Deepfake Detection Using Vision Transformer

A Comprehensive Analysis and Implementation on Celeb-DF-v2 Dataset

Adithya MS - CS22B1098
Vignesh Aravindh B - CS22B2004
Ashiq Irfan - CS22B2021

IIITDM Kancheepuram

May 4, 2025

Abstract

The rapid advancement of generative models has led to the proliferation of deepfake technology, posing significant challenges to digital media authenticity. This work presents a novel approach to deepfake detection leveraging Vision Transformers (ViT), a state-of-the-art architecture originally designed for image recognition tasks. Our implementation on the challenging Celeb-DF-v2 dataset demonstrates remarkable performance, achieving 97.59% accuracy with an AUC of 0.9943. The key innovation lies in our adaptive training strategy, which employs differential learning rates to optimize the balance between feature extraction and classification capabilities. Through comprehensive evaluation and visualization, we prove the effectiveness of transformer-based architectures in detecting sophisticated deepfake manipulations.

Contents

1 Introduction

1.1 The Deepfake Challenge

In recent years, the emergence of generative adversarial networks (GANs) and other deep learning techniques has revolutionized the field of visual content creation. While these advances have numerous beneficial applications, they have also enabled the creation of highly realistic fake videos, commonly known as "deepfakes." These synthetic media can be used to spread misinformation, create non-consensual content, and undermine trust in digital media.

The detection of deepfakes presents a significant technical challenge due to the increasing sophistication of generation algorithms. Traditional computer vision techniques that rely on obvious artifacts or telltale signs of manipulation are becoming less effective as the quality of synthetic content improves. This necessitates the development of more advanced detection methods that can identify subtle inconsistencies and patterns invisible to the human eye.

1.2 Our Approach: Vision Transformers

Our work explores the application of Vision Transformers (ViT) to the deepfake detection problem. While Convolutional Neural Networks (CNNs) have traditionally dominated computer vision tasks, the emergence of ViT has demonstrated superior performance in capturing global context and complex spatial relationships within images. These characteristics make ViT particularly well-suited for detecting the subtle artifacts and inconsistencies present in deepfake videos.

The fundamental insight driving our approach is that deepfakes, despite their visual fidelity, contain microscopic irregularities in facial movements, lighting consistency, and texture patterns. Traditional CNNs, with their localized receptive fields, may miss these global inconsistencies. In contrast, ViT's self-attention mechanism enables the model to establish connections between distant regions of an image, potentially identifying suspicious correlations that betray synthetic manipulation.

2 Dataset and Preprocessing

2.1 Celeb-DF-v2: A Challenging Benchmark

For our experiments, we utilized the Celeb-DF-v2 dataset, one of the most challenging benchmarks in deepfake detection. This dataset represents a significant advancement over earlier collections, featuring high-quality deepfakes that closely mimic real facial movements and expressions. The dataset's composition reflects the complexity of real-world deepfake scenarios:

The dataset comprises three distinct categories: celebrity videos featuring 59 prominent personalities, genuine YouTube content showcasing natural environmental variations, and corresponding synthesized deepfakes created using state-of-the-art generation techniques. This diversity ensures our model encounters a wide range of scenarios during training, improving its generalization capabilities.

What sets Celeb-DF-v2 apart is the quality of its synthetic content. Unlike earlier datasets where obvious artifacts aided detection, the deepfakes in Celeb-DF-v2 exhibit minimal compression artifacts and maintain temporal consistency. This characteristic makes the dataset particularly suitable for testing advanced detection algorithms and pushing the boundaries of what's possible in deepfake identification.

2.2 Intelligent Frame Extraction Strategy

Video analysis for deepfake detection traditionally faces a computational challenge: processing entire videos frame-by-frame is prohibitively expensive. Our solution employs an intelligent sampling strategy that balances computational efficiency with comprehensive coverage.

Rather than random sampling, we implemented a temporal distribution approach that extracts 20 evenly spaced frames from each video. This method ensures temporal representativeness while capturing key moments throughout the video sequence. The extraction algorithm adapts to varying video lengths, maintaining consistent sampling density regardless of the original content duration.

Listing 1: Intelligent Frame Extraction Algorithm

```python
def extract_frames_with_temporal_distribution(video_path, num_frames=20):
    cap = cv2.VideoCapture(video_path)
    total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))

    # Calculate frame indices for uniform temporal sampling
    if total_frames <= num_frames:
        indices = range(total_frames)
    else:
        # Ensure first and last frames are included
        indices = np.linspace(0, total_frames-1, num_frames).astype(int)

    extracted_frames = []
    for idx in indices:
        cap.set(cv2.CAP_PROP_POS_FRAMES, idx)
        ret, frame = cap.read()
        if ret:
            extracted_frames.append(frame)

    cap.release()
    return extracted_frames
```

2.3 Addressing Class Imbalance

A critical challenge in deepfake detection is the inherent class imbalance in real-world scenarios. After frame extraction, we obtained approximately 17,781 real frames versus 121,000 fake frames. This disparity could lead to a model biased toward detecting fakes, potentially increasing false positive rates.

Our approach to this challenge involved strategic oversampling of real frames combined with balanced batch composition during training. Rather than simply duplicating real frames, which could lead to overfitting, we applied diverse augmentation techniques to each real frame, creating variations that maintain the essential characteristics while introducing controlled variability.

3 Vision Transformer Architecture

3.1 The Power of Self-Attention

The Vision Transformer architecture represents a paradigm shift from traditional convolutional approaches. At its core, ViT transforms images into sequences of patches, treating each patch as a token similar to words in natural language processing. This tokenization allows the model to apply transformer machinery, originally developed for language tasks, to visual data.

The self-attention mechanism is particularly powerful for deepfake detection. Unlike convolutions that process local neighborhoods, self-attention computes relationships between all patches in

an image simultaneously. This global perspective enables the model to identify inconsistencies between distant facial regions – for instance, detecting when lip movements don't align with eye expressions, a common artifact in synthetic videos.

3.2 Architecture Adaptation for Deepfake Detection

Our implementation builds upon the ViT-B/16 architecture, which processes images in 16×16 pixel patches. This specific configuration balances computational efficiency with spatial granularity, allowing the model to capture fine details while maintaining reasonable processing speed.

The architecture consists of:

- **Patch Embedding Layer**: Converts the input image into a sequence of fixed-size patches, each represented as a vector through linear projection.
- **Position Encoding**: Adds spatial information to patch embeddings, preserving the relative locations of facial features.
- **Transformer Encoder Stack**: Twelve layers of transformer blocks, each containing multi-head self-attention and feed-forward networks.
- **Classification Head**: A specialized component we modified for binary classification, distinguishing real from fake content.

The modification of the classification head was crucial. The original ViT architecture was designed for multi-class ImageNet classification. For deepfake detection, we replaced the final layer with a single-neuron output coupled with sigmoid activation, optimizing for binary classification tasks.

Listing 2: ViT Architecture Adaptation

```
class DeepfakeViT(nn.Module):
    def __init__(self, backbone='vit_b_16', pretrained=True):
        super().__init__()

        # Load pre-trained ViT architecture
        if pretrained:
            self.vit = torchvision.models.vit_b_16(weights=ViT_B_16_Weights.
                DEFAULT)
        else:
            self.vit = torchvision.models.vit_b_16()

        # Adapt classification head for binary deepfake detection
        in_features = self.vit.heads.head.in_features
        self.vit.heads.head = nn.Sequential(
            nn.Linear(in_features, 512),
            nn.ReLU(),
            nn.Dropout(0.2),
            nn.Linear(512, 1)
        )

    def forward(self, x):
        # Forward pass through the adapted architecture
        features = self.vit(x)
        return features
```

3.3 Leveraging Pre-trained Knowledge

A key decision in our implementation was the utilization of ImageNet pre-trained weights. This transfer learning approach provides several advantages:

First, the pre-trained model has already learned fundamental visual features – edge detection, texture recognition, and object boundaries – that are invaluable for detecting inconsistencies in deepfakes. Second, the pre-trained weights provide a strong initialization point, reducing training time and improving convergence stability.

However, we recognized that deepfake-specific features differ from natural image features. Therefore, we implemented a careful fine-tuning strategy that allows the model to adapt its learned representations while preserving valuable pre-trained knowledge.

4 Training Methodology

4.1 Differential Learning Rate Strategy

Our training approach centers on an innovative differential learning rate strategy. This technique recognizes that different parts of the network require different optimization approaches:

The backbone layers, containing pre-trained knowledge, need subtle adjustments to adapt to deepfake-specific features. Aggressive learning rates could destroy valuable pre-trained representations. Conversely, the classification head starts with random initialization and requires more significant updates to develop discriminative capabilities.

We implemented this strategy as follows:

$$lr_{backbone} = 2 \times 10^{-5}, \quad lr_{head} = 2 \times 10^{-4} \tag{1}$$

This 10:1 ratio allows the classification head to adapt quickly to the deepfake detection task while the backbone layers gradually refine their feature extraction capabilities.

4.2 Advanced Optimization Techniques

Beyond learning rate differentiation, our training regimen incorporates several sophisticated techniques:

AdamW Optimization: We chose AdamW over standard Adam for its superior handling of weight decay. This regularization technique prevents the model from learning spurious correlations in the training data, a crucial consideration when detecting subtle deepfake artifacts.

Dynamic Learning Rate Scheduling: Our ReduceLROnPlateau scheduler monitors validation loss, reducing learning rates when improvements plateau. This approach ensures the model continues to refine its representations without overshooting optimal parameter values.

Early Stopping with Patience: To prevent overfitting while allowing sufficient training time, we implement early stopping based on validation AUC. The patience parameter ensures temporary fluctuations don't prematurely halt training.

Listing 3: Training Configuration

```
# Optimizer with differential learning rates
optimizer = optim.AdamW([
    {'params': model.vit.encoder.parameters(), 'lr': 2e-5},
    {'params': model.vit.heads.parameters(), 'lr': 2e-4}
], weight_decay=1e-4)

# Learning rate scheduler with monitoring
scheduler = optim.lr_scheduler.ReduceLROnPlateau(
    optimizer, mode='min', factor=0.5, patience=2, verbose=True
)
```

```
11
12 # Training loop with early stopping
13 best_auc = 0.0
14 patience_counter = 0
```

4.3 Data Augmentation Philosophy

Our augmentation strategy balances two competing objectives: increasing training data diversity while maintaining realistic facial characteristics. Excessive augmentation could introduce artifacts that confuse the model, while insufficient augmentation limits generalization.

We implemented:

- **Photometric Augmentations**: Color jitter with carefully tuned parameters to simulate lighting variations without creating unrealistic skin tones.
- **Geometric Augmentations**: Random horizontal flips to account for facial symmetry while avoiding vertical flips that could create unnatural appearances.
- **Normalization**: ImageNet statistics normalization to align with pre-trained weights.

5 Performance Analysis

5.1 Quantitative Results

Our model achieved remarkable performance across multiple evaluation metrics:

Metric	Description	Validation	Test	Interpretation
Accuracy	Overall correctness	97.55%	97.59%	Excellent classification
AUC	Discrimination ability	0.9991	0.9943	Superior performance
Precision	Positive prediction accuracy	0.9808	0.9843	High fake detection
EER	Error rate balance	0.040	0.038	Minimal errors

Table 1: Comprehensive Performance Metrics

The consistency between validation and test performance (97.55% vs 97.59% accuracy) indicates strong generalization. The AUC values near 1.0 demonstrate the model's exceptional ability to discriminate between real and fake content across all possible classification thresholds.

5.2 Confusion Matrix Analysis

The confusion matrix reveals the model's behavior in detail:

	Predicted Real	**Predicted Fake**
Actual Real	2,400 (90.0%)	267 (10.0%)
Actual Fake	205 (1.2%)	16,713 (98.8%)

Table 2: Confusion Matrix Breakdown

The asymmetry in error rates is noteworthy: false negative rate (10.0%) exceeds false positive rate (1.2%). This bias toward fake classification actually enhances the model's utility in real-world applications, where erroneously classifying fake content as real poses greater risks than the opposite scenario.

5.3 ROC Curve Interpretation

The ROC curve demonstrates exceptional performance with an AUC of 0.9943. The steep initial rise indicates the model can achieve high true positive rates with minimal false positives, crucial for practical deployment. The near-perfect curvature suggests the model maintains discrimination ability across all decision thresholds.

6 Implementation Insights

6.1 Architectural Advantages

The Vision Transformer architecture proved particularly effective for deepfake detection due to several key characteristics:

Global Context Awareness: Unlike CNNs that build up global understanding through hierarchical local features, ViT immediately processes global relationships. This capability enables detection of inconsistencies between facial regions – for instance, identifying when lip synchronization doesn't match eye movements.

Patch-Based Processing: The 16×16 patch size strikes an optimal balance. Smaller patches would capture excessive detail at the cost of global context, while larger patches might miss subtle artifacts. This granularity aligns well with facial feature scales.

Attention Mechanism Interpretability: The self-attention weights provide insights into which image regions the model prioritizes, offering a degree of explainability often lacking in traditional CNN approaches.

6.2 Training Methodology Benefits

Our differential learning rate strategy proved crucial for performance. Standard fine-tuning approaches often struggle with the trade-off between preserving pre-trained knowledge and adapting to new tasks. Our approach elegantly resolves this by allowing different network components to learn at appropriate rates.

The careful balance of regularization techniques prevented overfitting without sacrificing performance. Weight decay, dropout, and early stopping worked synergistically to ensure the model learned generalizable features rather than memorizing training artifacts.

7 Future Directions

7.1 Temporal Modeling Extensions

While our frame-based approach achieved excellent results, incorporating temporal information could further improve performance. Deepfakes often exhibit subtle inconsistencies in facial motion dynamics that single-frame analysis might miss.

Potential approaches include:

- **Sequential Models**: Combining ViT with LSTM or GRU layers to capture temporal patterns
- **Temporal Transformers**: Extending the attention mechanism to consider frame sequences
- **Motion Analysis**: Integrating optical flow or facial landmark tracking

7.2 Architecture Evolution

Future work could explore:

- **Larger ViT Variants**: Testing ViT-L/16 or ViT-H/14 for enhanced feature capacity
- **Hybrid Architectures**: Combining ViT's global awareness with CNN's local feature detection
- **Domain-Specific Adaptations**: Custom architectures designed specifically for facial manipulation detection

7.3 Advanced Detection Techniques

Additional enhancement possibilities include:

- **Frequency Analysis**: Incorporating DCT or wavelet transforms to detect frequency domain artifacts
- **Multi-Modal Fusion**: Combining visual analysis with audio inconsistency detection
- **Ensemble Methods**: Leveraging multiple models to improve robustness

8 Conclusion

This work demonstrates the exceptional effectiveness of Vision Transformers for deepfake detection, achieving state-of-the-art performance on the challenging Celeb-DF-v2 dataset. Our implementation showcases how thoughtful architectural adaptations, combined with sophisticated training strategies, can address complex computer vision challenges.

The success of our approach highlights the importance of global context awareness in detecting subtle manipulations. By processing images as sequences of interconnected patches rather than localized features, Vision Transformers capture the holistic inconsistencies that betray synthetic content.

Moving forward, the integration of temporal modeling and domain-specific architectural innovations promises to further advance the field of deepfake detection. As synthetic media generation techniques continue to evolve, our framework provides a robust foundation for developing adaptive and effective detection systems.

References

[1] Dosovitskiy, A., et al. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *International Conference on Learning Representations*.

[2] Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. *IEEE Conference on Computer Vision and Pattern Recognition*.

[3] Loshchilov, I., & Hutter, F. (2019). Decoupled Weight Decay Regularization. *International Conference on Learning Representations*.

[4] Goodfellow, I., et al. (2014). Generative Adversarial Networks. *Advances in Neural Information Processing Systems*.

[5] Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*.