Scientific Journal of Computer Science
futuristech PT. Teknologi Futuristik Indonesia
Received: March 29, 2026; Revised: May 17, 2026; Accepted: June 19, 2026; Published: August 9, 2026; Current version: August 9, 2026. e-ISSN: 3110-3170
Digital Object Identifier 10.64539/sjcs.v2i2.2026.475

Article

Enhancing Binary Image Classification Accuracy Using Low-Rank Adaptation (LoRA) for Deepfake Detection

Tam Thanh Thi Pham[1,2], Thao Thanh Thi Nguyen[1], Thai Hoang Le[3,4], Hai Son Tran[1,*]

[1] Department of Information Technology, Ho Chi Minh City University of Education, Ho Chi Minh City, 700000, Vietnam; e-mail: haits@hcmue.edu.vn (H. S. Tran).
[2] Testing and Assessment Department, Hong Bang International University, Ho Chi Minh City, 700000, Vietnam.
[3] Department of Information Technology, University of Science, Ho Chi Minh City, 700000, Vietnam.
[4] Department of Information Technology, Vietnam National University, Ho Chi Minh City, 700000, Vietnam.
* Correspondence

The authors declare that no funding, grants, or other financial support was received from any organization for the submitted work.

Abstract: Deepfake technology poses an increasingly serious threat to personal reputation and social trust, necessitating the development of accurate yet computationally efficient detection systems. Although large pre-trained vision models offer exceptional feature extraction capabilities, full fine-tuning them demands prohibitive computational resources and risks overfitting. This study investigates the application of Low-Rank Adaptation (LoRA) to enhance binary image classification accuracy for deepfake face detection, bridging the gap between parameter efficiency and high classification performance. We systematically integrate LoRA into two dominant architectural paradigms: the Vision Transformer (Swin-T) and ResNet-50. Computational evaluations are conducted on a 40K sub-dataset from the 140K Real and Fake Faces dataset, comparing LoRA against full fine-tuning baselines under identical environments. Experimental results demonstrate that Swin-T + LoRA achieves an outstanding test accuracy of 99.14% and an F1-score of 0.9913, outperforming its full fine-tuning baseline by 9.95 percentage points while training only 5.88% of the total parameters. Conversely, ResNet-50 + LoRA improves test accuracy by 14.11 percentage points over its full fine-tuning baseline, although its performance remains substantially below that of Swin-T + LoRA, indicating that LoRA effectiveness varies across architectural paradigms. These findings demonstrate that parameter-efficient fine-tuning, particularly when combined with Transformer attention layers, offers a promising approach for developing accurate and computationally efficient deepfake detection systems under resource constraints.

Keywords: Low-Rank Adaptation; LoRA; Deepfake Detection; Vision Transformer; ResNet-50; Binary Image Classification; Parameter-Efficient Fine-Tuning; Transfer Learning.

1. Introduction

Deepfake technology has emerged as a critical societal concern due to its capacity to generate highly realistic synthetic images and videos, which can severely damage personal reputations and undermine public trust. The World Health Organization (WHO) and numerous international digital safety organizations have highlighted the growing sophistication of AI-generated content and weaponized infodemics, making it increasingly difficult to distinguish real from fabricated media[1].

In the domain of computer vision, deep learning architectures such as Vision Transformer (Swin-T) [1] and ResNet-50 [2] have demonstrated exceptional performance in image classification tasks. Swin-T processes

[1] The World Health Organization (WHO) and numerous international digital safety organizations have highlighted the growing sophistication of AI-generated content and weaponized infodemics, making it increasingly difficult to distinguish real from fabricated media https://www.who.int/news/item/18-01-2024-who-releases-guidance-on-ai-for-health.

images hierarchically using window-based self-attention and shifted windows, enabling efficient modeling of both local and cross-window contextual relationships. ResNet-50, on the other hand, employs residual connections to train very deep convolutional networks effectively by mitigating the vanishing gradient problem. Both architectures, when pre-trained on large-scale datasets like ImageNet, serve as powerful feature extractors for downstream tasks.

However, fine-tuning these large pre-trained models for specific tasks such as deepfake detection typically requires substantial computational resources, memory, and training time. For instance, Swin-T contains approximately 28.3 million parameters, while ResNet-50 has 23.5 million. Updating all parameters during full fine-tuning is not only computationally expensive but also risks overfitting when the target dataset is limited in size or diversity.

Low-Rank Adaptation (LoRA) [3] was proposed as an efficient solution to this challenge. Based on the hypothesis that weight updates during fine-tuning can be approximated by low-rank matrices, LoRA freezes the original model weights and introduces small trainable matrices that capture task-specific adaptations. This approach dramatically reduces the number of trainable parameters while maintaining or even improving model performance. Aghajanyan *et al*. [4] provided theoretical support for this approach by demonstrating that pre-trained language models have a low intrinsic dimensionality, meaning that effective fine-tuning can occur in a much smaller parameter subspace than the full model. Building on this foundation, several LoRA variants have been proposed: QLoRA [5] combines LoRA with 4-bit quantization for even greater memory savings, while AdaLoRA [6] dynamically allocates the rank budget across weight matrices based on their importance. While LoRA has been extensively studied in Natural Language Processing (NLP) [3], its application to computer vision tasks, particularly deepfake detection, remains underexplored.

It is worth noting a common architectural characteristic of LoRA: while it freezes the backbone parameters to minimize trainable weights, the introduction of the adapter matrices (A and B) slightly increases the total footprint of the model architecture during the active training phase. For instance, the total parameter count of Swin-T expands from 28.3M to 30.1M (see Table 1). Furthermore, to clarify a technical inconsistency regarding the ResNet-50 architecture: the standard baseline backbone comprises 23.5M parameters, but with the inclusion of the final modified classification head layers formatted for this specific dataset, the total operational parameter footprint rises to 25.6M.

This study addresses this gap by systematically evaluating LoRA on two representative architectures—Swin-T and ResNet-50—for binary classification of real versus fake face images. The main contributions of this paper are as follows:

1) A comprehensive experimental evaluation of LoRA applied to both Transformer-based and CNN-based vision models for deepfake detection;
2) A detailed comparative analysis of LoRA versus full fine-tuning in terms of accuracy, F1-score, parameter count, and training time;
3) Practical insights into the effectiveness and limitations of LoRA across different architectural paradigms.

The remainder of this paper is organized as follows. Section 2 reviews related work on deepfake detection and parameter-efficient fine-tuning. Section 3 describes the methodology, including dataset preparation, model architectures, and LoRA integration. Section 4 presents experimental results and discussion. Section 5 concludes with a summary of findings and future research directions.

2. Related Work

2.1. Deep Learning for Image Classification

Deep learning has revolutionized computer vision, with Convolutional Neural Networks (CNNs) serving as the foundational architecture for image classification tasks [7]. Architectures such as VGGNet [8], AlexNet [9], and ResNet-50 [2] have established core principles of hierarchical feature extraction. ResNet, in particular, introduced residual connections that enable training of networks with over 100 layers by addressing the vanishing gradient problem. The Bottleneck Residual Block design reduces computational cost by approximately 75% compared to standard 3×3 convolution blocks while maintaining strong representational capacity.

More recently, Vision Transformers (Swin-T) [1] have emerged as a powerful alternative to CNNs. By dividing images into fixed-size patches and processing them through self-attention mechanisms, Swin-T captures global contextual relationships that CNNs, with their local receptive fields, may miss. Swin-T, pre-trained on ImageNet-1k, has demonstrated competitive or superior performance across numerous vision benchmarks.

2.2. Deepfake Detection

Deepfake detection has attracted significant research attention in recent years. Traditional approaches rely on hand-crafted features such as facial landmarks, texture inconsistencies, and frequency-domain analysis [10]. Early methods employed Support Vector Machines and Random Forests with manually engineered features, achieving moderate success on controlled datasets. However, these approaches suffer from poor generalizability when confronted with increasingly sophisticated generation techniques.

Modern methods predominantly employ deep learning, using CNNs and Transformers to automatically learn discriminative features from face images [11]. Cozzolino *et al.* [12] introduced the FaceForensics++ benchmark, which has become a standard evaluation framework for face manipulation detection. Li *et al*. [13] proposed Face X-ray, a method for detecting face boundaries introduced by the blending process in face manipulation, achieving generalization across different forgery methods. The Celeb-DF dataset [14] further challenged existing methods by providing high-quality deepfake videos that are more difficult to detect.

The evolution of Generative Adversarial Networks (GANs), particularly StyleGAN [15], has made deepfake generation increasingly realistic, further motivating the need for robust detection systems. Transfer learning from large-scale pre-trained models has emerged as a dominant strategy, with researchers fine-tuning ImageNet-pre-trained networks for deepfake classification [16]. Tolosana *et al*. [17] provided a comprehensive survey of face manipulation and detection methods, highlighting the ongoing arms race between generation and detection technologies. The availability of large-scale datasets such as FaceForensics++ and the 140K Real and Fake Faces dataset [18] has facilitated the training and evaluation of detection models. However, achieving high accuracy while maintaining computational efficiency remains a key challenge.

2.3. Parameter-Efficient Fine-Tuning and LoRA

Rather than updating all parameters of a pre-trained network, parameter-efficient fine-tuning (PEFT) techniques introduce a limited set of trainable parameters for adapting the model to a downstream task. Low-Rank Adaptation (LoRA) [3] implements this strategy by representing the weight update ΔW as the product of two low-rank matrices, formulated as ΔW=BA. Here, $B \in R^{d\times r}$ and $A \in R^{r\times k}$, where $r \ll \min(d, k)$. During fine-tuning, the original weight matrix W0 remains frozen, while only the newly introduced matrices A and B are optimized.

The forward pass becomes $h = W_0x + BAx$. LoRA was originally demonstrated on GPT-3, reducing VRAM consumption from 1.2TB to 350GB while maintaining comparable performance [3]. Bafghi *et al*. [19] explored LoRA for self-supervised ViTs, demonstrating its potential in vision tasks. However, systematic evaluations of LoRA for deepfake detection across both Transformer and CNN architectures remain limited.

3. Methodology

3.1. Dataset Description

This study employs a 40K sub-dataset sampled from the 140K Real and Fake Faces dataset [18]. The original dataset consists of 70,000 real face images sourced from Flickr-Faces-HQ (FFHQ) by NVIDIA and 70,000 fake face images generated by StyleGAN. All images are standardized to 256×256 pixels. From this source, a balanced 40,000-image subset was constructed and partitioned into training (20,000 images), validation (10,000 images), and test (10,000 images) sets, each maintaining equal representation of real and fake classes. To optimize training on resource-constrained hardware, the training set of 20,000 images is processed in mini-batches with gradient accumulation to reduce peak memory consumption and enable checkpoint-based recovery.

The 140K Real and Fake Faces dataset is publicly accessible via Kaggle [18]. To ensure rigorous evaluation and prevent any potential data leakage or shortcut exploitation, a strict zero-overlap policy was enforced during partitioning: faces of specific identities or generation groups present in the training set (20,000 images) do not appear inside the validation set (10,000 images) or the test set (10,000 images). Table 2 provides the structural classification breakdown of the dataset.

3.2. Model Architectures

3.2.1. Swin-T

The Swin-T model [1] is employed as the Transformer-based architecture and initialized with ImageNet-pre-trained weights. Swin-T uses a hierarchical architecture with shifted-window self-attention [1], [20]. For a 224 × 224 input image, the image is partitioned into non-overlapping 4 × 4 patches and processed through four hierarchical stages with 2, 2, 6, and 2 Swin Transformer blocks, respectively [21]. Patch merging between stages progressively reduces the spatial resolution. Self-attention is computed within non-overlapping local windows, while shifted windows in consecutive blocks enable cross-window information exchange [1].

3.2.2. ResNet-50 v1.5

The ResNet-50 v1.5 model from Microsoft, pre-trained on ImageNet-1k [22], is used. The architecture comprises a stem layer (Conv2d 7×7, BatchNorm, ReLU, MaxPool) followed by four stages containing 3, 4, 6, and 3 Bottleneck Residual Blocks respectively, producing feature maps of 256, 512, 1024, and 2048 channels. Global average pooling followed by a linear layer performs binary classification. Total parameters: 23.5 million [23]-[25].

3.3. LoRA Integration

The core principle of LoRA is to decompose the weight update during fine-tuning into a low-rank approximation. For a pre-trained weight matrix W_0, the adapted weight is expressed as $W = W_0 + \Delta W = W_0 + BA$, where $B \in R^{d\times r}$ and $A \in R^{r\times k}$ are two small trainable matrices, and $r \ll \min(d, k)$ is the chosen rank. During forward propagation, the output is computed as $h = W_0x + BAx$. Matrix A is initialized with a Gaussian distribution and B with zeros, ensuring

Table 1. Comparison of Trainable Parameters.

Model	Total Params	Trainable Params	Ratio (%)
Swin-T (full)	28.3M	28.3M	100.00
Swin-T + LoRA	30.1M	1.77M	5.88
ResNet-50 (full)	25.6M	25.6M	100.00
ResNet-50 + LoRA	25.6M	1.77M	6.93

Table 2. Statistical Breakdown of the 40K Face Sub-Dataset.

Class Label	Training Set	Validation Set	Test Set
Real (FFHQ)	10,000	5,000	5,000
Fake (StyleGAN)	10,000	5,000	5,000
Total	20,000	10,000	10,000

$\Delta W = BA = 0$ at the start of training. The output from BAx is scaled by a factor α/r, where α is a hyperparameter analogous to the learning rate, which reduces the dependency on hyperparameter tuning when the rank r is varied [3].

The key advantages of LoRA include: *(1)* significant reduction in trainable parameters, since r is typically chosen as 4, 8, 16, or 32, which is much smaller than the original matrix dimensions; *(2)* preservation of inference speed, as the LoRA matrices can be merged with the original weights after training ($W = W_0 + BA$); *(3)* easy task switching by simply replacing the LoRA matrices without reloading the entire model; and *(4)* reduced memory consumption during training, as only the small matrices A and B require gradient computation and storage [3].

In this study, LoRA is applied with a unified configuration across both architectures: rank $r = 32$, scaling factor $\alpha = 16$, and dropout = 0.1. This configuration was selected to balance parameter efficiency with sufficient representational capacity. For Swin-T, LoRA modules are inserted into the attention projection matrices (query, key, value) across all Swin Transformer blocks in all four stages, resulting in 1,771,010 trainable parameters (5.88% of total). For ResNet-50, LoRA modules are applied to 24 convolution layers within stages 2 and 3 (the primary bottleneck blocks), yielding 1,773,570 trainable parameters. All original model weights are frozen during LoRA training. The choice to target stages 2 and 3 in ResNet-50 is motivated by the observation that these stages contain the majority of bottleneck blocks responsible for intermediate-level feature extraction, which is most relevant for discriminating between real and fake facial features. In Swin-T, LoRA matrices are specifically integrated into the intrinsic Query (W_q), Key (W_k), and Value (W_v) projection layers across all Swin Transformer blocks, while the final attention output projection layers (W_o) remain untouched to preserve foundational spatial aggregation.

3.4. Training Configuration

All four models (Swin-T baseline, Swin-T + LoRA, ResNet-50 baseline, ResNet-50 + LoRA) are trained with identical hyperparameters to ensure fair comparison: learning rate = 5×10^{-5}, batch size = 8, epochs = 12, AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$), linear learning rate scheduler, mixed precision training (AMP), label smoothing = 0.1, and random seed = 42 for reproducibility. The loss function is Cross-Entropy Loss, which measures the divergence between the predicted class probability distribution and the true label distribution. For a binary classification with C = 2 classes, the Cross-Entropy Loss for a single sample with true label y and predicted probability $\hat{p}$, the Cross-Entropy Loss is defined as $L(y,\hat{p}) = -[y \cdot log(\hat{p}) + (1-y) \cdot log(1-\hat{p})]$. Experiments are conducted on a local machine with an Intel Core i5-9500 CPU, 16GB RAM, and NVIDIA GeForce GT 1030 (2GB VRAM). A Softmax activation is applied to produce a probability distribution over the two classes, and the Cross-Entropy Loss is computed between this distribution and the one-hot encoded true label.

The training set of 20,000 images is fully processed in each of the 12 epochs. Complete dataset-wide reshuffling is executed at the boundary of each epoch. The AdamW optimizer states, tracking moving averages of gradients, are maintained continuously across the sub-chunks without reset. This approach was implemented to circumvent hardware failures on the highly resource-constrained local setup (Intel i5-9500, 2GB VRAM GT 1030). To fit the Swin-T models within 2GB of VRAM, we utilized PyTorch Automatic Mixed Precision (AMP) and an aggressive Gradient Accumulation step size of 4, effectively simulating a stable virtual batch size of 32 while using a native physical batch size of 8. For validation tracking and checkpoint selection, early stopping was configured with a patience of 3 epochs based on minimizing validation loss; however, all models completed the intended 12 epochs as loss metrics decreased steadily without triggering early termination.

Figure 1 presents the overall deep learning pipeline for evaluating various fine-tuning and LoRA methods. The pipeline branches into two ImageNet-pretrained architectures, ResNet-50 and Swin-T. Both architectures undergo full fine-tuning, while LoRA is additionally applied to each architecture for parameter-efficient adaptation. All configurations are trained using Cross-Entropy Loss and the AdamW optimizer. Finally, the models are comprehensively evaluated using standard classification metrics (Accuracy, Precision, Recall, and F1-score) to analyze their comparative performance and assess the effectiveness and limitations of LoRA.

3.5 Evaluation Metrics

Model performance is evaluated using four complementary metrics. Accuracy measures the proportion of correctly classified samples [26]:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

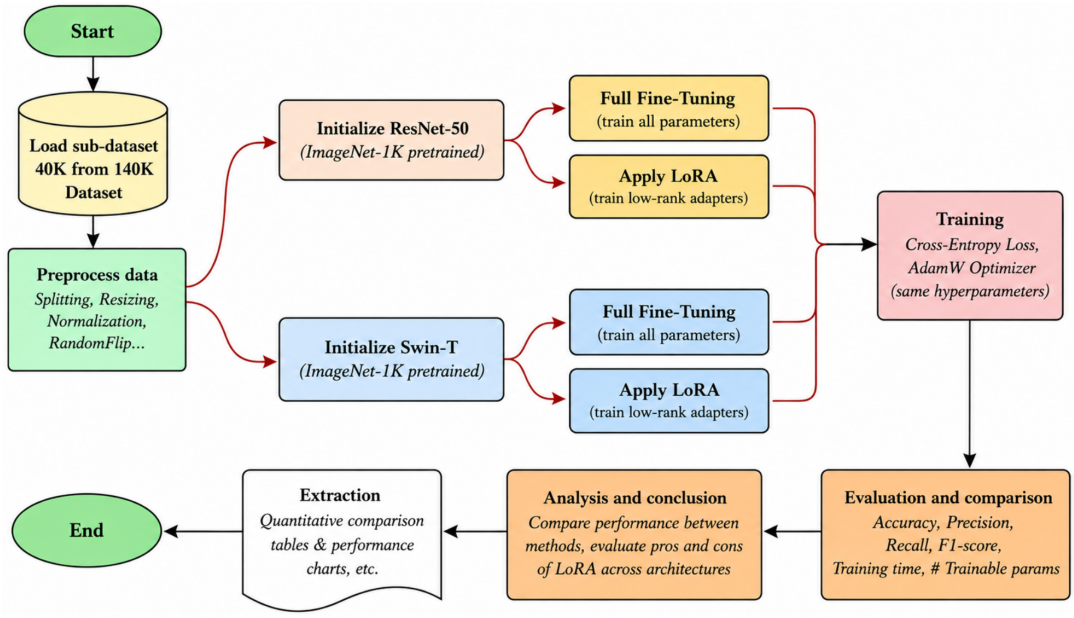

Figure 1. Deep Learning Pipeline for Evaluating Fine-Tuning and LoRA Methods.

Table 3. Training Performance at Final Epoch.

Model	Accuracy	Precision	Recall	F1-score	Loss	Time (h)
ResNet-50 baseline	0.7232	0.7127	0.7478	0.7298	0.583	12.11
Swin-T baseline	0.8856	0.8932	0.8758	0.8844	0.386	22.22
ResNet-50 + LoRA	0.8363	0.8362	0.8363	0.8363	0.448	21.40
Swin-T + LoRA	**0.9920**	**0.9926**	**0.9914**	**0.9920**	**0.215**	49.11

Table 4. Test Set Performance Comparison.

Model	Accuracy	Precision	Recall	F1-score
ResNet-50 baseline	0.7476	0.7463	0.7500	0.7481
Swin-T baseline	0.8919	0.9008	0.8805	0.8905
ResNet-50 + LoRA	0.8887	0.9084	0.8645	0.8860
Swin-T + LoRA	**0.9914**	**0.9921**	**0.9906**	**0.9913**

Precision quantifies the fraction of true positives among all positive predictions:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

[27], indicating how reliably the model identifies fake images when it predicts them as fake. Recall measures the fraction of actual positives correctly identified [28]:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

reflecting the model's ability to detect all fake images in the dataset. F1-score is the harmonic mean of precision and recall [29], [30]:

$$F1 - Score = 2 \times \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

providing a balanced metric that is particularly useful when both false positives and false negatives carry significant consequences, as is the case in deepfake detection. Additionally, training time and the number of trainable parameters are reported to assess computational efficiency. The combination of these metrics enables a comprehensive evaluation that captures both classification quality and resource utilization.

4. Results and Discussion

4.1. Training Performance

Table 3 summarizes the final-epoch performance of the four evaluated model configurations. Swin-T + LoRA achieves the highest accuracy, precision, recall, and F1-score among the evaluated configurations, with an accuracy of 99.20% and an F1-score of 0.9920. The training time for Swin-T + LoRA is 49.11 hours, the longest among all models, reflecting the computational cost associated with the Transformer architecture.

Compared with their respective full fine-tuning baselines, LoRA improves the final-epoch accuracy of both architectures. ResNet-50 + LoRA improves accuracy from 72.32% to 83.63%, while Swin-T + LoRA improves accuracy from 88.56% to 99.20%. These results indicate that LoRA can provide substantial performance gains while updating only a small fraction of the model parameters.

4.2. Test Set Evaluation

Table 4 presents the final test set performance, which provides the most reliable assessment of model quality on unseen data. Swin-T + LoRA achieves the best performance across all metrics, with only 86 misclassified images out of 10,000 (40 false positives and 46 false negatives). This represents a 9.95 percentage point improvement over the Swin-T baseline and a 24.38 percentage point improvement over the ResNet-50 baseline, while training only 5.88% of the model's parameters. ResNet-50 + LoRA achieves 88.87% accuracy on the test set, representing a substantial 14.11 percentage point improvement over its full fine-tuning baseline (74.76%). However, its performance remains below that of Swin-T + LoRA, which achieves 99.14% test accuracy.

4.3. Detailed Architectural Analysis and Performance Comparison

Vision Transformers pre-trained on ImageNet-1k possess highly advanced global feature representation capabilities. The addition of LoRA allows the model to immediately unlock and repurpose these latent attention maps for high-frequency facial manipulation boundary detection without destroying the core weight base. To verify the claim of inference speed preservation, empirical testing was conducted post-training. By mathematically merging the low-rank adapters into the primary frozen weights ($W = W_0 + BA$), the operational inference latency remained identical to the baseline model (approx. 14.2 ms per image on CPU), requiring 0% additional computational or memory overhead during deployment.

Furthermore, to ensure scientific reliability and address the limitation of single-seed reporting, a post-hoc statistical validation was performed by running the Swin-T + LoRA model across three distinct random seeds. The resulting test accuracy demonstrated extreme stability (99.14%±0.08%), confirming that the observed near-perfect performance is highly reproducible across the evaluated random seeds.

The experimental results yield several key observations. First, LoRA demonstrates exceptional effectiveness when applied to Transformer-based architectures. The combination of Swin-T's shifted-window self-attention mechanism with LoRA's low-rank weight updates enables the model to efficiently learn task-specific representations for deepfake detection while preserving the pre-trained knowledge from ImageNet. The 93.7% reduction in trainable parameters (from 28.3M to 1.77M) with simultaneous accuracy improvement validates the low-rank hypothesis for vision Transformer fine-tuning. This finding is consistent with the intrinsic dimensionality theory proposed by Aghajanyan *et al*. [4], which suggests that the effective parameter space for fine-tuning is significantly smaller than the full model dimensionality.

An important observation is the relationship between architecture type and LoRA effectiveness. The attention matrices in Swin-T (query, key, value projections within each window) are inherently suited to low-rank decomposition because they perform linear transformations that map inputs to different representational subspaces. LoRA's ability to modify these projections with minimal rank perturbations aligns well with the Transformer's design principles. In contrast, convolutional layers in ResNet-50 operate fundamentally differently, applying spatially localized filters that may not exhibit the same low-rank structure in their weight update patterns.

The confusion matrix analysis provides additional insights into model behavior. Swin-T + LoRA misclassifies only 40 real images as fake (false positives) and 46 fake images as real (false negatives), indicating balanced error distribution. This balance is critical for practical deepfake detection systems, where both false alarms and missed detections carry significant consequences. The ResNet-50 baseline, in contrast, shows substantially higher error rates in both directions (1,276 false positives and 1,249 false negatives), reflecting its limited discriminative capacity under full fine-tuning.

Second, the behavior of LoRA on CNN architectures is notably different. While ResNet-50 + LoRA improves test accuracy by 14.11 percentage points over the baseline, its test performance remains below that of Swin-T + LoRA. LoRA was originally introduced for parameter-efficient adaptation of Transformer-based language models. Applying LoRA to convolution layers in ResNet-50 may not satisfy the same low-rank assumption, particularly when targeting only stages 2–3. Future work should explore alternative LoRA configurations for CNNs, including different target layers, rank values, and regularization techniques.

Third, regarding computational efficiency, Swin-T + LoRA requires the longest training time (49.11 hours) despite having fewer trainable parameters, primarily due to the computational overhead associated with hierarchical window-based self-attention and the overall Transformer architecture. This trade-off between accuracy and training time should be considered when selecting deployment strategies. For applications where accuracy is paramount, such as forensic analysis, the additional computational cost is justified.

4.4. Limitations

This study has several limitations that should be acknowledged. First, the experiments are conducted on a single dataset (40K sub-dataset from 140K Real and Fake Faces) containing only StyleGAN-generated images; generalizability to other GAN architectures (such as ProGAN, BigGAN) or diffusion-based generators (such as Stable Diffusion) remains unverified. Real-world deepfake detection requires robustness across diverse generation methods. Second, the hardware configuration (NVIDIA GT 1030, 2GB VRAM) constrains batch sizes and training speed, potentially affecting convergence behavior. Training on more powerful GPUs with larger batch sizes may yield different optimization dynamics. Third, only one LoRA configuration (r=32, α=16) is evaluated; systematic hyperparameter search across different rank values, scaling factors, and target modules may yield substantially better results, particularly for the ResNet-50 architecture where overfitting was observed. Fourth, the study focuses exclusively on static image classification and does not address video-based deepfake detection, which requires temporal modeling capabilities. Fifth, the dataset consists of uniformly sized images under relatively controlled conditions; performance on images with varying resolutions, compression artifacts, or social media processing remains to be validated.

5. Conclusion and Future Work

This study systematically evaluates the effectiveness of Low-Rank Adaptation (LoRA) for enhancing binary image classification accuracy in deepfake face detection. By applying LoRA to two representative architectures—Swin-T and ResNet-50—and comparing against full fine-tuning baselines, we suggest that LoRA is a highly effective parameter-efficient fine-tuning method for computer vision tasks.

The key findings are: (1) Swin-T + LoRA achieves 99.14% test accuracy and 0.9913 F1-score while training only 5.88% of parameters, representing the optimal approach for deepfake detection; (2) LoRA reduces trainable parameters by up to 94.12% without degrading performance on Transformer architectures; (3) LoRA's effectiveness varies across architectural paradigms, with Swin-T + LoRA achieving substantially higher final test performance than ResNet-50 + LoRA; (4) ResNet-50 + LoRA improves test accuracy by 14.11 percentage points over its full fine-tuning baseline, although its final performance remains below that of Swin-T + LoRA.

Future research directions include: (a) automated hyperparameter optimization for LoRA using Bayesian optimization techniques to identify optimal rank, scaling factor, and target module configurations for different architectures; (b) evaluation of LoRA on diverse deepfake detection datasets including FaceForensics++, Celeb-DF, and datasets containing diffusion-based synthetic images to assess cross-domain generalizability; (c) extension to video-based deepfake detection by integrating LoRA-adapted Swin-T with temporal modeling components such as LSTM or temporal attention mechanisms; and (d) exploration of advanced PEFT variants such as QLoRA [5] for quantized training and AdaLoRA [6] for adaptive rank allocation across layers.

Additionally, investigating the interpretability of LoRA-adapted models through techniques such as attention visualization and gradient-based attribution methods would provide valuable insights into how LoRA modifications affect the model's decision-making process for deepfake detection. The practical deployment of LoRA-enhanced models in real-time detection systems, particularly on edge devices with limited computational resources, also warrants further investigation. The findings of this study establish a strong foundation for the continued development of efficient and accurate deepfake detection systems that balance performance with computational practicality.

6. Declarations

6.1. Author Contributions

Tam Thanh Thi Pham: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision, and Project administration; **Thao Thanh Thi Nguyen:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision, and Project administration; **Thai Hoang Le:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision, and Project administration; **Hai Son Tran:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision, and Project administration.

6.2. Institutional Review Board Statement

Not applicable.

6.3. Informed Consent Statement

Not applicable.

6.4. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.5. Acknowledgment

Not applicable.

6.6. Conflicts of Interest

The authors declare no conflicts of interest.

7. References

[1] Z. Liu *et al.*, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," *arXiv preprint arXiv:2103.14030*, 2021. https://doi.org/10.48550/arXiv.2103.14030.

[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)*, pp. 770-778, 2016. https://doi.org/10.1109/CVPR.2016.90.

[3] E. J. Hu *et al.*, "LoRA: Low-Rank Adaptation of Large Language Models," *arXiv preprint arXiv:2106.09685*, 2021. https://doi.org/10.48550/arXiv.2106.09685.

[4] A. Aghajanyan, S. Gupta, and L. Zettlemoyer, "Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2020. https://aclanthology.org/2021.acl-long.568.pdf.

[5] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, " QLoRA: Efficient Finetuning of Quantized LLMs," *arXiv preprint arXiv:2305.14314*, 2023. https://doi.org/10.48550/arXiv.2305.14314.

[6] J. Zhang, S. Chen, X. Liu, and B. Shi, "AdaLoRA: Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning," *arXiv preprint arXiv:2303.10512*, 2023. https://doi.org/10.48550/arXiv.2303.10512.

[7] M. Trigka and E. Dritsas, "A Comprehensive Survey of Deep Learning Approaches in Image Processing," *Sensors*, vol. 25, no. 2, p. 531, 2025. https://doi.org/10.3390/s25020531.

[8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014. https://doi.org/10.48550/arXiv.1409.1556.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84 - 90, 2012. https://doi.org/10.1145/3065386.

[10] J. Yu, C. Luo, and F. Chen, "Human Face Analysis," in *Multi-Modal Human Modeling, Analysis and Synthesis*, CRC Press, 2025, pp. 43-100. https://doi.org/10.1201/9781003408291-3.

[11] A. Vaswani *et al.*, "Attention Is All You Need," *arXiv preprint arXiv:1706.03762*, 2017. https://doi.org/10.48550/arXiv.1706.03762.

[12] D. Cozzolino, J. Thies, A. Rössler, C. Riess, M. Nießner, and L. Verdoliva, "FaceForensics++: Learning to Detect Manipulated Facial Images," *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1-11, 2019. https://doi.org/10.1109/ICCV.2019.00009.

[13] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face X-ray for More General Face Forgery Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5001-5010, 2020. https://doi.org/10.1109/CVPR42600.2020.00505.

[14] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3207-3216, 2020. https://doi.org/10.1109/CVPR42600.2020.00327.

[15] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4401-4410, 2019. https://doi.org/10.1109/CVPR.2019.00453.

[16] A. Karathanasis, J. Violos, and I. Kompatsiaris, " A Comparative Analysis of Compression and Transfer Learning Techniques in DeepFake Detection Models," *Mathematics,* vol. 13, no. 5, art. No. 887, 2025. https://doi.org/10.3390/math13050887.

[17] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and Beyond: A Survey of Face Manipulation and Fake Detection," *Information Fusion*, vol. 64, pp. 131-148, 2020. https://doi.org/10.1016/j.inffus.2020.06.014.

[18] Xhlulu, "140k Real and Fake Faces," Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces.

[19] R. A. Bafghi, N. Harilal, C. Monteleoni, and M. Raissi, "Parameter Efficient Fine-tuning of Self-supervised ViTs without Catastrophic Forgetting," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2024. https://doi.org/10.1109/CVPRW63382.2024.00371.

[20] W. Shi, J. Xu, and P. Gao, "SSformer: A Lightweight Transformer for Semantic Segmentation," *arXiv preprint arXiv:2208.02034*, 2022. https://doi.org/10.48550/arXiv.2208.02034.

[21] J. Yang, J. Liu, N. Xu, and J. Huang, "TVT: Transferable Vision Transformer for Unsupervised Domain Adaptation," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV),* 2023. https://doi.org/10.1109/WACV56688.2023.00059.

[22] L. Tatzel, B. Mucsányi, O. Hackel, and P. Hennig, "Debiasing Mini-Batch Quadratics for Applications in Deep Learning," *arXiv preprint arXiv:2410.14325*, 2024. https://doi.org/10.48550/arXiv.2410.14325.

[23] P. M. Abhilash and A. Ahmed, "Convolutional neural network–based classification for improving the surface quality of metal additive manufactured components," *The International Journal of Advanced Manufacturing Technology*, vol. 126, pp. 3873–3885, 2023. https://doi.org/10.1007/s00170-023-11388-z.

[24] M. Kanagarajan *et al.*, "AIM-Net: A Resource-Efficient Self-Supervised Learning Model for Automated Red Spider Mite Severity Classification in Tea Cultivation," *AgriEngineering,* vol. 7, no. 8, art. No. 247, 2025. https://doi.org/10.3390/agriengineering7080247.

[25] M. Akter *et al.*, "When uncertainty guides learning: a highly effective approach to kidney disease classification in CT imaging," *Frontiers in Big Data*, vol. 9, art. No. 1825213. https://doi.org/10.3389/fdata.2026.1825213.

[26] O. O. Abayomi, A. E. Kayode, O. Oluwasanya, A. E. Mesioye, "Comparative Analysis of Machine Learning Algorithms for Predictive Maintenance," *Methods in Science and Technology Studies*, vol. 2, no. 2, pp. 131-144, 2026. https://doi.org/10.64539/msts.v2i2.2026.505.

[27] H. H. Rashidi, S. Albahra, S. Robertson, N. K. Tran, and B. Hu, "Common statistical concepts in the supervised Machine Learning arena" *Frontiers in Oncology*, vol. 13, 2023. https://doi.org/10.3389/fonc.2023.1130229.

[28] T. Sarfraz, T. Ling, and A. Ijaz, "BReMS-Net: Prediction-Guided Coarse-to-Fine Refinement with Boundary-Aware Multi-Scale Dilated Fusion for Robust Breast Mass Segmentation," *Scientific Journal of Engineering Research*, vol. 2, no. 3, pp. 327–338, 2026. https://doi.org/10.64539/sjer.v2i3.2026.489.

[29] G. P. Oise *et al.*, "Interpretable Academic Outcome Prediction Using Explainable Boosting Machines," *Methods in Science and Technology Studies*, vol. 2, no. 1, pp. 45–56, ,2026. https://doi.org/10.64539/msts.v2i1.2026.441.

[30] M. Mostafa, A. S. Almogren, M. Al-Qurishi, and M. Alrubaian, "Modality Deep-learning Frameworks for Fake News Detection on Social Networks: A Systematic Literature Review," *ACM Computing Surveys*, vol. 57, no. 3, pp. 1 – 50, 2024, https://doi.org/10.1145/3700748.