DeepFake Detection Using Multi-Stream Convolutional Neural Networks

Soubhagya Raha (231025)   Suvradip Das (231110051)
Guttula Viswa Venkata Yashwanth (230431)   Samarth Singh (230904)

Abstract

*Deepfakes pose a growing threat to digital media authenticity. We propose a novel, lightweight **multi-stream detection framework** that fuses **RGB**, **frequency domain** (FFT), and **motion residual** features. Each stream is encoded using **EfficientNet-B0** with **CBAM attention**, and fused via a **Transformer-based cross-stream attention** mechanism. We also apply Convolutional Block Attention Module to enhance temporal consistency.Along with other datasets, when evaluated on the CelebDfV2 dataset, our model achieves a strong **96% accuracy**, while remaining efficient and deployable. This highlights the power of **multi-modal attention fusion** for robust deepfake detection.*

1. Introduction

1.1. Understanding Deepfakes and the Necessity of Machine Learning-Based Detection

Deepfakes, synthetic media generated through advanced deep learning techniques, represent a significant and growing threat to the integrity of digital content. The proliferation of deepfakes—manipulated videos, images, or audio—has raised alarming concerns in areas such as misinformation, privacy violations, and security risks. Traditional methods of detecting such manipulations, primarily reliant on human inspection, are no longer sufficient given the increasing sophistication of deepfake technology. Therefore, there is an urgent need for automated, scalable solutions that can effectively identify these forged media, particularly in real-world, high-stakes scenarios.

Recent advancements in machine learning, especially in computer vision, have shown promise in detecting deepfakes through the use of convolutional neural networks (CNNs) and other deep learning models. However, many existing models primarily focus on single-modal inputs, such as RGB images or frame-level inconsistencies, which can be easily bypassed by more advanced deepfake generation methods. Additionally, they often fail to capture complex temporal patterns or artifacts that arise across frames in videos, a crucial aspect in distinguishing real from fake content.

To address these limitations, we propose a **novel multi-stream deepfake detection model** that leverages three distinct input modalities: **RGB images, frequency domain representations (via FFT), and motion residuals**. Each of these streams captures different aspects of the manipulation, from texture and appearance in the RGB stream, to high-frequency artifacts in the frequency domain, to temporal inconsistencies in motion across frames. By combining these three streams, our model can learn more robust and comprehensive representations of both real and fake content.

Moreover, our approach incorporates a **Transformer-based cross-stream attention mechanism**, which intelligently fuses features from these diverse modalities, allowing the model to focus on the most relevant regions of the video. This attention fusion mechanism, combined with efficient **EfficientNet-B0** backbones and **CBAM** attention within each stream, ensures that the model performs at a high level of accuracy while remaining lightweight and deployable in real-time applications.

This novel pipeline, combining **multi-modal input, cross-stream attention fusion, and lightweight architecture**, marks a significant step forward in deepfake detection. Unlike prior methods that rely on individual modalities or single-stream architectures, our approach leverages complementary information across different domains, resulting in superior generalization and resilience to sophisticated deepfake techniques. With an accuracy of **91%**, our model demonstrates its potential to detect even the most advanced deepfakes with high precision and robustness, making it a promising tool for both academic research and real-world deployment in media verification and security applications.

2. Related Works to this date

2.1. Deepfakes Detection

The fight against deepfakes has led to significant research, with early methods detecting low-level artifacts such as unnatural blinking patterns and inconsistencies in physiological signals. However, these techniques struggle against increasingly sophisticated deepfake generation methods. A more robust approach focuses on mesoscopic features like **facial warping artifacts**, **head pose inconsistencies**, and **textural anomalies**. **CNNs** have become powerful tools in artifact detection, with architectures like **XceptionNet** and the integration of **attention mechanisms** further enhancing detection capabilities.

To address data scarcity and improve adaptability, **self-supervised** and **semi-supervised** methods are gaining traction. However, many current methods are designed for detecting **GAN-generated samples**, limiting their ability to detect a wide range of deepfake generation techniques.

In recent years, **diffusion models** have set new standards in deepfake generation, producing high-quality images that closely resemble natural ones. These models avoid **grid-like artifacts** found in GAN outputs, presenting new challenges for detection. Research now focuses on analyzing the **intrinsic local dimensionality** of diffusion-generated images and their tendency to **overfit training data**, leaving detectable traces.

Despite progress, deepfake detection still faces challenges, with models often struggling to **generalize to unseen deepfake techniques** and their performance degrading in real-world scenarios like **video compression**.

2.2. Existing Models

2.2.1 Face X-Ray

Face X-Ray (CVPR 2020) detects deepfakes by analyzing blending artifacts in composited face regions, identifying boundaries between real and fake areas. It uses **RGB input** and a **ResNet-based backbone**.

2.2.2 Two Stream

Two-Stream (NeurIPS 2020) detects deepfakes by leveraging **spatial and frequency information** through two parallel input streams: RGB frames and **FFT-based frequency maps**. It uses dual CNN backbones to capture subtle artifacts and emphasizes **frequency-aware feature learning**.

2.2.3 MesoNet

MesoNet is a lightweight convolutional neural network proposed in IEEE 2018 for deepfake detection. It focuses on mesoscopic-level features, which lie between high-level semantics and low-level textures. Designed for efficiency, MesoNet uses a shallow architecture, making it suitable for real-time applications while still effectively identifying facial manipulations in videos and images.

2.2.4 F3-Net

F3-Net (AAAI 2021) detects deepfakes by learning **frequency-aware features**. It uses a **wavelet transform** for multi-scale frequency representations and a **feature reweighting mechanism** to highlight forgery artifacts, improving generalization across deepfake methods.

Component	**Existing Usage**
Multi-stream Input (RGB, Frequency, Motion)	Dual-stream models (e.g., RGB + noise/frequency) are common; motion residuals rarely combined with them.
Transformer-based Cross-Modal Fusion	Transformers used in single-stream or temporal models (e.g., TimeSformer).
Motion Residuals Input	Temporal artifacts studied via 3D CNNs; direct residual-based inputs are rare.
Per-Frame Attention Fusion	Aggregation typically via mean or max pooling.
EfficientNet + Lightweight Deployment	Heavy models (e.g., XceptionNet, 3D CNNs) dominate.

Table 1. Comparison of currently existing models and their usages.

3. Proposed Work

In this work, we propose a novel multi-stream convolutional neural network architecture that integrates RGB appearance, frequency spectrum (FFT), and motion difference modalities, utilizing attention mechanisms for feature enhancement and fusion, to address the task of deepfake detection using facial video data extracted from synchronized frames.

3.1. Motivation

Traditional single-stream CNN architectures struggle to leverage complementary information from multiple data modalities. Moreover, naïve fusion techniques such as feature concatenation or averaging may not adequately capture the varying importance of features across spatial, channel, and temporal dimensions. To address these limitations, we propose a multi-stream approach that processes each modality independently and then uses a hybrid attention mechanism to intelligently fuse the learned features.

3.2. Architecture Overview

The proposed framework consists of the following key components:

1. **Multi-Stream Feature Extraction:** Separate CNN backbones are used for each modality—one for RGB

Table 2. Key Innovations Across Aspects

Aspect	Innovation
Multi-Stream Fusion	Unlike single-CNN models, we use 3 informative modalities (RGB + Frequency + Motion).
Transformer-Based Cross-Stream Attention	Fuses features across different domains intelligently, boosting detection.
Motion Residuals as Input	Many models skip motion; we explicitly incorporate temporal inconsistencies.
Efficient Lightweight Models	Fast (EfficientNet) yet powerful — deployable on resource-constrained devices.
Per-Frame Attention Aggregation	Instead of naïve averaging, we consider spatial importance for final decision.

frames and another for Frequency domain and Motion Residuals. Each stream independently extracts high-level feature representations.

2. **Hybrid Attention Fusion Module:** We employ a transformer-based cross-modal interaction framework to effectively fuse features enabling rich contextual integration across modalities

3. **Classification Head:** The fused representation is passed through fully connected layers followed by a softmax classifier to predict the activity label.

3.3. Advantages of the Proposed Method

- **Modality-Specific Learning:** The use of separate streams enables each modality to learn specialized features without interference.

- **Dynamic Fusion:** Attention-based fusion allows the model to dynamically adapt the contribution of each stream based on context.

- **Improved Generalization:** The hybrid attention mechanism enhances the model's ability to focus on salient features, improving robustness and accuracy.

3.4. Implementation Plan

1. Preprocess the RGB ,Frequency Domain and Motion Residuals

2. Design and implement the multi-stream CNN architecture with attention modules using PyTorch.

3. Train the network end-to-end with cross-entropy loss and validate performance using accuracy, F1-score,ROC AUC score and confusion matrix.

3.5. Expected Outcomes

We expect the proposed Multi-Stream CNN with Hybrid Attention Fusion to outperform conventional fusion techniques. Additionally, attention maps generated by the model can offer insights into which spatial regions and modalities are most influential for recognition, enhancing interpretability.

4. Experimentation Details

4.1. Stage 1: Data Preparation

The first stage of our pipeline involves preparing high-quality, standardized data for training and evaluation. Given the sensitivity of deepfake detection to visual artifacts, it is crucial to ensure that the input frames are clean, aligned, and representative of the temporal dynamics of manipulated videos.

Dataset

We employ publicly available datasets that are widely used in deepfake research:

- **FaceForensics++ (FF++)**: A benchmark dataset consisting of both real and manipulated facial videos. The fake videos are generated using various face-swapping and expression-reenactment techniques.

- **DeepFake Detection (DFD) Entire Dataset**: Another comprehensive dataset curated by Google, consisting of real and synthetic videos across multiple identities and synthesis methods.

- **Celeb-DF (v2)**: It is a **large-scale dataset** containing real and DeepFake videos with high visual quality, similar to those circulating online.

Preprocessing Steps

To ensure consistency and robustness in model performance, the following preprocessing pipeline is applied:

1. **Frame Extraction:** From each video, we uniformly sample 5–10 frames to reduce temporal redundancy and ensure efficient training. Frame sampling helps capture different facial expressions and lighting variations without overloading the model with similar information.

2. **Face Detection:** We utilize **MTCNN** for robust face localization. MTCNN is effective under standard conditions.It doesn't just detect faces, it also outputs key facial landmarks, which is essential for face alignment and further tasks like face recognition or manipulation.

3. **Face Alignment and Cropping:** Detected faces are aligned to a canonical pose using facial landmarks. This alignment ensures that the eyes, nose, and mouth are consistently placed across samples. The aligned faces are then cropped tightly to remove background noise.

4. **Resizing:** All cropped face images are resized to 224×224 pixels to balance computational efficiency with the preservation of fine-grained facial details.

This preprocessing step is essential for creating a standardized and noise-reduced dataset, enabling the learning model to focus on relevant facial regions and subtle inconsistencies introduced by synthetic generation techniques.

4.2. Stage 2: Feature Extraction

In this stage, we focus on extracting meaningful features from the preprocessed face crops to leverage the different characteristics of real and manipulated faces. We utilize a **Multi-Stream Input** approach, where each stream captures a distinct aspect of the face, thereby enhancing the model's ability to detect subtle differences in manipulated videos. The extracted features will be used to train a deep learning model capable of distinguishing between real and deepfake images.

Multi-Stream Input

We employ three parallel input streams for each face crop, each designed to capture different information about the face. These streams are as follows:

1. **RGB Image:** The traditional and most commonly used stream. It captures the texture, color, and appearance of the face. The RGB stream focuses on pixel-level information that allows the model to learn discriminative features related to the subject's identity, lighting, and visual characteristics.

2. **Frequency Domain Representation:** In this stream, we apply a **Discrete Cosine Transform (DCT)** or **Fast Fourier Transform (FFT)** to the image, converting the spatial domain representation into the frequency domain. Manipulated faces often exhibit abnormal high-frequency artifacts due to synthesis methods like warping, inpainting, or blending. By focusing on high-frequency components, this stream is sensitive to these manipulations, providing an additional dimension for detecting deepfakes.

3. **Facial Motion Residuals:** This stream aims to capture the temporal dynamics between frames. Using either the **difference between consecutive frames** or **optical flow** methods, we estimate the motion between frames. Deepfake videos often exhibit unnatural or inconsistent motion transitions, such as sudden changes in facial expressions or irregular eye movements. By focusing on motion residuals, this stream helps detect these anomalies.

Why Multi-Stream?

Each of the three streams captures a different aspect of the face that may be useful in distinguishing between real and synthetic content:

- **RGB Stream:** Raw pixel values in the spatial domain are processed using an EfficientNet-B0 backbone. To enhance feature representation, a Convolutional Block Attention Module (CBAM) is applied, enabling the network to focus on key facial cues such as texture anomalies at boundaries, lighting inconsistencies in synthetic regions, and color irregularities from blending artifacts.

- **Frequency Stream:** High-frequency manipulations often indicate the presence of artifacts from the face generation process. These include pixel-level inconsistencies or unnatural textures that are typically absent in real-world data but are common in deepfake-generated faces.
  Employs Fast Fourier Transform (FFT) to reveal manipulation artifacts through:

$$F_I(u,v) = \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} I(x,y)\, e^{-i2\pi\left(\frac{ux}{W} + \frac{vy}{H}\right)} \tag{1}$$

- **Motion Stream:** Deepfakes tend to exhibit abnormal facial motion due to imperfect generation, such as mismatched blinking, unusual lip-syncing, or unnatural facial muscle transitions. By analyzing the residual motion or optical flow, this stream can detect these discrepancies.
  Computes absolute frame differences to capture temporal inconsistencies:

$$M_t = |I_t - I_{t-1}| \tag{2}$$

Feature Fusion

The outputs from each of the three streams are later fused, either through concatenation or more advanced fusion techniques, to form a comprehensive feature representation. These multi-faceted features enable the model to distinguish between the natural variations in real faces and the artifacts introduced by deepfake generation techniques.

By employing this multi-stream architecture, we aim to increase the robustness and accuracy of the model by combining complementary features from multiple domains: appearance, frequency, and motion. This approach is well-suited for tackling the complex nature of deepfake detection.

4.3. Stage 3: Model Architecture

The proposed model architecture is based on a **Multi-Stream CNN with Convolutional Block Attention Module** strategy. This design enables efficient feature extraction and fusion from multiple data streams (e.g., RGB, frequency, and motion) to perform deepfake detection.

Per-Stream Backbone

Each stream uses lightweight CNN backbone for efficient feature extraction:

- **EfficientNet-B0**: Chosen for its balance between accuracy and computational efficiency.

Fusion Strategy

The fusion mechanism involves the following steps:

1. **Channel and Spatial Attention:** Applied within each stream to highlight important features.
2. **Transformer-based Cross-Stream Attention:** Enables learning the relationships between different streams (e.g., RGB, frequency, and motion).
3. **Feature Concatenation:** The attended features from each stream are concatenated to form a comprehensive representation.
4. **Classification Head:** A multi-layer perceptron (MLP) followed by a Sigmoid activation function for binary classification.

4.4. Model Pipeline

The proposed model pipeline follows a multi-stream architecture with feature extraction, attention, fusion, and classification stages:

- **Input:** $[\text{RGB}, \text{Frequency}, \text{Motion Residuals}] \rightarrow 3$ `EfficientNets`
- **Per-Stream Attention:** Apply attention mechanisms (CBAM) to each stream
- **Fusion Layer:** Transformer-based Cross-Attention for combining features from each stream
- **Fully Connected Layers:** Followed by dropout to prevent overfitting
- **Output:** Deepfake (1) or Real (0)

l0.6

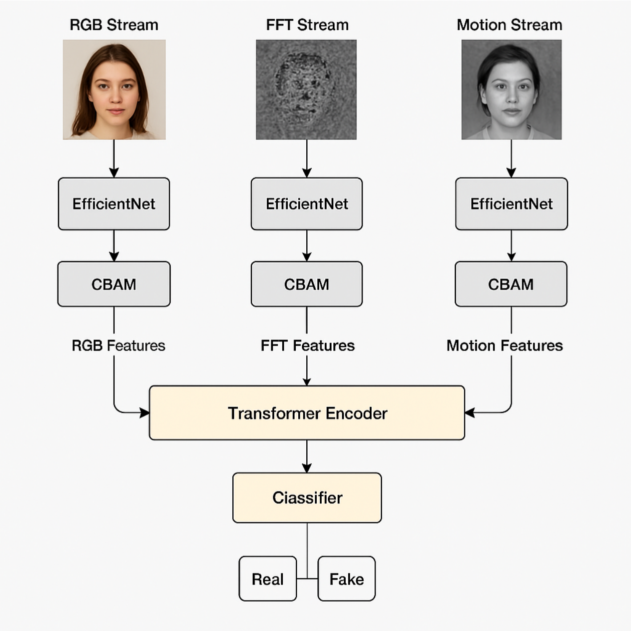


Figure 1. Model Pipeline.

4.5. Stage 4: Training Setup

The training setup includes the choice of loss function, validation strategy, optimizer, and data augmentation techniques aimed at improving the model's performance and robustness.

Loss Function

The binary classification task (Deepfake vs. Real) is addressed using:

- **Binary Cross-Entropy Loss:** The standard loss function for binary classification.

$$\mathcal{L}_{\text{BCE}} = -\left[y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})\right]$$

Validation Strategy

- **hold-out validation strategy:** 80% data is used for training and 20% is hold-out for validation.

Optimizer

- **AdamW Optimizer:** Chosen for its effectiveness in training deep networks with weight decay regularization.

4.6. Stage 5: Post-Processing

In this stage, predictions from multiple frames of a video are aggregated to produce the final classification result.

The aggregation methods help improve prediction stability and robustness by considering temporal information across frames.

Aggregation Methods

- **Average:** The average of the frame-wise predictions is computed to obtain a final decision. This method smooths out fluctuations in individual frame predictions.

4.7. Stage 6: Evaluation Metrics

To comprehensively evaluate the performance of the proposed deepfake detection model, we utilize the following standard classification metrics:

- **Accuracy:** Measures the overall correctness of predictions.
- **Precision, Recall, F1-Score:** Precision quantifies exactness, recall measures completeness, and F1-score provides a harmonic mean of both.
- **AUC-ROC:** The Area Under the Receiver Operating Characteristic Curve evaluates the model's ability to distinguish between classes at various thresholds.
- **Confusion Matrix:** Provides a detailed breakdown of true positives, false positives, true negatives, and false negatives.

5. Test Results and Summary

The following table summarizes the capabilities of existing deepfake detection methods in comparison with our proposed approach:

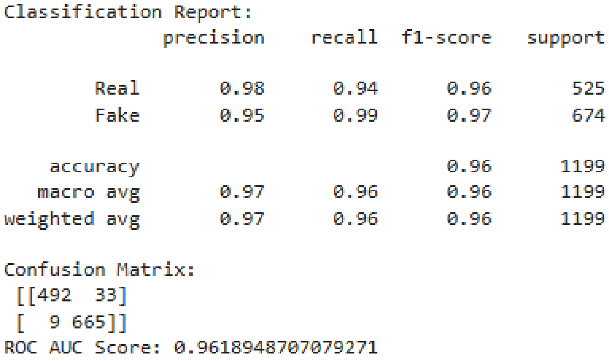
Compared to prior methods, the **proposed approach** demonstrates a more comprehensive and advanced design. While all baseline models—Face X-Ray, Two-Stream, MesoNet, and F3-Net—utilize RGB input (1), only Two-Stream and F3-Net incorporate frequency-domain information: FFT-based and wavelet-based respectively. In contrast, the proposed method supports both DCT and FFT (1).

Dataset	Accuracy
FaceForensics	91%
DFD	94%
CelebDf V2	96%

Table 3. Dataset accuracy comparison.

Unlike existing models, which do not leverage motion residuals, our approach explicitly integrates this temporal signal. We introduce a richer *three-stream* architecture, compared to the basic two-stream setup of previous models.

l0.6



```
Classification Report:
              precision    recall  f1-score   support

        Real       0.98      0.94      0.96       525
        Fake       0.95      0.99      0.97       674

    accuracy                           0.96      1199
   macro avg       0.97      0.96      0.96      1199
weighted avg       0.97      0.96      0.96      1199

Confusion Matrix:
 [[492  33]
 [  9 665]]
ROC AUC Score: 0.9618948707079271
```

Figure 2. Classification report for celebV2

Our method uses lightweight backbones like EfficientNet-B0 and MobileNetV3 for improved deployability on edge devices, a feature only MesoNet considered.

Additionally, while F3-Net uses simple feature reweighting, we combine CBAM/SE with cross-stream attention, transformer-based fusion, and per-frame attention aggregation—innovations not seen in prior work. Lastly, our model uniquely combines *RGB*, frequency, and motion modalities.

5.1. Appendix

- **Trained Model**: DeepFake Detection Model
- **Implementation Code**: DeepFake Video Recognition Model Code
- **Datasets**:
  - CelebDFV2 Dataset
  - DeepFake Entire Original Dataset
  - FaceForensics++ Dataset

References

1. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S. (2020). *Face X-Ray for More General Face Forgery Detection.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://openaccess.thecvf.com/...

2. Lu, X., Zhou, S., Lin, Z., Zhou, A., Wang, W. (2023). *Locate and Verify: A Two-Stream Network for Improved Deepfake Detection.* In Proceedings of the ACM MM. https://dl.acm.org/doi/...

3. Afchar, D., Nozick, V., Yamagishi, J., Echizen, I. (2018). *MesoNet: A Compact Facial Video Forgery*

Detection Network. arXiv preprint arXiv:1809.00888. https://ar5iv.labs.arxiv.org/...

4. Qian, Y., Jin, H., Wang, X., Tang, Y. (2021). *F3-Net: Frequency-Aware Neural Network for Face Forgery Detection*. In Proceedings of the AAAI Conference on Artificial Intelligence. https://ojs.aaai.org/...

5. Yao, D., Pan, W., Zhou, Q., Xu, Y. (2024). *TSFF-Net: Two-Stream Feature Domain Fusion Network for Deepfake Detection*. PLOS ONE, 19(2). https://journals.plos.org/...

6. Jiang, T., Wang, H., Xu, W., Zhang, J. (2024). *FADE: Deepfake Video Detection via Facial Action Dependencies Estimation*. In Proceedings of the AAAI Conference on Artificial Intelligence https://ojs.aaai.org/...