease with PCA for dimension reduction. *Eng. Appl. Artif. Intell.*, 25, 1523–1528.

[21] Touahria, R., Hacine-Gharbi, A., & Ravier, P. (2024). Phonocardiogram segmentation based on HMM modelling combined with LWE: Application for heart valve disorder classification. In *NCASEE'24*.

[22] Touahria, R., Hacine-Gharbi, A., & Ravier, P. (2021). Discrete wavelet-based features for PCG signal classification using hidden Markov models. In *Proc. ICPRAM*.

[23] Bentley, P., Nordehn, G., Coimbra, M., Mannor, S., & Getz, R. (2011). The PASCAL classifying heart sounds challenge.

[24] Praat. (n.d.). https://praat.fr.softonic.com/

[25] Young, S., Kershaw, D., Odell, J., & Ollason, D. (1999). *The HTK Book*. Cambridge: Entropic Ltd.

Genetic Algorithm Learning Operators to Solve the Vehicle Routing Problem

Souad Abdoune and Menouar Boulif

LIMOSE laboratory, Department of Computer Science, Mhamed Bouguerra University Boumerdes, Algeria

s.abdoune@univ-boumerdes.dz, boumen7@gmail.com

Abstract

In recent years, researchers have focused on solving real-world optimization problems that impact logistics and transportation. Among these, the Vehicle Routing Problem (VRP) and its various variants have gained significant attention due to their wide ranging applications. One of the most widely used techniques for solving VRP is the Genetic Algorithm (GA). As part of the Evolutionary Intelligence approaches, GA leverages its operators to learn how to adapt its prospecting of the problem search space, in order to reach efficiently good solutions. This paper provides an overview of GA operators tailored to solve VRP, and evaluates different operator configurations for the Capacitated VRP by using a benchmark set taken from the literature, to trial their learning performance.

Keywords: Evolutionary Intelligence, Combinatorial Optimization, Vehicle Routing Problem, Genetic Algorithm, Genetic operators, Selection, Crossover, Mutation.

1 Introduction

Logistics, transportation, and supply chain management have been among the key areas of research in recent years. Two fundamental optimization challenges in these domains are the Traveling Salesman Problem (TSP) and the Vehicle Routing Problem.

The TSP is a well-known combinatorial optimization problem where a salesman must visit a set of customers exactly once and return to the starting point while minimizing travel distance [1]. Over time, this concept has been extended to incorporate multiple vehicles and additional constraints beyond just minimizing the overall travel distance. This extension is known as the Vehicle Routing Problem (VRP) (Figure1).

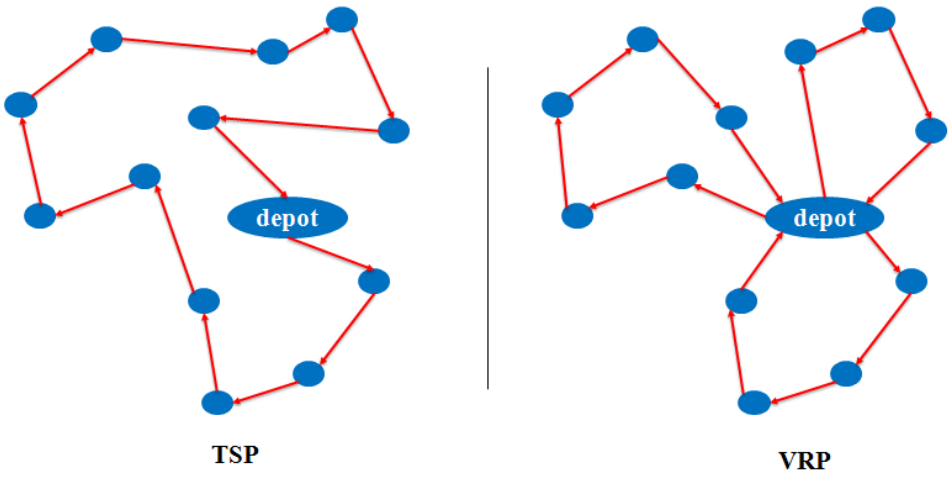


Figure 1: TSP vs. VRP

VRP generalizes the TSP by determining a set of vehicle routes, each assigned to serve a group of customers with known demands. The objective is to minimize the total travel distance while satisfying problem-specific constraints [2]. Due to its broad real-world applications, researchers have continuously enhanced the classical VRP model by introducing additional constraints and various optimization criteria to make it more practical.

Different VRP constraints have led to the development of multiple VRP variants such as (Figure 2):

- Capacitated VRP (CVRP): Involves a fleet of vehicles with limited capacity, where the total quantity delivered on each route must not exceed the vehicle's capacity [3, 4].
- Heterogeneous VRP (HVRP): If all vehicles have the same characteristics (e.g., volume, speed, capacity), the problem is considered homogeneous [5]; otherwise, it is heterogeneous [6].
- VRP with Time Windows (VRPTW): Considers customer availability, ensuring that deliveries are made within a predefined time window [3, 7].
- Periodic VRP (PVRP): Deals with customers who require deliveries on a recurring schedule over a specific planning horizon [8].
- Multi-Depot VRP (MDVRP): Incorporates multiple distribution centers from which vehicles are dispatched to serve customers [9].
- Open VRP (OVRP): In this variant, vehicles are not required to return to the distribution center after completing their routes[10].
- Dynamic VRP (DVRP): Unlike static VRP, this version assumes that customer requests can change in real time during the delivery process. New requests may be added, existing ones may be canceled, or delivery conditions may change, requiring continuous route re-optimization [10, 11].

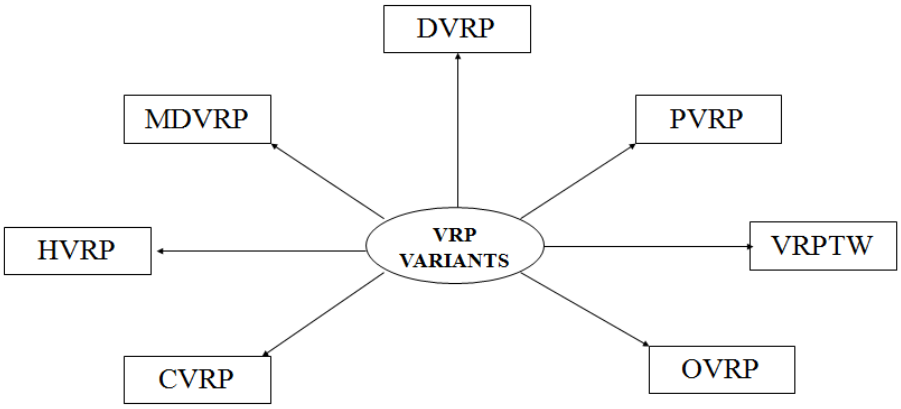


Figure 2: Some VRP variations

The objective function in VRP varies depending on the problem context. It may be defined by using one of the following criteria or a combination of them:

- minimize total distance,
- minimize travel cost,
- minimize penalties,
- maximize customer satisfaction,
- maximize service quality.

These criteria will be revisited with full details in section 2.3

Over the years, various solution approaches have been proposed to tackle the complexity of different VRP variants. These approaches are broadly classified into exact and approximate methods. Since VRP is NP-hard, and thus, finding an optimal solution for large instances is computationally expensive, exact methods become impractical for large-scale problems. Instead, approximate methods (heuristics and meta-heuristics) are the preferred choice. Among them, Genetic Algorithms (GA) have gained popularity due to their ability to explore large solution spaces efficiently and produce high-quality solutions.

In VRP applications, the performance of Genetic Algorithms can be boosted or hindered by the operators to be used, which directly influence solution quality. Indeed, GA operators give the GA its ability to learn from generation to generation how to correct its trajectory towards promising areas of the solutions' search space. This paper surveys the different GA operators applied to solve VRP.

The remainder of this paper is organized as follows: Section 2 provides an overview of the Genetic Algorithm. Section 3 presents different GA operators applied to VRP, Sections 4 and 5 describe the experimental methodology and the results and discussion, respectively. Finally, Section 4 concludes the paper with insights and future research directions.

2 Genetic Algorithm learning mechanism

Genetic Algorithms are a class of metaheuristic optimization techniques inspired by the principles of natural selection and evolution, as described by Charles Darwin [12, 13, 14]. GAs, even in their basic form, embody key concepts of Artificial Intelligence as they can *learn*, *adapt* and *search* for good solutions to problems difficult to solve by humans. GAs are particularly effective in solving complex optimization problems by intelligently exploring the search space and finding optimal or near-optimal solutions within a reasonable time frame. This makes GAs highly useful for NP-hard problems, such as the Vehicle Routing Problem and its variants.

GAs work by maintaining a population of candidate solutions that can learn across multiple generations through the application of genetic operators, including selection, crossover, and mutation. These operations guide the search towards high-quality solutions.

2.1 Key concepts in GAs

To understand how GAs function, the following fundamental concepts should be introduced [15, 16]:

- Population: A set of chromosomes, each representing a potential solution to the problem.
- Chromosome: An encoded representation of a solution in a specific format (e.g., sequence of customer visits in VRP).
- Gene: A component of a chromosome that represents a decision variable (e.g., a customer or a route segment).
- Allele: A specific value that a gene can take.
- Offspring: New chromosomes produced through crossover and mutation operations.
- Objective function: A function that evaluates the fitness (quality) of each solution, based on the problem's optimization goal (e.g., minimizing total travel cost).

2.2 Genetic Algorithm process

The standard process of a GA follows these steps (Figure 3) [17]:

1. Initialization: Generate an initial population of candidate solutions, either randomly or using problem-specific heuristics.
2. Evaluation: Compute the fitness of each chromosome in the population using the objective function.
3. Selection: Choose parent solutions from the population based on their fitness values, ensuring that better solutions have a higher chance of being selected.
4. Reproduction (Crossover & Mutation):
   - Crossover: Combine genetic material from two parent solutions to generate offspring.
   - Mutation: Introduce small random modifications to maintain diversity in the population.
5. Replacement: Replace some or all of the existing population with newly generated offspring, forming the next generation.

6. Termination condition: Check if a predefined stopping criterion is met (e.g., reaching a maximum number of generations, convergence of solutions, or stability in fitness values). If the criterion is met, return the best chromosome as the final solution; otherwise, repeat the process from step 2.

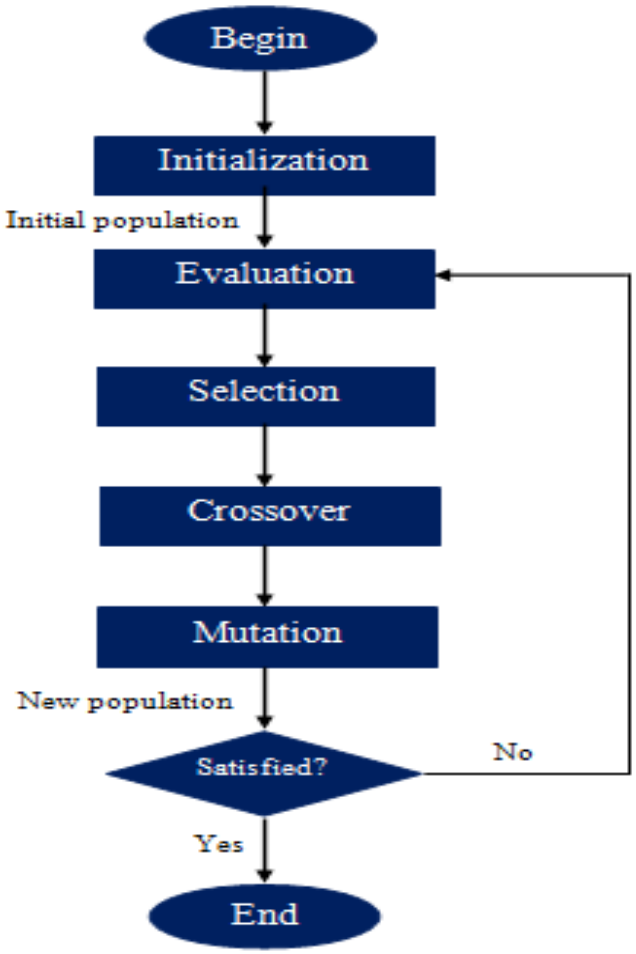


Figure 3: Genetic algorithm flowchart

2.3 Optimization Criteria in VRP

The Vehicle Routing Problem encompasses a variety of optimization objectives, which vary depending on the problem variant and practical application domain. Traditionally, the primary objective in VRP is to minimize the total distance travelled or to optimize the number of routes (vehicles). However, many real-world scenarios introduce additional or alternative goals, such as minimizing total travel time, reducing fuel consumption, balancing the workload among vehicles, or maximizing customer satisfaction by respecting service time windows and delivery preferences.

When applying Genetic Algorithms to solve VRP, the fitness function is a critical component that encodes these objectives into a quantifiable criterion. Depending on the VRP variant, the fitness function can be single-objective (minimizing total cost) or multi-objective (minimizing cost while maximizing service quality), and often involves penalization terms for constraint violations such as time windows. For instance, VRP with Time Window and Capacitated VRP, infeasible solutions may be penalized based on the degree of violation. Furthermore, the GA must be carefully tailored to preserve feasibility during the search process while maintaining diversity in the population. By clearly defining and incorporating these objectives into the evolution process and evaluation mechanisms, GA can effectively explore the solution space and adapt to different VRP scenarios.

2.4 Learning mechanisms in Genetic Algorithms

Since Genetic Algorithms are inspired by biological evolution they are designed to adapt their behaviour over time, a property often referred to as learning.
This feature is further stressed in Adaptive GAs, whre the probabilities of applying genetic operators (selection, crossover, mutation) are dynamically adjusted based on their historical performance across generations. This process enables the algorithm to emphasize the most successful operators, improving convergence speed and maintaining population diversity [18].

Self-adaptive GAs go a step further by encoding control parameters directly into the chromosomes, allowing these parameters to evolve alongside the solutions themselves. This mechanism enables the algorithm to autonomously learn optimal operator settings and adjust to the characteristics of the problem over time [19]. For example, in dynamic VRPs, such approaches automatically increase mutation rates when new customer requests arrive, enabling faster adaptation to changing conditions.

Recently, hybrid approaches have combined self-adaptation with reinforcement learning or deep learning to further enhance the algorithm's ability to navigate complex and dynamic search spaces. These learning-driven strategies are particularly valuable in solving real-world optimization problems such as the Vehicle Routing Problem, where constraints and environmental conditions can vary significantly. Such mechanisms highlight the potential of learning-enabled GAs to intelligently and autonomously explore the solution space, aligning with the goal of achieving robust and efficient optimization in logistics and transportation.

Different Genetic Algorithm operators are designed for specific problem domains. The next section surveys the GA operators commonly applied to VRP.

3 GA operators for VRP

The effectiveness of GAs in solving the Vehicle Routing Problem largely depends on the choice and implementation of the genetic operators. These operators play a crucial role in guiding the search process, maintaining population diversity, and ensuring convergence toward high quality solutions. While the fundamental GA operators (selection, crossover, and mutation) are common across different optimization problems, their adaptation to VRP requires specialized mechanisms to handle route based representations, feasibility constraints, and solution quality. Various modifications of selection, crossover, and mutation have been proposed in the literature to enhance GA performance in VRP. After presenting the initial steps of the GA to solve VRP, the most widely used operators are described in what follows.

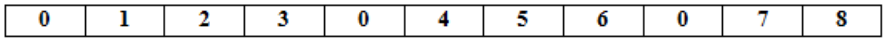
3.1 Chromosome representation

The encoding method directly impacts GA's ability to find optimal or near-optimal solutions efficiently. The chromosome representation in GA is crucial for determining good solutions to VRP. However, discussing all the encoding approaches is beyond the scope of this work. Instead, we present the presentation that allows to understand the described operators.
Among the various encoding techniques proposed in the literature, *path representation* is the most widely used for VRP. In this approach[14]:

- Customers are represented by integer identifiers, with each integer corresponding to a specific customer
- The order of these integers within the chromosome defines the sequence of customer visits
- The Depot Index (typically 0) indicates the start and end points of each route, segmenting the chromosome into separate routes
- Each route is marked by a depot index, with customers visited in the order specified by the sequence of integers

This encoding structure provides a direct and clear representation of customer visit sequences and vehicle routes, as shown in Figure 4a. After constructing the full chromosome, depot indexes can be removed to improve readability, as demonstrated in Figure 4b.

(a) with depot index



(b) without depot index

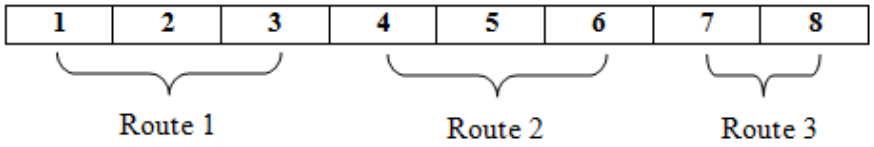


Figure 4: VRP chromosome representations

3.2 Selection operators

The selection process in GAs determines which individuals (solutions) are chosen for reproduction. While no specific selection method is designed exclusively for the Vehicle Routing Problem (VRP) and its variants, the following widely used selection operators can be effectively applied:

- Roulette wheel selection (RWS): assigns a probability to each chromosome based on its fitness. A virtual wheel is spun, and the chromosome closest to the stopping point is selected. Since elements with higher fitness occupy a larger portion of the wheel, they have a greater chance of being chosen[20].
- Elitism selection (ES): preserves the best solutions by directly transferring them to the next generation, ensuring that high-quality solutions are not lost during the evolutionary process [14].
- Rank selection (RS): orders chromosomes based on fitness and assigns selection probabilities accordingly. This method prevents highly fit individuals from dominating the selection process too early, leading to a more balanced exploration of the solution space [14].
- Tournament selection (TS): involves randomly selecting a subset of chromosomes and conducting a competition, where the one with the highest fitness is chosen. This method maintains diversity and prevents premature convergence to local optima [14].

3.3 Crossover operators

The crossover process in genetic algorithms mimics a natural biological phenomenon where genetic material is exchanged between parents to create offspring. The most fundamental crossover methods are *one-point crossover* and *two-point crossover*.

In one-point crossover, a random cutting point along the chromosome is selected, and the segments of the chromosome are exchanged between two parent chromosomes [12]. In two-point crossover, two cutting points are selected, and the portion between these points is swapped between the parents [13].

While these methods are simple and intuitive, they are often not well-suited for combinatorial optimization problems like the Traveling Salesman Problem (TSP) and Vehicle Routing Problem (VRP). One of the main issues is that these basic crossover methods can generate duplicate genes or customer visits within the chromosomes, which is a critical problem in TSP and VRP, where each customer must be visited exactly once in a valid solution. As a result, offspring created through these methods may require post-crossover repair to remove duplicates and restore feasibility.

To address these limitations,several advanced and specialized crossover techniques have been developed. These methods aim to preserve the validity of the solution, ensuring that the offspring generated do not violate the constraints (such as visiting each customer exactly once) and enhancing the efficiency of the genetic search process. Some of the key techniques are:

- Order crossover: This recombination technique is designed for permutation-based problems such as the VRP. It preserves the relative ordering of cities by transferring a contiguous segment from one parent while filling the remaining positions with elements from the second parent in their original

sequence, ensuring that no duplicates occur. The process begins with the selection of two cut points, defining a subsequence to be directly copied into the offspring. The remaining positions are then filled by sequentially inserting elements from the other parent, starting immediately after the second cut point and skipping those already present in the offspring (see Figure 5) [20, 21, 22].

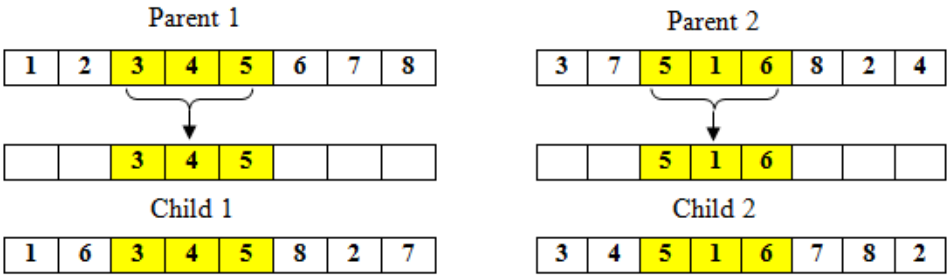


Figure 5: Order crossover

- Cycle crossover: this method used for permutation problems. It works by identifying cycles of genes between two parent solutions and transferring them to the offspring. The process starts by copying genes from Parent 1 to the offspring, then follows the positions of corresponding genes in Parent 2 to complete the cycle. Once a cycle is finished, the remaining genes are copied from the other parent. This ensures a valid permutation without duplicates, preserving the relative order of genes from both parents. It's particularly useful for problems like TSP or VRP. (see Figure 6) [23, 24, 25].

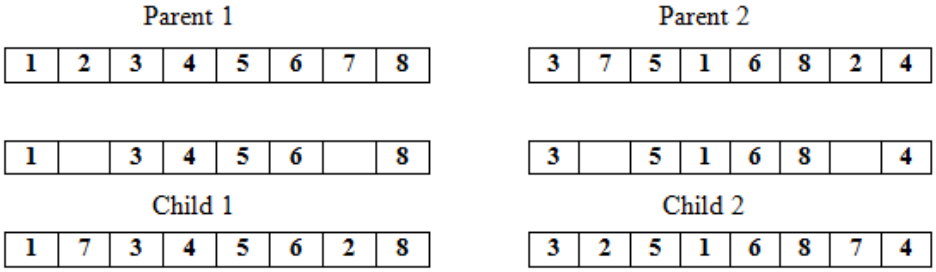


Figure 6: Cycle crossover

- Partially mapped crossover: This method works by selecting a random subsequence from one parent and copying it into the offspring. The remaining positions in the offspring are filled with genes from the other parent in the order they appear, while preserving the relative order of the cities from the first parent. This technique avoids duplicates and preserves the structure of the parent solutions (see Figure 7) [26, 27].

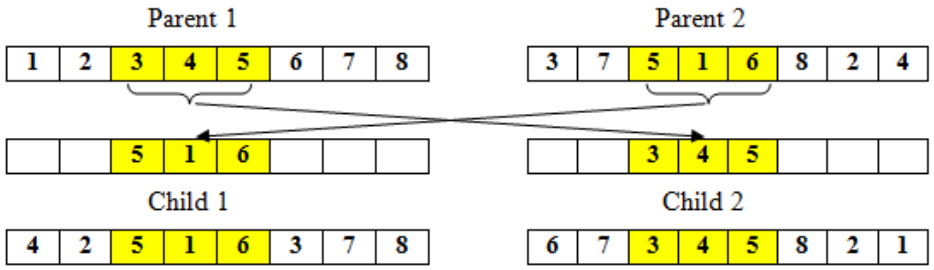


Figure 7: Partially mapped crossover

- Order based crossover: This method begins by randomly selecting a set of positions in Parent 1. Next, the genes from Parent 2 are copied into Child 1, excluding the genes located in the

selected positions of Parent 1. Finally, loop over parent 1 and transfer to child 1 the genes that are not already transferred to it. This ensures that the offspring maintains a valid permutation and preserves key structural properties from both parents. (see Figure 8) [28, 29, 30].

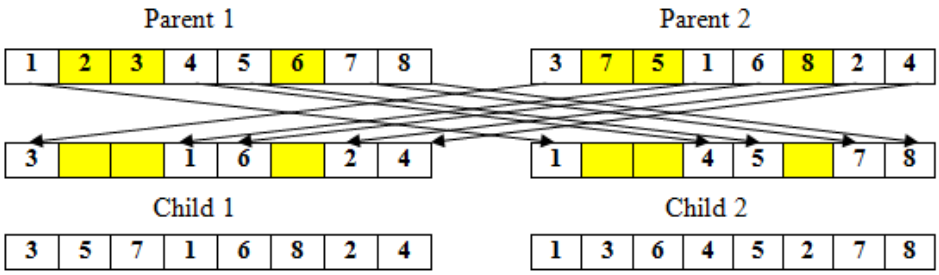


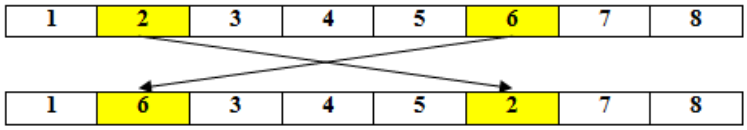
Figure 8: Order based crossover

3.4 Mutation operators

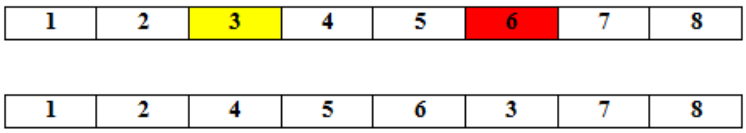
Many mutation types are proposed in the literature, such as:

- Exchange (or Swap) mutation: This mutation operator randomly selects two cities in the tour and exchanges their positions as shown in Figure 9a [31, 23, 32].
- Insertion mutation: The insertion mutation operator randomly chooses a city in the tour, removes it from this tour, and inserts it in a randomly selected place as shown in Figure 9b [33, 34].
- Inversion mutation: This operator randomly selects a sub-tour, removes it from the tour, then inserts it in reversed order at a randomly selected position (see Figure 9c) [33, 35, 34].

(a) Exchange



(b) Insertion



(c) Inversion

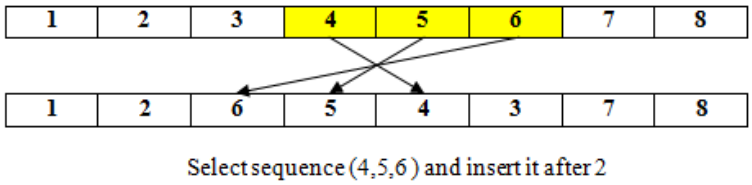


Figure 9: Mutation operators

4 Experimental Methodology

To support the theoretical analysis of genetic algorithm (GA) operators, namely, selection, mutation, and crossover, this study conducted a set of experiments evaluating various combinations of these operators. In total, 48 distinct GA configurations are examined. The objective is to assess the influence of these operators on the algorithms performance in solving the Capacitated Vehicle Routing Problem (CVRP), using a well-established benchmark set [36].

Our experimental analysis focuses on the A-n32-k5 instance, which is characterized by the following features:

- 31 customer locations, each associated with a specific demand.
- A homogeneous fleet consisting of 5 vehicles with identical capacity.
- A known optimal total distance of 784 kilometers.

The experimental setup adopted in this study includes:

- Population size: 100 individuals.
- Maximum number of generations: 500.
- Constraint-handling strategy: penalty functions are applied to penalize individuals that violate the problem's constraints, including capacity limitations and the number of available vehicles.

5 Results and Discussion

The results in Table 1 demonstrate significant variability in performance across the 48 genetic operator combinations. Notably, the OBX crossover paired with inversion mutation and elite selection achieved the lowest distance of 838.42 km, which is close to the benchmarks optimal distance (784 km) by just 6.9%. This configuration also exhibited moderate computation time (4.69 s), suggesting a favourable balance between solution quality and computational effort. Conversely, configurations using CX crossover with insertion mutation (1214.52 km) highlight the risks of poor operator synergy, where premature convergence and limited exploration lead to suboptimal solutions.

Table 2 underscores the superiority of OBX crossover, which achieved both the lowest best-case distance (838.42 km) and the lowest average (950.11 km). In contrast, CX crossover exhibited the highest worst-case distance (1214.52 km) and average (1025.71 km), indicating instability in maintaining solution quality.

Table 2: Distance results for Crossover operators

Crossover operator	**Best**	**Worst**	**Average**
PMX	875.50	1183.41	1016.87
OX	897.57	1089.07	984.62
OBX	**838.42**	1143.70	**950.11**
CX	870.60	**1214.52**	1025.71

Additionally, Table 3 summarizes the performance of each mutation operator across all selection and crossover combinations. The inversion mutation yielded the most promising results overall, providing both the lowest best-case distance and the lowest average distance. This demonstrates its advantage in maintaining solution quality while efficiently exploring the solution space.

Table 3: Performances of Mutation operators

Mutation operator	**Best**	**Worst**	**Average**
Insertion	967.48	1214.52	1059.82
Swap	863.99	1146.60	960.86
Inversion	**838.42**	1178.33	962.25

Table 1: Performance evaluation of genetic operator combinations on the CVRP (Instance A-n32-k5)

Genetic Operators			**Performance Metrics**		
Crossover	**Mutation**	**Selection**	**Distance (Km)**	**Time (s)**	**Convergence**
PMX	Insertion	Elite	1005.27	4.58	51
PMX	Insertion	Tournament	1054.96	2.38	493
PMX	Insertion	Roulette	1183.41	2.62	379
PMX	Insertion	Rank	1047.46	4.53	491
PMX	Swap	Elite	1001.64	5.16	69
PMX	Swap	Tournament	929.07	2.26	257
PMX	Swap	Roulette	1146.60	2.56	460
PMX	Swap	Rank	921.60	5.50	491
PMX	Inversion	Elite	946.69	3.55	74
PMX	Inversion	Tournament	875.50	2.42	440
PMX	Inversion	Roulette	1178.33	2.71	492
PMX	Inversion	Rank	911.89	5.59	491
OX	Insertion	Elite	1010.58	3.66	70
OX	Insertion	Tournament	1014.45	2.50	478
OX	Insertion	Roulette	1013.40	3.50	437
OX	Insertion	Rank	976.00	4.77	390
OX	Swap	Elite	916.75	3.68	100
OX	Swap	Tournament	1047.53	2.80	481
OX	Swap	Roulette	1067.24	3.64	462
OX	Swap	Rank	897.57	4.52	459
OX	Inversion	Elite	977.26	3.84	94
OX	Inversion	Tournament	903.68	3.72	487
OX	Inversion	Roulette	1089.07	2.73	457
OX	Inversion	Rank	901.89	4.50	497
OBX	Insertion	Elite	1143.70	4.52	63
OBX	Insertion	Tournament	1018.18	2.25	486
OBX	Insertion	Roulette	996.05	2.42	439
OBX	Insertion	Rank	1035.97	4.18	393
OBX	Swap	Elite	917.28	4.51	111
OBX	Swap	Tournament	863.99	2.20	486
OBX	Swap	Roulette	918.42	2.38	430
OBX	Swap	Rank	894.42	4.16	486
OBX	Inversion	Elite	838.42	4.69	73
OBX	Inversion	Tournament	872.04	2.29	309
OBX	Inversion	Roulette	983.75	2.43	401
OBX	Inversion	Rank	919.11	4.89	417
CX	Insertion	Elite	1214.52	6.06	40
CX	Insertion	Tournament	967.48	3.60	323
CX	Insertion	Roulette	1127.22	4.50	461
CX	Insertion	Rank	1149.44	5.35	464
CX	Swap	Elite	1005.40	6.61	39
CX	Swap	Tournament	899.72	4.13	468
CX	Swap	Roulette	1034.52	3.73	483
CX	Swap	Rank	911.94	7.04	465
CX	Inversion	Elite	956.47	5.65	81
CX	Inversion	Tournament	870.60	4.51	490
CX	Inversion	Roulette	1115.62	3.59	487
CX	Inversion	Rank	1055.63	5.44	492

Note: All solutions are feasible and utilize all 5 vehicles. Distances reflect total route lengths in kilometers. The Convergence column indicates the generation at which the best solution was first identified.

Regarding selection mechanisms, Table 4 illustrates that elitism consistently helped in preserving high-quality individuals, while tournament selection offered a good balance between exploration and exploitation. Roulette and rank selection performed moderately, with less consistent results.

Table 4: Summary Statistics for Selection Operators

Selection operator	**Best**	**Worst**	**Average**
Elite	**838.42**	1214.52	994.50
Tournament	863.99	1054.96	943.10
Roulette	918.42	1183.41	1071.05
Rank	894.42	1149.44	968.58

The top performing configurations in Table 5 highlight the dominance of OBX crossover and inversion mutation, which appear in three and four of the five best combinations, respectively. The prevalence of tournament selection (four entries) alongside elite selection (one entry) further supports its role in maintaining population diversity. Notably, the winning configuration (OBX + inversion + elite) achieved a 6.9% deviation from the benchmarks optimal distance, demonstrating the potential of carefully tuned GAs for near-optimal CVRP solutions.

These results validate the critical role of operator selection in GA performance. The synergy between OBX crossover, inversion mutation, and tournament/elite selection emerges as a robust strategy for CVRP optimization.

Table 5: Top 5 Genetic Operator Combinations by Distance

Crossover	**Mutation**	**Selection**	**Distance**
OBX	Inversion	Elite	**838.42**
OBX	Swap	Tournament	863.99
CX	Inversion	Tournament	870.60
OBX	Inversion	Tournament	872.04
OX	Inversion	Tournament	903.68

6 Conclusion

The Vehicle Routing Problem (VRP) is a well-known problem extensively studied in the literature due to its wide-ranging practical applications. VRP is NP-hard. As a result, approximate methods, such as Genetic Algorithms (GAs), which are a key component of the Evolutionary Intelligence protfolio, offer a more viable solution approach for larger-scale problems.

Inspired from the evolution process of natural species, GAs can learn to adapts themselves by leveraging its operators capability to find and maintain good genetic information from generation to generation. In this paper, we reviewed various GA operators specifically designed for the VRP and its variants. The empirical evaluation of various operator configurations on the Capacitated VRP using a well-known benchmark set demonstrated that the combination OBX crossover, inversion mutation, and tournament selection achieve near-optimal results, deviating by only 6.9% from the benchmarks theoretical optimum.

As a prospective extension of this work, we recommend applying the tested operator configurations across a broader range of benchmark datasets and VRP variants. Such an extension would facilitate a comparative analysis aimed at assessing the operators' robustness and generalization ability with respect to solution quality. This comparative analysis could provide valuable insights into the learning strength of each operator for solving VRP variants effectively.

References

[1] P. Oberlin, S. Rathinam, and S. Darbha, "Today's traveling salesman problem," *Robotics & Automation Magazine*, vol. 17, pp. 70–77, 01 2011.

[2] S. Bansal and R. Goel, "Multi Objective Vehicle Routing Problem: A Survey," *Asian Journal of Computer Science and Technology*, vol. 7, no. 3, pp. 1–6, 2018.

[3] A. Verma, "Electric vehicle routing problem with time windows, recharging stations and battery swapping stations," *EURO Journal on Transportation and Logistics*, vol. 7, no. 4, pp. 415–451, 2018.

[4] C. Archetti, E. Fernndez, and D. L. Huerta-Muoz, "A two-phase solution algorithm for the flexible periodic vehicle routing problem," *Computers & Operations Research*, vol. 99, pp. 27–37, 2018.

[5] X. Ma and C. Liu, "Improved ant colony algorithm for the split delivery vehicle routing problem," *Applied Sciences*, vol. 14, no. 12, 2024.

[6] N. Nepomuceno, R. Barboza Saboia, and P. Rogrio Pinheiro, "A fast randomized algorithm for the heterogeneous vehicle routing problem with simultaneous pickup and delivery," *Algorithms*, vol. 12, no. 8, 2019.

[7] S. Dabia, S. Ropke, and T. van Woensel, "Cover inequalities for a vehicle routing problem with time windows and shifts," *Transportation Science*, vol. 53, no. 5, pp. 1354–1371, 2019.

[8] A.-K. Rothenbächer, "Branch-and-price-and-cut for the periodic vehicle routing problem with flexible schedule structures," *Transportation Science*, vol. 53, no. 3, pp. 850–866, 2019.

[9] W. Zhang, Y. Gajpal, S. S. Appadoo, and Q. Wei, "Multi-depot green vehicle routing problem to minimize carbon emissions," *Sustainability*, vol. 12, no. 8, 2020.

[10] R. Ben Jelloun, K. Jebari, and A. El Moujahid, "Open competency optimization: A human-inspired optimizer for the dynamic vehicle-routing problem," *Algorithms*, vol. 17, no. 10, 2024.

[11] F. Xie, Z. Chen, and Z. Zhang, "Research on dynamic takeout delivery vehicle routing problem under time-varying subdivision road network," *Mathematics*, vol. 12, no. 7, 2024.

[12] J. H. Holland, *Adaptation in Natural and Artificial Systems.* Ann Arbor, MI: University of Michigan Press, 1975.

[13] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* New York: Addison-Wesley, 1989.

[14] J. Ochelska-Mierzejewska, A. Poniszewska-Marada, and W. Marada, "Selected genetic algorithms for vehicle routing problem solving," *Electronics*, vol. 10, no. 24, 2021.

[15] M. Boulif and K. Atif, "A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem," *Computers & operations research*, vol. 33, no. 8, pp. 2219–2245, 2006.

[16] V. Chahar, S. Katoch, and S. Chauhan, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools and Applications*, vol. 80, 02 2021.

[17] T. Rajora, A. Gaur, T. Kapoor, A. Kushwaha, Y. Prashar, J. Parashar, D. Akhilesh, and Gupta, "Implementation of genetic algorithm on vehicle routing system," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, pp. 1405–1414, 12 2023.

[18] B. A. Julstrom, "Adaptive operator probabilities in a genetic algorithm that applies three operators," in *Proceedings of the 1997 ACM symposium on Applied computing*, pp. 233–238, 1997.

[19] A. Tuson and P. Ross, "Adapting operator settings in genetic algorithms," *Evolutionary computation*, vol. 6, no. 2, pp. 161–184, 1998.

[20] C.-H. Wang and J.-Z. Lu, "A hybrid genetic algorithm that optimizes capacitated vehicle routing problems," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2921–2936, 2009.

[21] L. Davis, "Applying adaptive algorithms to epistatic domains," in *Proceedings of the 9th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'85, (San Francisco, CA, USA), p. 162164, Morgan Kaufmann Publishers Inc., 1985.

[22] L. M. Hvattum, "Adjusting the order crossover operator for capacitated vehicle routing problems," *Computers & Operations Research*, vol. 148, p. 105986, 2022.

[23] I. M. Oliver, D. J. Smith, and J. R. C. Holland, "A study of permutation crossover operators on the traveling salesman problem," in *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application*, (USA), p. 224230, L. Erlbaum Associates Inc., 1987.

[24] G. N. Yücenur and N. Ç. Demirel, "A new geometric shape-based genetic clustering algorithm for the multi-depot vehicle routing problem," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11859–11865, 2011.

[25] K. Alabdulkareem and Z. H. Ahmed, "Comparison of four genetic crossover operators for solving distance-constrained vehicle routing problem," *IJCSNS International Journal of Computer Science and Network Security*, vol. 20, no. 7, pp. 114–123, 2020.

[26] G. Shanmugam, P. Ganesan, and P. T. Vanathi, "Meta heuristic algorithms for vehicle routing problem with stochastic demands," *Journal of Computer Science*, vol. 7, no. 4, p. 533, 2011.

[27] E. Tonbul, M. A. Takan, G. T. Büyükköse, and N. Erginel, "Modeling open vehicle routing problem with real life costs and solving via hybrid civilized genetic algorithm," *Sigma Journal of Engineering and Natural Sciences*, vol. 42, no. 3, pp. 714–730, 2024.

[28] G. Syswerda, "Schedule optimization using genetic algorithms," *Handbook of genetic algorithms*, 1991.

[29] P. Larranaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "Genetic algorithms for the travelling salesman problem: A review of representations and operators," *Artificial intelligence review*, vol. 13, pp. 129–170, 1999.

[30] A. Bolotbekova, H. Hakli, and A. Beskirli, "Trip route optimization based on bus transit using genetic algorithm with different crossover techniques: a case study in konya/türkiye," *Scientific Reports*, vol. 15, no. 1, p. 2491, 2025.

[31] W. Banzhaf, "The molecular traveling salesman," *Biological Cybernetics*, vol. 64, no. 1, pp. 7–14, 1990.

[32] Z. H. Ahmed, N. Al-Otaibi, A. Al-Tameem, and A. K. J. Saudagar, "Genetic crossover operators for the capacitated vehicle routing problem.," *Computers, Materials & Continua*, vol. 75, no. 1, 2023.

[33] D. B. Fogel, "An evolutionary approach to the traveling salesman problem," *Biological Cybernetics*, vol. 60, no. 2, pp. 139–144, 1988.

[34] M. N. Mageswari, "Vehicle routing problem (vrp) using genetic algorithm," *Science*, vol. 9, no. 3, pp. 1–3, 2024.

[35] D. B. Fogel, "Applying evolutionary programming to selected traveling salesman problems," *Cybernetics and systems*, vol. 24, no. 1, pp. 27–36, 1993.

[36] P. Augerat, D. Naddef, J. Belenguer, E. Benavent, A. Corberan, and G. Rinaldi, "Computational results with a branch and cut code for the capacitated vehicle routing problem," 1995.

From Ants to People: the Vaporization of Social Relationships in Dynamic Community Detection

Rachid Djerbi[1] **and Mohamed Tahar Bennai**[1]

[1] ***Department of Computer Science, Faculty of Sciences, University M'Hamed Bougara of Boumerdes,*** {r.djerbi, m.bennai}@univ-boumerdes.dz

Abstract

Community detection in social networks is crucial for understanding social dynamics and interactions. In this paper, we propose LFM2ACO, an algorithm designed to detect dynamic communities by combining the principles of the static Large Families Model (LFM) with those of Ant Colony Optimization (ACO). Inspired by the biological phenomenon of ant colonies, where pheromones guide and reinforce paths to food sources, we model social relationships as dynamic trails that require constant renewal. Just as pheromones evaporate over time unless refreshed by ant activity, social connections weaken without continued interaction. The LFM2ACO algorithm captures this essence, simulating the strengthening of relationships through repeated communication (e.g., messages, likes, comments) and their decay in the absence of such interactions. Comprehensive experiments on real-world social networks, including the Facebook Wall/Links dataset and the Enron email dataset, demonstrate the robustness and efficacy of LFM2ACO in accurately detecting dynamic communities. This work not only enhances the understanding of community evolution but also provides a practical, implemented solution, validated through experimentation, offering valuable insights for future research and development in community detection algorithms.

Keywords: Community detection; LFM; Dynamic communities; Ant Colony Optimization (ACO); Social networks; Pheromone; Community evolution.

1 Introduction

1.1 Background

Social networks are complex environments modeled by graphs [1]. Analyzing these networks allows the extraction of hidden characteristics, such as community detection [2].

1.2 Motivation

Community detection has received considerable attention, allowing a macroscopic view of network structure. Social networks are dynamic, requiring consideration of their evolution. The LFM algorithm [3] is effective for static communities but doesn't support dynamic graphs [1].

1.3 Objectives

This study extends the LFM algorithm for dynamic networks using ACO principles. The main objective is to hybridize LFM and ACO for dynamic community detection. Specifically, we aim to:

- Study and implement the LFM algorithm for static community detection.
- Propose a dynamic extension of LFM based on ACO principles (LFM2ACO).
- Evaluate the performance of LFM2ACO in real-world datasets.

Section 2 provides related work, Section 3 outlines the methodology, Section 4 presents the experimental setup, Section 5 presents results and analysis, and Section 6 concludes the work.

2 Related Work

2.1 Overview

Community detection algorithms identify groups of densely connected nodes [4]. Approaches include graph partitioning [5] and hierarchical clustering [6]. Modularity-based methods, like the Louvain algorithm, optimize the modularity score [9].

2.2 Static vs. Dynamic

Algorithms are classified into static [7] and dynamic approaches [10]. Dynamic algorithms account for the temporal evolution of networks [8].

2.3 Hybrid Approaches

Hybrid approaches combine different techniques. There is growing interest in combining metaheuristic algorithms, such as ACO, with traditional algorithms [13, 14]. LFM is effective for static communities [3]. This work addresses this limitation by combining LFM with ACO principles.

3 Methodology

This section details the hybridization of LFM with ACO for dynamic community detection.

3.1 Large Families Model (LFM) Algorithm

The LFM algorithm [3] identifies communities based on "large families." The algorithm operates in three main steps:

1. Initial Community Detection
2. Out-Node Integration
3. Community Merging

The LFM algorithm uses notations shown in Table 1.

Table 1: Notations used in the LFM approach [3].

Notation	Description
V	Set of nodes
E	Set of edges of the graph
G=(V, E)	The graph associated with the social network
A	Adjacency matrix of G
A[i, j]	Boolean value representing the relationship between nodes i and j
MRC	Set of maximum connected components
MRC(l)	The lth MRC

The LFM algorithm maximizes the modularity of the resulting community structure but doesn't account for the temporal evolution of social networks.

3.2 LFM2ACO: Dynamic Extension of LFM using ACO Principles

To address the limitations of LFM, we propose LFM2ACO, a dynamic extension of LFM based on ACO principles. The key idea is to use ACO to model the temporal evolution of relationships.

Figure 1 illustrates the LFM2ACO approach.

The LFM2ACO approach consists of the following steps:

1. Data Preprocessing
2. Pheromone Initialization

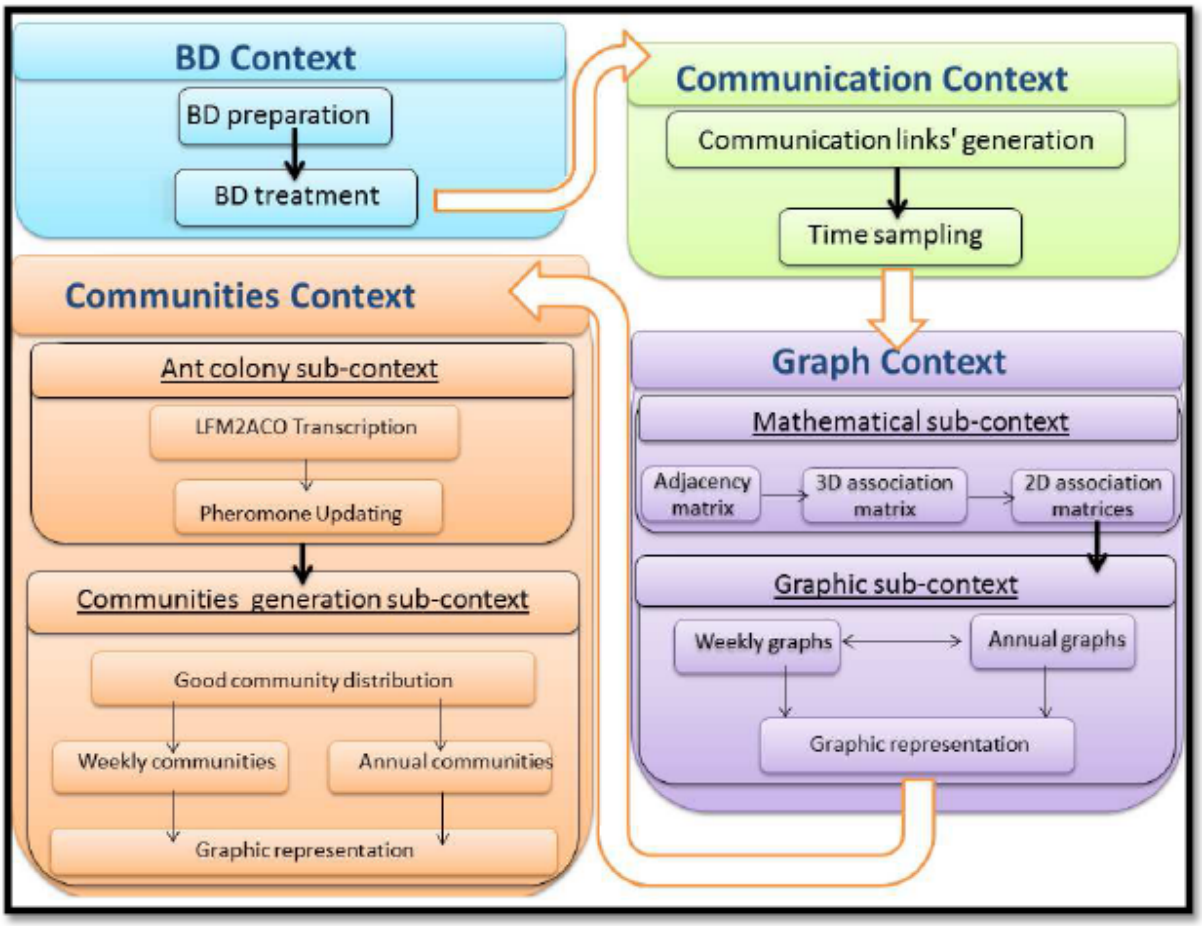


Figure 1: LFM2ACO Approach

3. Ant Colony Optimization
4. Dynamic Community Detection
5. Community Evolution Analysis

Figure 2 shows an example of how the dynamic network can be represented mathematically and as a 3D matrix.

The pheromone update rule is a crucial component:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t)$$

where:

* ij(t) is the pheromone value on edge (i, j) at time t * is the pheromone evaporation rate * m is the number of ants * kij(t) is the amount of pheromone deposited by ant k on edge (i, j) at time t

Table **??** shows the transcription of the LFM model into ACO terminology.

4 Experimental Setup

This section describes the experimental setup used to evaluate the performance of the LFM2ACO approach.

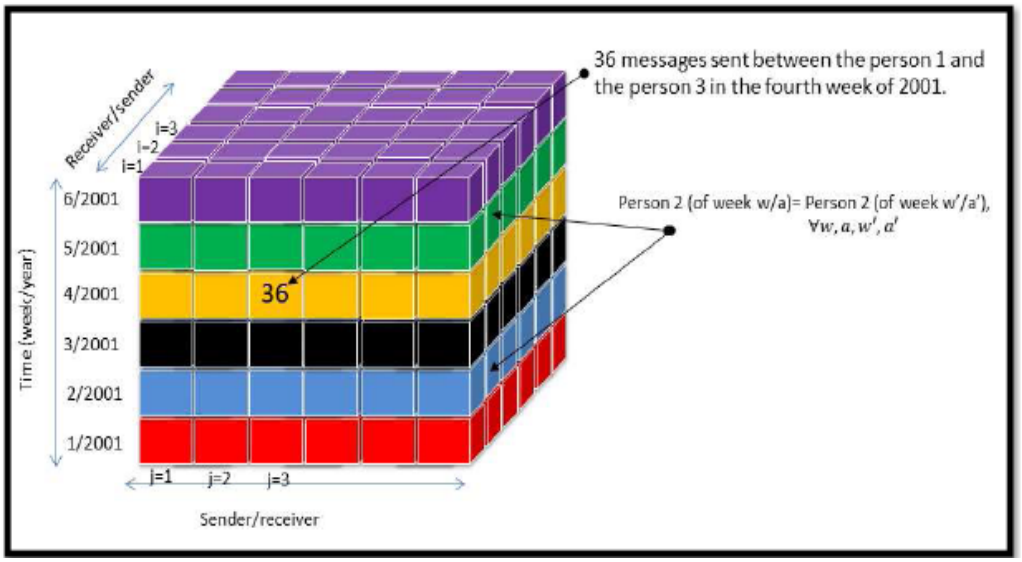


Figure 2: LFM2ACO, concept 3D.

Table 2: Transcription ACO of the LFM model.

LFM Model	**ACO Terminology**	**Description**
Social network	Set of ants, nests, and food sources	Represents the overall social environment
Two employees communicating	Two adjacent vertices contacted	Represents interaction between individuals
Week (in our case study)	Evaporation of a quantity (Beta) of pheromone	Represents the decay of social relationships over time
Number of days without contact	Number of days of absence	Represents the duration of inactivity between individuals

4.1 Dataset Description

We used the Facebook Wall/Links dataset [15, 16] and the Enron Email dataset [17].

4.1.1 Facebook Wall/Links Dataset

The Facebook Wall/Links dataset contains user-to-user links and user posts.

4.1.2 Enron Email Dataset

The Enron Email dataset is a database of emails from the Enron Corporation. We used a subset of the Enron Email dataset consisting of messages exchanged between employees.

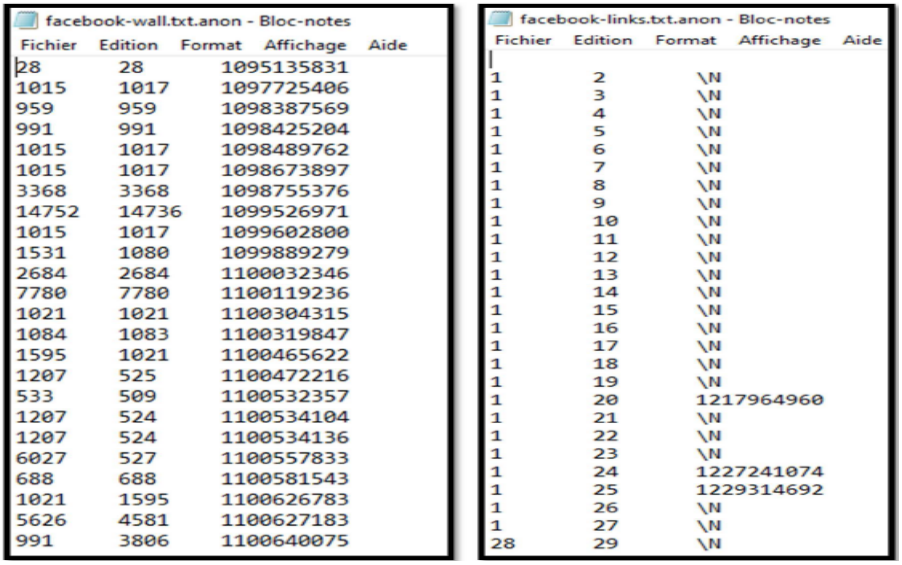
Figures 3 and 4 illustrate the datasets.

4.2 Preprocessing and Data Preparation

The preprocessing and data preparation steps varied depending on the dataset. For the Enron Email dataset, we performed preprocessing steps to construct a dynamic social network.

4.3 Implementation Environment and Tools

Our work was performed on an INTEL CORE™i5 processor with 4GB of memory and a 64-bit Windows operating system. We used text files, a MySQL database, Uwamp, and PHP.



facebook-wall.txt.anon - Bloc-notes

Fichier Edition Format Affichage Aide

28	28	1095135831
1015	1017	1097725406
959	959	1098387569
991	991	1098425204
1015	1017	1098489762
1015	1017	1098673897
3368	3368	1098755376
14752	14736	1099526971
1015	1017	1099602800
1531	1080	1099889279
2684	2684	1100032346
7780	7780	1100119236
1021	1021	1100304315
1084	1083	1100319847
1595	1021	1100465622
1207	525	1100472216
533	509	1100532357
1207	524	1100534104
1207	524	1100534136
6027	527	1100557833
688	688	1100581543
1021	1595	1100626783
5626	4581	1100627183
991	3806	1100640075

facebook-links.txt.anon - Bloc-notes

Fichier Edition Format Affichage Aide

1	2	\N
1	3	\N
1	4	\N
1	5	\N
1	6	\N
1	7	\N
1	8	\N
1	9	\N
1	10	\N
1	11	\N
1	12	\N
1	13	\N
1	14	\N
1	15	\N
1	16	\N
1	17	\N
1	18	\N
1	19	\N
1	20	1217964960
1	21	\N
1	22	\N
1	23	\N
1	24	1227241074
1	25	1229314692
1	26	\N
1	27	\N
28	29	\N

Figure 3: Benchmarks used: Facebook Wall on the left and Facebook links on the right.

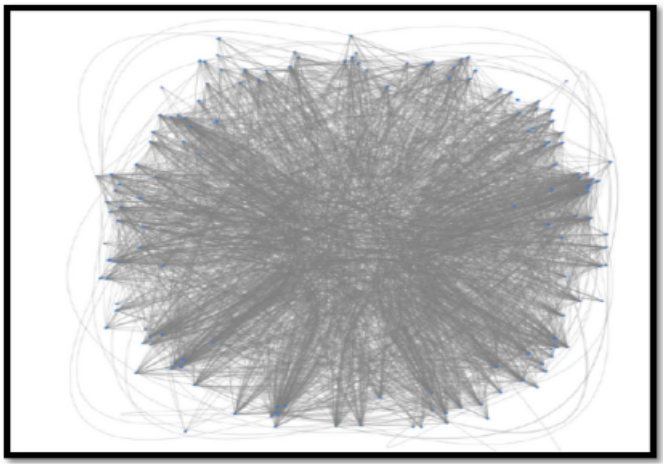


Figure 4: ENRON network before any community detection algorithm was applied.

5 Results and Analysis

This section presents and analyzes the results obtained from applying the static LFM algorithm to the Facebook dataset and the dynamic LFM2ACO algorithm to the Enron dataset.

5.1 Static LFM Results on Facebook Dataset

The static LFM algorithm was applied to the Facebook Wall/Links dataset. Table 3 summarizes the results.

Figure 5 shows the final community structure obtained for ‘CadjMax‘ = 2.

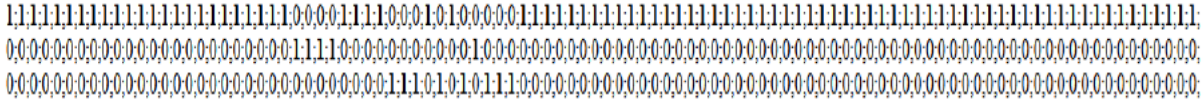


Figure 5: Final distribution result comment for cadjmax=2.

Table 3: Summaries of community distributions of each iteration (CadjMax–) with modularity.

CadjMax	First generation	Latest generation	Number of out nodes	Modularity(Q)
9	40	1	87	5.9799773157978E-17
8	62	1	88	5.9799773157978E-17
7	76	1	89	5.9799773157978E-17
6	102	1	90	5.9799773157978E-17
5	147	2	83	0.17051866319444
4	207	2	83	0.17051866319444
3	283	2	85	0.15957151813272
2	417	3	83	0.15970413773148

5.2 Dynamic LFM2ACO Results on Enron Dataset

The dynamic LFM2ACO algorithm was applied to the Enron Email dataset. The algorithm was executed iteratively, updating the pheromone matrix.

Figures 6, 7, 8, and 9 illustrate the pheromone update process.

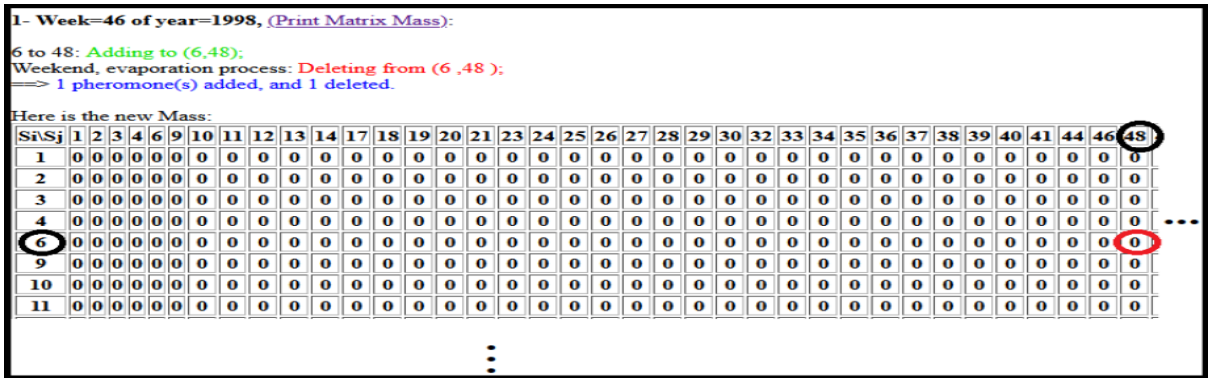


Figure 6: Result of the pheromone quantity update, week=46 and year 1998

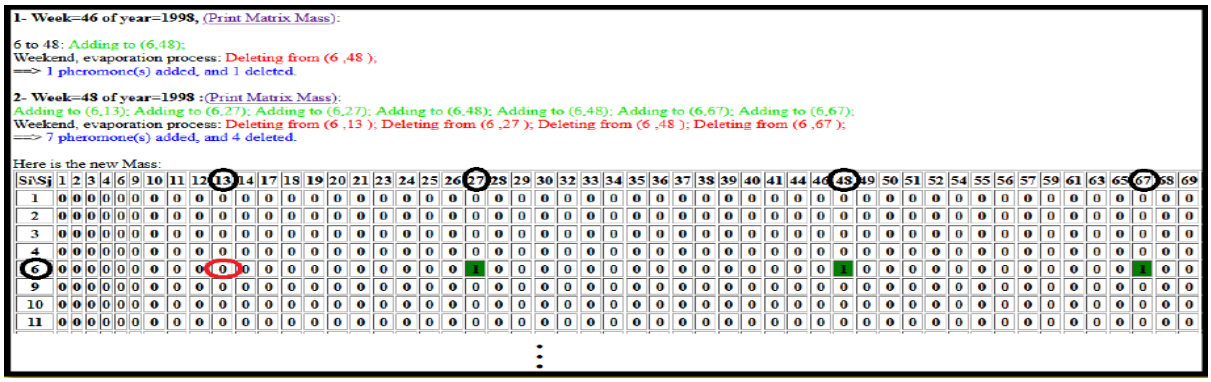


Figure 7: Result of the pheromone quantity update, week=48 and year 1998

Table 4 summarizes the results obtained for week 48 of 1998.

Figure 10 shows the community structure obtained for 'CadjMax' = 472.

5.3 Comparison

The static LFM algorithm achieved a maximum modularity of 0.17051866319444 on the Facebook dataset, while the dynamic LFM2ACO algorithm achieved a maximum modularity of 0.53652645659928 on the Enron dataset.

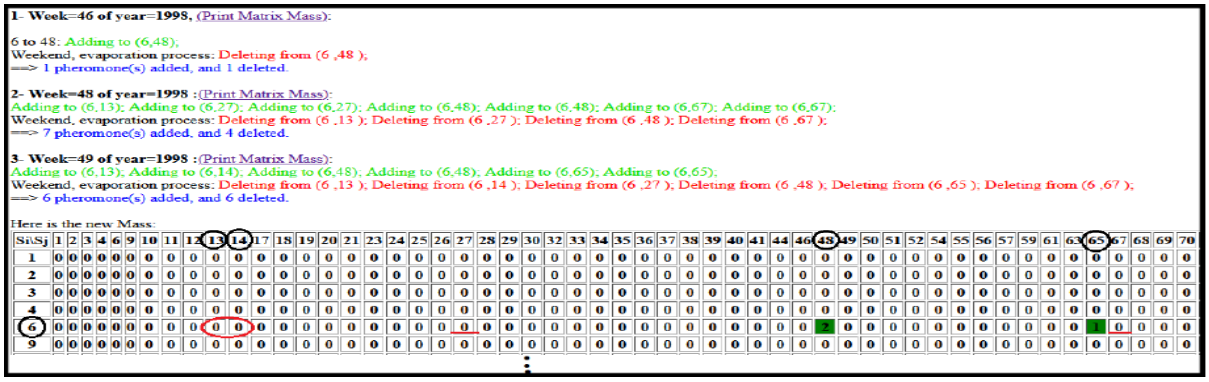


Figure 8: Result of the pheromone quantity update, week=49 and year 1998.

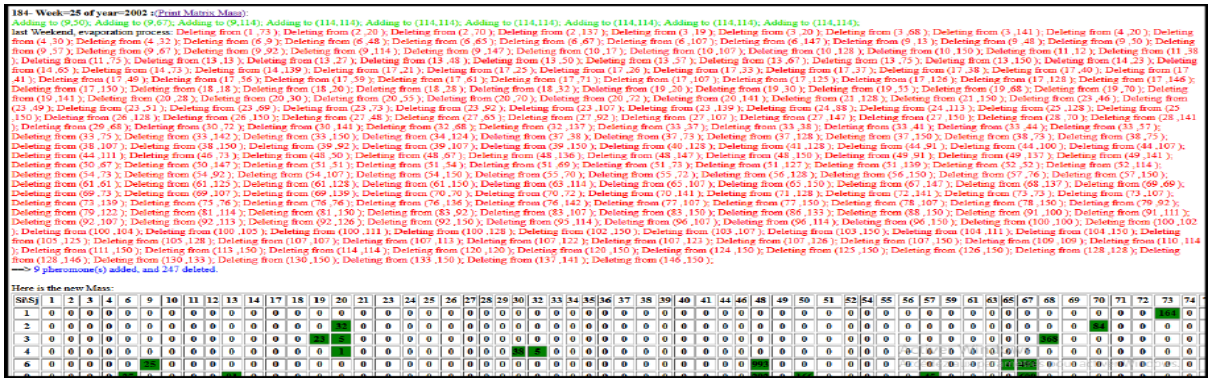


Figure 9: Result of the pheromone quantity update, week=25 and year 2002.

5.4 Discussion

The results provide insights into the dynamics of social networks. The dynamic LFM2ACO algorithm captures the community structure in dynamic networks.

6 Conclusion and Future Work

This section summarizes the contributions, discusses limitations, and suggests directions for future research. This paper has explored community detection in social networks. The main contributions include the following:

- A comprehensive review of existing community detection approaches.
- An in-depth study of the LFM algorithm.
- The development of LFM2ACO.
- An adaptation of the LFM algorithm to handle weighted networks (that we called WLFM for "Weighted LFM").
- Implementation and evaluation of LFM and LFM2ACO on real-world datasets.

This paper has some limitations, like:

- The LFM2ACO algorithm was only evaluated on two datasets.
- The LFM2ACO algorithm does not explicitly handle overlapping communities.
- Scalability to very large networks.

Based on these limitations, future research can be identified like:

Table 4: Summaries of community distributions of each CadjMAx- iteration - with the modularity of the WLFM algorithm.

CadjMax	Nombre de communauties	Modularity (Q)
472	10	0.53652645659928

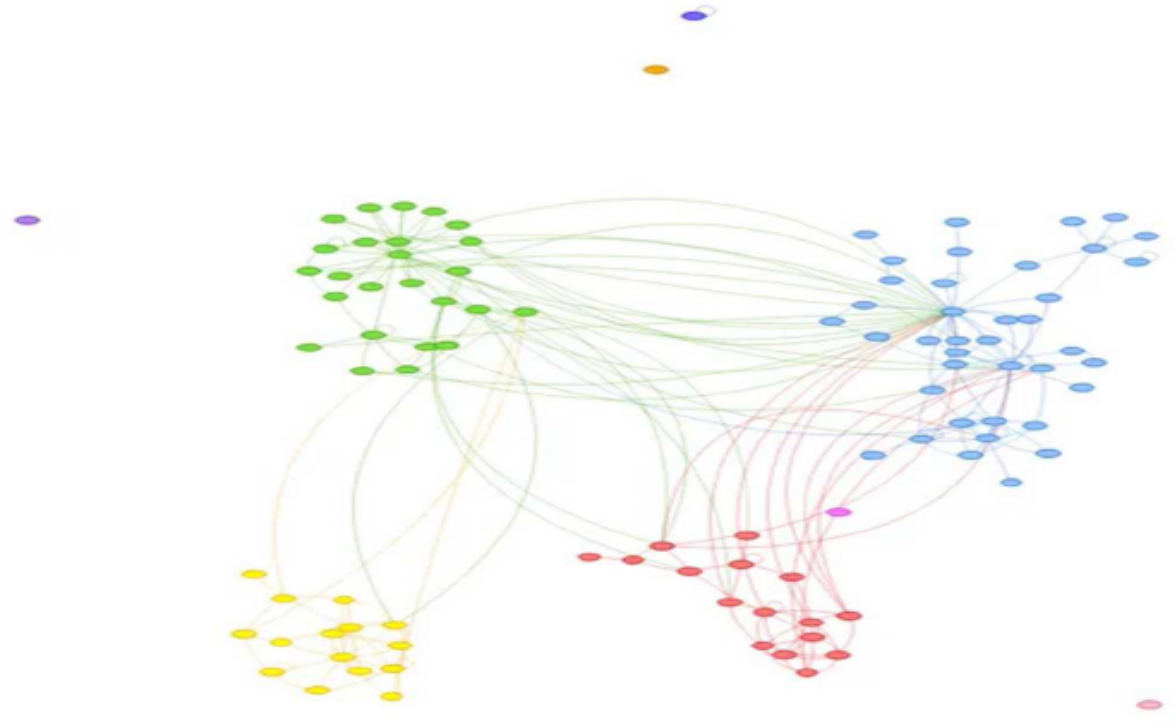


Figure 10: Graph result for the best distribution CadjMax=472.

- Extend the LFM2ACO algorithm to handle overlapping communities.
- Improve the scalability of the LFM2ACO algorithm.
- Apply the LFM2ACO approach to directed graphs.
- Evaluate the performance of the LFM2ACO algorithm on a wider range of datasets.

Another AI perspective regarding the use of AI to our original LFM algorithm or the one proposed in this work (LFM2ACO) such as:

- **Integrating Deep Learning for Predictive Community Dynamics**: Leverage temporal graph neural networks (TGNNs) or transformer-based architectures to model the evolution of social interactions, enabling the prediction of future community structures (e.g., births, mergers, or splits) based on historical trajectory patterns and individual behavior embeddings.
- **Behavior-Aware Forecasting with Reinforcement Learning**: Develop hybrid models combining LFM2ACO with deep reinforcement learning (DRL) to simulate adaptive agent behaviors, where AI-driven individuals dynamically switch communities based on learned reward mechanisms reflecting social preferences.
- **LLM-Enhanced Relationship Semantics**: Utilize large language models (LLMs) to analyze textual interaction data (e.g., social media content), extracting semantic signals to enrich edge weighting in WLFM and predict community formation triggers from latent topic shifts.
- **Neural Attention for Overlap Resolution**: Implement multi-head attention mechanisms to detect overlapping community boundaries by learning node-community affiliation probabilities, complementing ACO's pheromone dynamics with neural interpretability.

- **Graph Generation for Scenario Projection**: Train generative adversarial networks (GANs) or diffusion models on temporal network snapshots to synthesize plausible future graph states, enabling stress-testing of LFM2ACO under predicted social configurations.
- **Embedding-Driven Scalability**: Combine hyperbolic graph embeddings with LFM2ACO's optimization process to reduce computational complexity in large-scale networks while preserving hierarchical community structures.
- **Multimodal Fusion for Event Prediction**: Architect multimodal pipelines that jointly process network topology (via GNNs), temporal activity sequences (via LSTMs), and user metadata to forecast macro-level community events like mass migrations or influencer-driven splits.

References

[1] Stanley Wasserman, Katherine Faust, Stanley (University of Illinois Wasserman, Urbana-Champaign) Social Network Analysis: Methods and Applications, Volume 8 de Structural Analysis in the Social Sciences, ISSN 0954-366X, editeur:Cambridge University Press 1994,825 pages.

[2] C. Dawson and C. Dawson, "Social network analysis," A–Z Digit. Res. Methods, pp. 356–361, 2019, doi: 10.4324/9781351044677-54.

[3] Djerbi, R., Amad, M., & Imache, R. (2020). A new model for communities' detection in dynamic social networks inspired from human families. International Journal of Internet Technology and Secured Transactions, 10(1-2), 24-60.

[4] S. Fortunato, "Community detection in graphs," Phys. Rep., vol. 486, no. 3–5, pp. 75– 174, 2010, doi: 10.1016/j.physrep.2009.11.002.

[5] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," vol. 99, no. 12, 2002.

[6] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," Phys. Rev. E - Stat. Nonlinear,

[7] M.NEDIOUI, M´emoire fouille de donn´ee et apprentissage automatique dans les r´eseaux sociaux dynamiques, 2015.

[8] NEDIOUI, MED ABDELHAMID. Fouille et apprentissage automatique dans les reseaux sociaux dynamique. 2015. Th'ese de doctorat. Universit´e Mohamed Khider-Biskra, Alg´erie.

[9] Blondel, 2008, V.D. Blondel, J.L. Guillaume, R. Lambiotte et E. Lefebvre. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, vol. 2008, page P10008, 2008.

[10] Palla,2007G. Palla, A.L. Barabasi, and T. Vicsek. Quantifying social group evolution. Nature, 446(7136) :664667, 2007.

[11] Aynaud T., Fleury E., Guillaume J.-L., Wang Q. (2013). Communities in evolving networks: definitions, detection, and analysis techniques. In Dynamics on and of complex networks, volume 2, p. 159–200. Springer.

[12] Z. Chen, K. a. Wilson, Y. Jin, W. Hendrix, and N. F. Samatova. Detecting and Tracking Community Dynamics in Evolutionary Networks. 2010 IEEE International Conference on Data Mining Workshops, pages 318–327, Dec. 2010.

[13] Dorigo Gambardella, 1997] Dorigo, M., & Gambardella, L.M. 1997. Ant Colony System: A Cooperative Learning Approach to the Traveling SalesmanProblem. IEEE Transactions on Evolutionary Computation,1(1), 53 66.

[14] H.BELLEILI ,2020 Ant ColonyOptimization (ACO) optimisation par colonies de fourmis

[15] The Facebook Wall dataset: http://socialnetworks.mpi-sws.mpg.de/data/facebook-wall.txt.gz, Last accessed 24 Mars 2025

[16] The Facebook Links dataset http://socialnetworks.mpi-sws.mpg.de/data/facebook-links.txt.gz, Last accessed 24 Mars 2025

[17] Shetty, J., & Adibi, J. (2005, August). Discovering important nodes through graph entropy the case of enron email database. In Proceedings of the 3rd international workshop on Link discovery (pp. 74-81).

The Second National Conference on Applications of Artificial Intelligence (A2I-25) brings together researchers, professionals, and students to explore innovative applications of Artificial Intelligence that address real-world challenges in fields such as healthcare, agriculture, energy, cybersecurity, and urban development. Hosted by the University M'Hamed Bougara of Boumerdes (UMBB), this event fosters interdisciplinary collaboration and highlights the transformative power of AI in improving quality of life and societal well-being.

The conference also features an exclusive NVIDIA-certified training on Fundamentals of Deep Learning, led by Dr. Tayeb Benzenati, providing participants with hands-on experience in neural networks, optimization, and real-world AI applications.

Dates: April 16–17, 2025

Venue: Department of Computer Science, Faculty of Sciences, University of M'hamed Bougara UMBB, Boumerdes, Algeria.

9 789969 965001