VLM-based Universal Deepfake Detection: A Parameter-Efficient Approach

Cheng-Hao Chi

June 16, 2025

Abstract

This paper presents a comprehensive study on Parameter-Efficient Fine-Tuning (PEFT) methods for universal deepfake detection using Vision-Language Models (VLMs). We focus on Adapter Networks applied to CLIP for cross-forgery generalization, achieving **76.87%** accuracy with only **589K** trainable parameters. Our approach demonstrates significant improvement over linear probing while maintaining computational efficiency, establishing a practical baseline for cross-forgery detection.

1 Introduction

The proliferation of deepfake technology poses significant challenges to digital media authenticity. While existing detection methods achieve high performance on specific forgery types, they often fail to generalize across different generation techniques. This work addresses the critical need for universal deepfake detectors that can identify manipulated content regardless of the underlying generation method.

1.1 Motivation

Traditional deepfake detection approaches suffer from several limitations:

- **Domain Specificity**: Models trained on one forgery type perform poorly on others
- **Computational Cost**: Full fine-tuning requires substantial computational resources
- **Scalability**: Difficulty in adapting to new forgery methods without retraining

1.2 Contributions

Our main contributions include:

1. **PEFT for Deepfake Detection**: First comprehensive evaluation of Adapter Networks for cross-forgery detection
2. **Rigorous Evaluation Protocol**: Novel paired comparison methodology for cross-domain assessment
3. **Practical Implementation**: Efficient training pipeline with dual GPU optimization
4. **Comprehensive Analysis**: Detailed performance analysis across different forgery types

2 Methodology

2.1 Problem Formulation

Given a dataset $\mathcal{D} = \{(x_i, y_i)\}$ where x_i represents facial images and $y_i \in \{0, 1\}$ indicates real (0) or fake (1), we aim to learn a function $f : \mathcal{X} \to \mathcal{Y}$ that generalizes across different forgery methods $\mathcal{F} = \{F_1, F_2, \ldots, F_n\}$.

2.2 Adapter Networks

Adapter Networks insert small neural networks between layers of a pre-trained model:

$$h_{l+1} = \text{Adapter}(h_l) + h_l \tag{1}$$

where h_l is the hidden representation at layer l, $\text{Adapter}(\cdot)$ is a small feedforward network, and the residual connection preserves original information.

Architecture Details:

- **Bottleneck Design**: Down-projection → ReLU → Up-projection
- **Dimension**: 768 → 64 → 768 (for ViT-L/14)
- **Parameters**: 589,824 trainable (vs. 400M+ in full fine-tuning)

2.3 CLIP Integration

CLIP (Contrastive Language-Image Pre-training) provides strong visual representations:

- **Visual Encoder**: ViT-L/14 (768-dimensional features)
- **Frozen Weights**: Only adapter parameters are trainable
- **Classification Head**: Linear layer for binary classification

2.4 Cross-Forgery Evaluation Protocol

Dataset Split Strategy:

- Training: Real + FaceSwap (80%)
- Validation: Real + FaceSwap (10%)
- Testing: Real + FaceSwap + NeuralTextures (10%)

Paired Comparison Analysis: For videos v with both FaceSwap (F_s) and NeuralTextures (F_n) versions, we extract frames from both versions, evaluate model performance on each, and compare cross-domain generalization.

3 Experimental Setup

3.1 Dataset

FaceForensics++: High-quality facial manipulation dataset

- **Real Videos**: 1,000 YouTube videos
- **FaceSwap**: Face replacement using computer graphics
- **NeuralTextures**: Texture-based facial reenactment
- **Quality**: Raw (uncompressed) for maximum fidelity

3.2 Training Configuration

Table 1: Training Configuration

Parameter	Value	Justification
Epochs	50	Sufficient for convergence
Batch Size	64	Optimal for dual GPU
Learning Rate	0.004	Linear scaling rule
Optimizer	SGD	Stable convergence
Scheduler	Cosine	Smooth decay
Warmup	2 epochs	Stable initialization

3.3 Hardware Setup

- **GPUs**: 2× NVIDIA TITAN RTX (24GB each)
- **CPU**: AMD Ryzen Threadripper 3960X (24 cores)
- **Memory**: 128GB RAM
- **Training Time**: 3 hours 42 minutes

Table 2: Overall Performance Results

Metric	Value	Comparison
Test Accuracy	**76.87%**	+6.49% vs. Linear Probing
Average Precision	**97.26%**	Excellent ranking
Macro F1	**58.38%**	Balanced performance
Parameters	**589,824**	1400× fewer than full FT

4 Results

4.1 Overall Performance

4.2 Cross-Forgery Analysis

Domain-Specific Performance:

- **FaceSwap (Training Domain)**: 93.5% accuracy (1,010 samples)
- **NeuralTextures (Cross-Domain)**: 80.9% accuracy (10,100 samples)
- **Performance Gap**: +12.6% (indicates moderate domain bias)

Paired Video Analysis:

- Analyzed Videos: 100 paired videos
- Correlation: 0.73 between domain performances
- Consistency: 82% videos show same prediction trend

4.3 Training Dynamics

Convergence Analysis:

- Epoch 1-5: Rapid learning (55% → 92% accuracy)
- Epoch 5-20: Steady improvement (92% → 96% accuracy)
- Epoch 20-50: Fine-tuning (96% → 98.7% accuracy)

4.4 Comparison with Baselines

Table 3: Method Comparison

Method	Accuracy	Parameters	Time
Adapter Network	**76.87%**	**589K**	**3.7h**
Linear Probing	70.38%	1.5K	0.9h
Full Fine-tuning*	∼85%*	400M+	∼20h*

5 Discussion

5.1 Strengths

1. **Parameter Efficiency**: Achieves strong performance with minimal parameters
2. **Practical Training**: Reasonable computational requirements
3. **Cross-Domain Capability**: Generalizes to unseen forgery types
4. **Scalable Architecture**: Easy to adapt to new domains

5.2 Limitations

1. **Domain Bias**: 12.6% performance gap between domains
2. **Limited Scope**: Only evaluated on 2 forgery types
3. **Dataset Scale**: Relatively small compared to modern standards
4. **Temporal Information**: Ignores video-level temporal cues

5.3 Analysis of Domain Bias

The observed domain bias (FaceSwap > NeuralTextures) suggests:

- **Artifact Specificity**: Model learns FaceSwap-specific artifacts
- **Generalization Challenge**: Cross-domain features are harder to learn
- **Training Strategy**: May benefit from domain adaptation techniques

6 Conclusion

This work demonstrates the viability of Parameter-Efficient Fine-Tuning for universal deepfake detection. Adapter Networks achieve 76.87% accuracy with only 589K trainable parameters, representing a significant improvement over linear probing while maintaining computational efficiency.

Key findings include:

- **PEFT Effectiveness**: Adapter Networks provide strong performance with minimal parameters
- **Cross-Domain Challenge**: 12.6% performance gap highlights generalization difficulty
- **Evaluation Importance**: Paired comparison reveals true cross-domain capability
- **Practical Viability**: Reasonable training requirements enable broader adoption

The results establish a strong baseline for PEFT-based deepfake detection and highlight important directions for future research in universal manipulation detection.

References

[1] S. A. Khan and D.-T. Dang-Nguyen, “CLIPping the Deception: Adapting Vision-Language Models for Universal Deepfake Detection,” Feb. 20, 2024, arXiv: arXiv:2402.12927. doi: 10.48550/arXiv.2402.12927