Deep Fake Detection using Hybrid Neural Architectures

NEW JERSEY INSTITUTE OF TECHNOLOGY

FA25-DS677 **Shaury Pratap Singh**
Instructor: Akshay Rangamani **Bhanu Prakash**

Abstract

The proliferation of deepfake technology - synthetic media generated by Generative Adversarial Networks (GANs) -poses a significant threat to global information integrity, democracy, and public trust. As evidenced by recent high-profile incidents involving public figures such as Elon Musk and Jeff Bezos, the capability to manipulate facial identities has outpaced traditional verification methods. This project addresses the critical challenge of distinguishing between real and digitally manipulated faces, specifically focusing on the detection of subtle texture artifacts that often escape human perception.

Our research implements and evaluates a diverse set of neural architectures to tackle this problem, ranging from standard Convolutional Neural Networks (CNNs) to modern Vision Transformers (ViT) and frequency-domain classifiers. We utilized a perfectly balanced dataset of real and fake face crops to ensure unbiased model evaluation. The experimental pipeline compared five distinct architectures: LeNet (baseline), CBAM-ResNet18 (spatial attention), XceptionNet (depth wise texture analysis), Vision Transformer (global attention), and a Frequency-CNN (FFT-based).

Our findings demonstrate that while traditional CNNs struggle with the high-frequency noise inherent in deepfakes, modern architectures like the Vision Transformer (ViT) excel, achieving a single-model accuracy of 93.27%. Furthermore, we observed that heavy geometric augmentations degrade performance in this domain, whereas minimal preprocessing preserves the necessary manipulation traces. The study culminates in the development of a Weighted Ensemble model, which fuses the predictions of our top performers. This ensemble achieved a final test accuracy of 95.62%, proving that combining global attention (ViT) with local texture analysis (Xception) provides the most robust defense against deepfake manipulation.

Introduction

1.1 Background and Motivation In recent years, "Deepfakes" have transitioned from a niche academic curiosity to a global security concern. The term, a portmanteau of "deep learning" and "fake," refers to hyper-realistic video or audio forgeries generated using deep learning techniques, particularly Autoencoders and Generative Adversarial Networks (GANs). The accessibility of these tools has led to a surge in misinformation. As noted in our preliminary research, deepfakes have infiltrated sectors ranging from entertainment to politics, with notable examples including manipulated videos of tech leaders in the "Star Trek" universe and unauthorized synthesized audio used to scam CEOs out of millions of dollars.

The implications of this technology are profound. If left unchecked, deepfakes threaten to destabilize democratic processes, as seen in recent Indian election campaigns where manipulated content was used to sway voter opinion. The ability to seamlessly swap faces or lip-sync audio creates a "threat to truth," forcing society to question the veracity of video evidence.

1.2 Problem Statement Detecting deepfakes is uniquely challenging because modern GANs have become adept at synthesizing realistic facial geometry. The artifacts that distinguish a fake face from a real one are no longer obvious shape deformities or blurring. Instead, the cues are hidden in fine-grained texture inconsistencies, irregular noise patterns, and compression artifacts that are often invisible to the naked eye.

Standard image classifiers often fail in this domain because they prioritize high-level shape features (e.g., "is there a nose?") over the low-level texture statistics where deepfake traces reside. Furthermore, our initial experiments revealed that these models are highly sensitive to data augmentation; standard transformations used to improve robustness in general object detection can inadvertently destroy the subtle digital fingerprints needed to identify a deepfake.

1.3 Project Objectives The primary goal of this project was to build a robust detection system capable of generalizing across different types of manipulation. Specifically, we aimed to:

1. **Evaluate Multiple Architectures:** Compare the efficacy of standard CNNs (LeNet, ResNet) against specialized architectures (Xception) and Transformers (ViT).
2. **Analyze Feature Domains:** Investigate whether frequency-domain features (FFT) could expose artifacts that spatial models miss.
3. **Optimize for Texture:** Design a training pipeline that preserves manipulation artifacts through careful preprocessing and normalization strategies.
4. **Develop an Ensemble:** Construct a weighted ensemble model to maximize accuracy by leveraging the complementary strengths of different architectures.

Dataset and Preprocessing

2.1 Dataset Characteristics

Data quality is the cornerstone of any deep learning project. For this study, we utilized a consolidated dataset derived from FaceForensics++ and DFFD sources. A critical feature of our dataset setup was class balance; the data was perfectly split between "Fake" and "Real" samples. This balance was instrumental in stabilizing our validation and test performance, ensuring that our high accuracy metrics were not the result of the model simply guessing the majority class.

The dataset consists exclusively of tightly cropped face images. This homogeneity - where every image is a face without significant background noise - means that the models were forced to learn internal facial features rather than relying on background context. However, this also introduced a challenge: because the images were so similar structurally, the models became highly sensitive to slight variations in texture and noise.

2.2 Preprocessing Pipeline

Given the subtlety of deepfake artifacts, our preprocessing pipeline required careful tuning. We established a standardized input transformation process consisting of Resizing, Horizontal Flipping, and Normalization.

- **Normalization:** We found that normalizing images to a range of [-1, 1] (mean 0.5, std 0.5) was critical. This specific normalization drastically stabilized the early training epochs, particularly for the Xception and Vision Transformer models. Without this, the gradients in the early stages were too volatile, leading to slower convergence.
- **Augmentation Strategy:** A key finding of our experimental phase was the detrimental effect of heavy data augmentation. In standard computer vision tasks, heavy rotation and color jittering help models generalize. However, in deepfake detection, these augmentations distorted the facial geometry and "washed out" the fine texture cues we were trying to detect. We observed a consistent drop in validation accuracy across all CNNs when heavy augmentation was applied. Therefore, we restricted our pipeline to minimal transforms to preserve the integrity of the manipulation artifacts.

2.3 Data Splitting

To ensure the reliability of our results, we maintained strict separation between our training, validation, and testing sets. We observed that our validation accuracy correlated almost identically with our test accuracy throughout the experiments. This strong correlation indicates that our data splits were clean, with no "data leakage" (where training data accidentally bleeds into the test set), validating the integrity of our final performance metrics.

Methodology - Convolutional Architectures

3.1 Baseline Model: LeNet-224

To establish a baseline for performance, we first implemented LeNet-224. This simple convolutional network served as a control variable to determine if the task required complex feature extraction. The results were telling: LeNet underfit the data significantly, with accuracy saturating near 60% regardless of the training "tricks" we employed. The model lacked the representational capacity to capture the complex, high-frequency noise patterns indicative of deepfakes. It also displayed extreme sensitivity to learning rates; it only trained correctly at a high learning rate (1e-3), and any value lower than this caused it to get stuck making random predictions. This failure confirmed that deepfake detection requires deep, high-capacity architectures.

3.2 Spatial Attention: CBAM-ResNet18

Moving beyond the baseline, we implemented ResNet-18 augmented with the Convolutional Block Attention Module (CBAM). The hypothesis was that adding attention mechanisms would help the model focus on specific facial regions that are prone to manipulation, such as the eyes and mouth.

This model proved to be highly stable. Unlike LeNet, CBAM-ResNet18 reached a respectable accuracy of ~91.5% with low variance across epochs. The CBAM module significantly improved the model's localization capabilities, allowing it to suppress background noise and focus on relevant facial textures. We found that a learning rate of 1e-4 was optimal for this architecture; lowering it further caused a performance drop of 2-3%.

3.3 Texture Specialist: XceptionNet

The third model in our pipeline was XceptionNet. This architecture is widely regarded as a standard in deepfake detection due to its use of Depth wise Separable Convolutions. By decoupling spatial correlations from cross-channel correlations, Xception is theoretically better at capturing fine-grained texture noise.

Our experiments confirmed this theory. XceptionNet captured fine-grained texture noise extremely well, making it ideal for spotting deepfake traces. It consistently outperformed ResNet, reaching a validation accuracy of ~92.3%. However, this sensitivity came at a cost: Xception was the most fragile model regarding augmentations. Even small distortions or rotations caused fluctuations in its validation curve, as the model would attend to the high-frequency noise introduced by the augmentation rather than the deepfake itself.

3.4 Vision Transformer (ViT-B16)

The most advanced model in our lineup was the Vision Transformer (ViT). Unlike CNNs, which process images via local receptive fields (sliding windows), ViT splits the image into patches and processes them using a global self-attention mechanism. This allows the model to understand the relationship between distant parts of the face simultaneously - for example, recognizing that the texture of the left eye does not match the texture of the right eye.

Initially, ViT performed poorly in the first few epochs. However, once we implemented a cosine scheduler with a learning rate of 3e-5, the performance jumped significantly, eventually reaching 93.27%, making it our best single model. Our observations suggest that ViT succeeds because it attends to global facial relationships, excelling in cases where local texture cues are inconsistent or too subtle for CNNs to pick up. Interestingly, ViT was the only model that benefited from slightly heavier augmentation, likely because its attention mechanism is more robust to geometric shifts than the rigid grid of a CNN.

3.5 Frequency-Domain Analysis (Freq-CNN)

To complement our RGB-based models, we introduced a Frequency-CNN (Freq-CNN). Deepfake generation processes often leave behind periodic artifacts that are invisible in standard images but become obvious in the frequency domain. We implemented a pipeline that converted images using the Fast Fourier Transform (FFT) before feeding them into a CNN.

While the standalone accuracy of this model was lower (approx. 74%), its contribution was strategic. The Freq-CNN successfully revealed global compression artifacts that the spatial models missed. Training was computationally expensive - the first epoch was extremely slow due to the FFT calculations on every batch - but it sped up in later epochs. Crucially, this model provided a unique signal that boosted our final ensemble, proving that multi-domain learning improves overall robustness.

Training Strategy and Optimization Challenges

4.1 Optimization Protocol

Training deep neural networks on texture-sensitive data requires a highly tuned optimization strategy. We employed the Adam optimizer for all architectures due to its adaptive learning rate capabilities. Through our experimentation, we identified a batch size of 32 as the "sweet spot" for convergence. We observed that increasing the batch size beyond this point caused the loss landscape to flatten too early, reducing the final generalization accuracy. Conversely, smaller batches introduced too much noise into the gradient updates, destabilizing the high-frequency feature extraction of XceptionNet.

To further stabilize training, we implemented a Cosine Annealing Learning Rate Scheduler. This approach starts with a relatively high learning rate and decays it following a cosine curve. We found this essential for all our models, as it allowed them to escape local minima in the early epochs while settling into a precise optimum towards the end of training.

4.2 Technical Hurdles and Solutions

Our experimental logs reveal several critical optimization challenges that are unique to this project:

- **The AMP Slowdown:** Initially, we attempted to use Automatic Mixed Precision (AMP) to speed up training. However, incorrect usage led to extreme slowdowns during the first epoch. We discovered that the overhead of casting operations was misconfigured; explicitly fixing the autocast("cuda") scope restored normal training speeds and allowed us to leverage faster GPU computations.
- **Learning Rate Sensitivity:** We observed massive performance shifts caused by minute changes in learning rates, particularly for the Vision Transformer and Xception models. For instance, a small adjustment from 1e-4 to 3e-5 was the deciding factor between model collapse and state-of-the-art performance. This confirms that deepfake detection models occupy a narrow convergence basin that requires precise hyperparameter tuning.
- **Early Stopping Nuances:** We implemented early stopping to prevent overfitting, but we found that the "patience" parameter (the number of epochs to wait before stopping) had to be set strictly greater than 5. Lower values prematurely killed the training of XceptionNet, which often showed a plateau in mid-training before discovering finer texture features in later epochs.

Experimental Results (Quantitative Analysis)

5.1 Single Model Performance

The quantitative results of our experiments highlight a clear hierarchy in model capability. The list below summarizes the best test accuracy achieved by each individual architecture:

- **LeNet224:** ~61.9%
- **Freq-CNN:** 74.3%
- **CBAM-ResNet18:** 91.54%
- **XceptionNet:** 92.29%
- **ViT**: 93.27%

5.2 Performance Analysis

- **The Baseline Failure:** As expected, LeNet performed poorly, saturating near 60% accuracy. This result validates our hypothesis that shallow architectures lack the capacity to model the complex, non-semantic noise patterns inherent in GAN-generated images.
- **The Texture vs. Attention Battle:** The comparison between XceptionNet (92.29%) and ViT (93.27%) is the most revealing aspect of our study. XceptionNet, designed for texture, performed exceptionally well but was prone to fluctuations because it attends to high-frequency noise. ViT, however, emerged as the superior single model. Its ability to attend to global facial relationships allowed it to excel even when local texture cues were ambiguous.
- **The Stability of ResNet:** While not the top performer, CBAM-ResNet18 demonstrated high stability and low variance across epochs. It served as a reliable "middle-ground" model, offering better spatial localization than standard CNNs due to its attention mechanism.

5.3 Validation Dynamics

Our training logs showed that validation and test accuracy correlated almost identically, confirming that our dataset splits were clean with no leakage. Interestingly, all high-capacity models (ResNet, Xception, ViT) learned strongly in the first 2-3 epochs, indicating the strength of utilizing pretrained backbones even for this specialized task.

Qualitative Analysis (Explainability)

6.1 Class Activation Mapping (CAM)

To ensure our models were learning genuine deepfake artifacts rather than relying on background biases, we employed explainability techniques: Grad-CAM for CNNs and EigenCAM for the Vision Transformer. The visualizations confirmed that our models successfully targeted facial regions.

(Heatmap Images: CBAM, Xception, ViT)

- **CBAM-ResNet18 Focus:** The attention maps for this model concentrated heavily on the eyes, mouth, and face center. This aligns with common knowledge that deepfake generators often struggle with blinking patterns and lip-syncing, creating spatial artifacts in these regions.
- **XceptionNet Focus:** The visualizations for Xception were distinct; they highlighted high-frequency boundary textures and the edges of the face. This confirms that the model utilizes its depth wise convolutions to hunt for "blending artifacts"—the seams where a fake face is stitched onto a target background.
- **ViT Focus (EigenCAM):** The Vision Transformer displayed a distributed attention pattern across various patches of the face. Unlike the CNNs which focused on specific edges, ViT appeared to assess the consistency of the face as a whole.

6.2 Interpretation

The divergence in these visualizations is a critical finding. It proves that Xception and ViT are looking at fundamentally different signals: one looks at texture edges while the other looks at global consistency. This orthogonality (independence) of features is exactly why these models are perfect candidates for an ensemble approach; they do not make the same mistakes.

Ensemble Logic and Discussion

7.1 Ensemble Strategy

Given the complementary nature of our models - ResNet (spatial), Xception (texture), ViT (global), and Freq-CNN (frequency) - we constructed an ensemble to maximize performance. We tested three fusion strategies:

1. **Simple Averaging:** Averaging the probability scores of all models yielded a strong accuracy of **95.46%**, which already surpassed every single model.
2. **Hybrid Feature Fusion (MLP):** We attempted to concatenate the feature vectors of the models and train a Multi-Layer Perceptron (MLP) on top. This achieved **95.24%**. While effective, it slightly underperformed compared to weighted averaging, likely due to the added complexity of training a new classifier on limited data.
3. **Weighted Soft Voting (Best):** The optimal approach was a weighted ensemble, where weights were assigned based on the individual validation accuracy of each model. This method achieved the highest overall accuracy of **95.62%**.

7.2 Confusion Matrix Analysis

The robustness of the weighted ensemble is best illustrated by its confusion matrix.

```
Ensemble Accuracy: 95.46827794561933
              precision    recall  f1-score   support

           0       0.96      0.94      0.95       655
           1       0.95      0.97      0.96       669

    accuracy                           0.95      1324
   macro avg       0.95      0.95      0.95      1324
weighted avg       0.95      0.95      0.95      1324

[[618  37]
 [ 23 646]]
Weighted Ensemble Accuracy: 95.61933534743203
```

The matrix shows a balanced distribution of False Positives (FP) and False Negatives (FN). This balance is crucial for a security system; a model that is too aggressive (high FP) ruins user experience, while a model that is too lax (high FN) poses a security risk. Our ensemble manages to reduce the bias of individual models - specifically correcting ViT's occasional overconfidence - resulting in a highly reliable detector.

7.3 Why the Ensemble Works

The success of the ensemble (approx. 2.3% gain over the best single model) confirms that deepfake detection is a multi-modal problem.

- **ResNet** contributes spatial structure.
- **Xception** contributes fine texture analysis.
- **ViT** contributes global coherence.
- **Freq-CNN** contributes frequency artifact detection.

  By combining these signals, the ensemble becomes resilient to "adversarial" samples that might fool one specific architecture.

Conclusion and Future Scope

8.1 Conclusion

This project successfully demonstrated that a hybrid deep learning approach is superior to single-architecture solutions for deepfake detection. Through rigorous experimentation, we established that texture cues are the strongest signal for detection, but they are fragile under augmentation.

We identified the Vision Transformer (ViT) as the most capable individual model (93.27%), debunking the assumption that CNNs are strictly better for image forensics. However, the Weighted Ensemble (95.62%) proved to be the definitive solution, effectively covering the blind spots of individual networks. Our work highlights that normalizing data to [-1, 1] and using cosine schedulers are critical optimization tricks for stabilizing these high-capacity models.

8.2 Limitations

Despite our success, the system has limitations:

- **Augmentation Sensitivity:** Both ViT and Xception are highly sensitive to geometric augmentations like rotation, which can degrade performance.
- **Computational Cost:** The Freq-CNN component is computationally expensive during the first epoch due to the Fourier Transform operations.
- **Temporal Blindness:** Our models operate on a frame-by-frame basis, meaning they do not utilize the temporal inconsistencies (e.g., flickering) often present in deepfake videos.

8.3 Future Work

To address these limitations, future iterations of this project will focus on:

1. **Temporal Modeling:** Integrating Long Short-Term Memory (LSTM) networks or 3D-CNNs to analyze video sequences rather than static images.
2. **Landmark Analysis:** Training a PatchGNN classifier based on facial landmarks to detect geometric inconsistencies in warping.
3. **Real-Time Deployment:** Optimizing the ensemble for inference to deploy it as a real-time API for content moderation.

.