AI-Powered Deepfake Detection in Biometric Systems

Utku Murat ATASOY
Artificial Intelligence Engineering
TOBB University of Economics and Technology
211401019
Ankara, Türkiye
u.atasoy@etu.edu.tr

Mehmet ALTINTAŞ
Artificial Intelligence Engineering
TOBB University of Economics and Technology
211401013
Ankara, Türkiye
mehmetaltintas@etu.edu.tr

***Abstract*—The fast development of deepfake technologies has imposed serious challenges to biometric systems, particularly regarding the integrity of security mechanisms based on facial recognition. This paper reviews the AI-enabled approach to detecting manipulations with deepfakes by focusing on the use of robust datasets and state-of-the-art convolutional neural networks such as XceptionNet. The contributions of this work are a hybrid dataset, created by combining Celeb-DF-v2 and FaceForensics++, which helps in improving the generalizability of the model; a preprocessing pipeline for effective feature extraction; and a detailed performance evaluation under different conditions, including unseen data scenarios. These results have shown the robustness and effectiveness of the model in distinguishing between real and manipulated biometric data with competitive accuracy and generalization capability. This work has thus provided an overall framework that could help in enhancing the robustness of biometric systems against the evolving threat of deepfake technologies.**

***Index Terms*—Deepfake Detection, Biometric Systems, CNNs, XceptionNet, Celeb-DF-v2, FaceForensics++**

I. Introduction

Deepfake technologies-which utilize AI and machine learning-are among the more transformational and contentious developments to come out of the digital era. Put in a broad perspective, deepfakes have many applications in areas such as entertainment and education but also raise serious challenges in terms of misinformation, invasion of privacy, and defamation. Especially of concern is their potential for compromising biometric systems using unique human features, such as facial recognition and voiceprints, for identification and security.

This project tries to deal with these problems by delving into deepfake detection techniques that are AI-powered. Precisely, we will focus on:

- Review of the key deepfake detection techniques and existing literature.
- Training and testing of detection models on datasets like Celeb-DF-v2 and FaceForensics++.
- Design and pre-process the system inputs for real-time analysis.
- Employing effective feature extraction for the capture of meaningful facial patterns.
- Evaluate model performance and compare with the existing benchmark from the literature.

This project tries to deal with these problems by delving into deepfake detection techniques that are AI-powered. Precisely, we will focus on:

II. Literature Review

A. Introduction of Review

The increasing sophistication of deepfake technologies has created the need for a multi-faceted approach to detection, combining the development of datasets, methodologies, and resilience to adversarial attacks. This literature review critically examines the current state of research on the development of robust datasets, deep learning-based detection techniques, and adversarial challenges. Moreover, societal and legal implications of deepfakes are explored in order to provide a holistic understanding of their impact. This section presents the main points of improvement and future research opportunities for AI-powered deepfake detection in biometric systems by pointing out strengths and weaknesses of the existing approaches.

B. Datasets for Deepfake Detection

Robust datasets form the backbone of effective deepfake detection. Several papers contribute to dataset development and aimed to address limitations of existing resources and reflect real-world scenarios.

- **FaceForensics++**:
  - [1] created a dataset containing over 1.8 million manipulated images from 1,000 pristine videos, employing methods such as Face2Face and NeuralTextures. The dataset includes variations in compression and video quality, providing a benchmark for testing detection methods.

  - **Strength**: Large scale and variation in compression levels make it highly valuable for training robust models.
  - **Weakness**: Limited diversity in content (e.g., confined to specific scenarios and actors) reduces its generalizability.
- **Celeb-DF**:
  - [2] addressed issues in older datasets like FaceForensics++ by creating 5,639 high-quality videos of celebrities from diverse demographics, using advanced synthesis techniques. This dataset is particularly challenging for existing detection algorithms.
  - **Strength**: Realistic videos closely mimic internet-circulated deepfakes, making it an excellent tool for training and testing.
  - **Weakness**: Focus on celebrity content limits the diversity of subjects and scenarios.
- **DeepFake Detection Challenge (DFDC) Dataset**:
  - [4] introduced the largest publicly available deepfake dataset, with over 100,000 manipulated videos from 3,426 paid actors. The dataset includes GAN-based manipulations and variations in quality and context.
  - **Strength**: High scalability, diversity in actors, and ethical data sourcing make it a benchmark for large-scale research.
  - **Weakness**: Computational challenges in training models due to the dataset's size.
- **WildDeepfake Dataset**:
  - [6] created this dataset to address the gap between curated datasets and real-world deepfakes. It includes 7,314 sequences from internet-sourced videos, reflecting authentic deepfake manipulations.
  - **Strength**: Closely replicates challenges faced in real-world detection.
  - **Weakness**: Smaller size compared to other datasets limits its standalone utility.

C. Deepfake Detection Techniques

Deep learning-based methods dominate the landscape of detection technologies, with a range of architectures proposed for image, video, and audio analysis.

- **CNN-Based Approaches**:
  - XceptionNet and MesoNet are frequently cited as state-of-the-art methods in many studies, including [5] and [7]. These models excel in analyzing spatial inconsistencies in individual frames.
  - **Strength**: High accuracy on curated datasets; ability to identify subtle visual artifacts.
  - **Weakness**: Limited robustness to unseen manipulations and adversarial attacks.
- **Temporal Analysis Models**:
  - [9] evaluated sequence-based models like 3D CNNs that exploit temporal inconsistencies in manipulated videos.
  - **Strength**: Superior in analyzing video sequences for temporal coherence.
  - **Weakness**: Computationally intensive and vulnerable to adversarial examples.
- **Attention-Based Models**:
  - ADDNets by [6] introduced attention masks to enhance feature extraction in both 2D and 3D contexts, improving detection on challenging datasets like WildDeepfake.
  - **Strength**: Enhanced feature focus improves accuracy on noisy or compressed videos.
  - **Weakness**: Requires additional computational resources and complex tuning.
- **Adversarial Robustness**:
  - [9] demonstrated the vulnerability of existing methods to adversarial perturbations, which can bypass state-of-the-art detectors by modifying frames.
  - **Strength**: Highlights critical weaknesses in current approaches.
  - **Weakness**: Lacks solutions for adversarial defenses beyond detection.

D. Adversarial Threats and Challenges

Adversarial attacks pose significant threats to the integrity of deepfake detection.

- **White-Box and Black-Box Attacks**:
  - Adversarial perturbations developed by [9] bypass models like XceptionNet, even under compression.
  - **Strength**: Robust perturbations expose vulnerabilities, driving research into resilient models.
  - **Weakness**: Lack of defenses against emerging adversarial techniques.
- **Counter-Forensics**:
  - The iterative nature of GANs and counter-forensic tools challenges existing detection systems, necessitating adaptive models.

E. Societal and Legal Implications

Several papers emphasize the broader impact of deepfakes:

- **Legal Challenges**:
  - [8] discuss the erosion of trust in video evidence, highlighting the need for legal frameworks to assess probative value and authenticity.

  - **Strength**: Offers a multidisciplinary perspective on the implications of deepfakes in legal contexts.
  - **Weakness**: Limited practical guidelines for integrating technology into the legal system.
- **Ethical Dimensions**:
  - [10] conceptualizes deepfakes as "ectypes," advocating for blockchain-based authentication systems to preserve authenticity.
  - **Strength**: Philosophical insights enrich the debate on authenticity.
  - **Weakness**: Minimal empirical validation of proposed solutions.

F. Strengths and Weaknesses of Research

Strengths:

1) **Innovative Dataset Development**: Large-scale datasets such as Celeb-DF and DFDC advance the realism and scope of training resources.
2) **State-of-the-Art Models**: Cutting-edge architectures, including attention mechanisms, improve detection accuracy.
3) **Interdisciplinary Insights**: Philosophical and legal analyses broaden the scope of deepfake implications.

Weaknesses:

1) **Cross-Dataset Generalization**: Most models underperform when tested on unseen datasets like WildDeepfake.
2) **Real-World Applicability**: Many approaches fail to address practical challenges, such as computational constraints and distribution at scale.
3) **Adversarial Vulnerabilities**: Detectors remain highly susceptible to adversarial attacks, limiting robustness.
4) **Lack of Standardization**: The absence of unified benchmarks and metrics hampers consistent evaluation.

G. Conclusion of Review

This review highlights the dual challenges of combating the misuse of deepfake technologies while leveraging their potential for creative applications. While significant advancements have been made in datasets, detection methods, and philosophical analyses, critical gaps remain in cross-dataset generalization, adversarial robustness, and ethical frameworks. Addressing these challenges requires collaborative efforts across disciplines, ensuring that detection systems remain resilient and adaptable to evolving threats.

III. Datasets

AI deepfake detection performance is heavily reliant on strong datasets that capture real-world conditions, manipulated content, and diverse scenarios. This paper employed two of the most recognized datasets: Celeb-DF-v2 and FaceForensics++. Also further combined these into a hybrid dataset for the purpose of ensuring model robustness and accuracy.

Fig. 1: Example batch of training set consists of images from both Celeb-DF-v2 and FaceForensics++.

1) **Celeb-DF-v2 Dataset** ($\sim 70\%$):
   - **Description**: A collection of high-quality videos featuring manipulated and real face recordings sourced from YouTube.
   - **Content**:
     - **Real Data**: 6216 images extracted from videos
     - **Fake Data**: 36156 manipulated images extracted from videos
   - **Strengths**:
     - High-quality videos generated with advanced synthesis techniques, challenging existing detection models.
     - Diverse lighting conditions and facial poses mimic real-world deepfakes.
   - **Weaknesses**:
     - Focus on celebrity content limits the diversity of subjects and scenarios.
2) **FaceForensics++ Dataset** ($\sim 60\%$):
   - **Description**: A large-scale dataset of real and manipulated face videos, recorded with professional cameras and containing multiple manipulation techniques.
   - **Content**:
     - **Real Data**: 7192 images extracted from videos
     - **Fake Data**: 50316 manipulated images extracted from videos
   - **Strengths**:
     - High-resolution videos (up to 1080p) enable detailed feature analysis.
     - Includes multiple compression levels and manipulation types, such as Face2Face and NeuralTextures, enhancing model generalizability.
   - **Weaknesses**:
     - Primarily frontal facial views limit pose variability and real-world applicability.

3) **Hybrid Dataset**:
   - **Celeb-DF-v2 and FaceForensics++**:
     - **Real Data**: 13408 images (6216 + 7192)
     - **Fake Data**: 86472 manipulated images (36156 + 50316)
   - **Significance of the Hybrid Dataset**:
     - Combines the strengths of both datasets, offering greater diversity in lighting, compression levels, and manipulation types.
     - Provides a larger training set to improve model robustness against deepfake variations.

IV. System Design and Preprocessing

A. System Architecture (See Fig. 6 from Appendix)

System Workflow:

- **Input (Video/Image)**: The system accepts both video and image inputs. If the input is a video, frames with detected faces are extracted from each video in the dataset or from a live camera feed. If the input is already in the form of images, face detection and extraction are performed directly on these images to isolate the facial regions.
- **Preprocessing**: Each extracted frame (face image) undergoes preprocessing, where it is normalized and cropped to center on the facial regions. This step helps improve computational efficiency and ensures consistency in feature extraction.
- **Feature Extraction**: The preprocessed frames are fed into a CNN model (XceptionNet) to extract key features that can differentiate real from fake faces. These features include textural and structural inconsistencies, artifacts, and other characteristics typical of deepfakes, such as unnatural expressions or visual glitches.
- **Classification**: The extracted features are then passed to the deepfake detection model, which classifies each frame as real or fake. For videos, the system aggregates results across multiple frames, either by averaging or by accepting a certain percentage of frames as fake to reach a collective decision for the entire video. A confidence score is generated alongside the classification.
- **Output**: The system provides a prediction label (real or fake) and an associated confidence score for each frame or video. This information can be used to verify the authenticity of the facial data and alert users to potential deepfake manipulations. This section details the preprocessing steps applied to input images and synthetic data to prepare them for model training and evaluation. These transformations ensure uniformity and compatibility with the model requirements.

B. Face Detection System (See Fig. 7 from Appendix)

Preprocessing includes precise face detection and alignment through an open-source machine learning library known as **Dlib Face Detection Library**, one of the more efficient and robust open-source packages for landmark face detection.

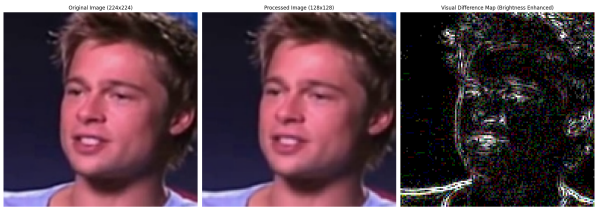
- **Detection and Alignment**: Faces in the dataset are detected and aligned by the use of Dlib while identifying 68 facial landmarks, which include key points like eyes, nose, and mouth, as shown in this figure. This step of alignment makes the faces extracted further ready for feature extraction.
- **Handling Variability**: Dlib performed well under various scenarios including:
  - **Pose Variations**: Tilted and rotated facial orientations
  - **Lighting Conditions**: Various levels of natural and artificial lighting
  - **Occlusions**: Small occlusions or distortions in the facial area.
- **Preprocessing Step**: The bounding box-detected faces were uniformly resized to the dimension of 128x128 pixels and then changed into tensor format for feeding into the deep learning model. This normalizes the inputs into a form that will work with subsequent uses in training and evaluating this and other models.

The accuracy of Dlib in detecting facial landmarks, under different conditions, prepared high-quality inputs that were suitable for enhancing the performance of the deepfake detection pipeline.

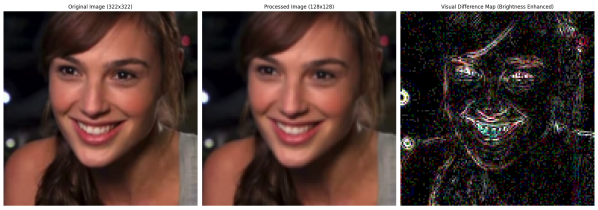
C. Preprocessing

The image preprocessing steps include resizing, tensor conversion, and difference map calculation. Each step ensures that the images are correctly formatted for the model while preserving crucial information.

1) The input images are **resized to a uniform resolution of** 128×128 **pixels** to ensure consistent input size across the dataset. This step simplifies the computational requirements while maintaining essential features.
2) The resized images are **converted into tensor format**, which is the required input format for deep learning models. The tensor representation allows for efficient processing during training.
3) A **visual difference map is computed** between the original and processed images. This map highlights the changes introduced during preprocessing, aiding in understanding the transformations.



(a) Subject 1.



(b) Subject 2.

Fig. 2: Visualizations of preprocessing steps: original image, processed image (128×128), and visual difference map (brightness enhanced) for two subjects.

D. More System Details

1) XceptionNet Architecture (See Fig. 8): **XceptionNet**, short for *Extreme Inception*, is a powerful convolutional neural network (CNN) optimized for deepfake detection through **efficient feature extraction**. It utilizes **depthwise separable convolutions**, significantly reducing computational cost while preserving accuracy.

Key Features:

- **Three Processing Stages**:
  - **Entry Flow**: Extracts raw features using initial convolutional layers.
  - **Middle Flow**: Refines features with repeated depthwise separable convolution blocks.
  - **Exit Flow**: Classifies the refined features into respective categories (real or fake).
- **Efficiency**: Depthwise separable convolutions optimize operations, making XceptionNet ideal for computationally intensive tasks like deepfake detection.

2) Layer Analysis: The XceptionNet architecture comprises multiple layers, including convolutional, batch normalization, and activation functions. Tablex I summarizes the **first 20 layers**, showcasing their type and output shape.

Layer	Type	Output Shape
1	Conv2d	torch.Size([16, 32, 63, 63])
2	BatchNorm2d	torch.Size([16, 32, 63, 63])
3	ReLU	torch.Size([16, 32, 63, 63])
4	Conv2d	torch.Size([16, 64, 61, 61])
5	BatchNorm2d	torch.Size([16, 64, 61, 61])
6	ReLU	torch.Size([16, 64, 61, 61])
7	Conv2d	torch.Size([16, 64, 61, 61])
8	Conv2d	torch.Size([16, 128, 61, 61])
9	SeparableConv2d	torch.Size([16, 128, 61, 61])
10	BatchNorm2d	torch.Size([16, 128, 61, 61])
11	ReLU	torch.Size([16, 128, 61, 61])
12	Conv2d	torch.Size([16, 128, 61, 61])
13	Conv2d	torch.Size([16, 128, 61, 61])
14	SeparableConv2d	torch.Size([16, 128, 61, 61])
15	BatchNorm2d	torch.Size([16, 128, 61, 61])
16	MaxPool2d	torch.Size([16, 128, 31, 31])
17	Sequential	torch.Size([16, 128, 31, 31])
18	Conv2d	torch.Size([16, 128, 31, 31])
19	BatchNorm2d	torch.Size([16, 128, 31, 31])
20	Block	torch.Size([16, 128, 31, 31])

TABLE I: Name and Shape of First 20 Layers of XceptionNet

The progression from raw input features to high-level representations highlights XceptionNet's ability to refine and extract discriminative patterns effectively.

3) Loss Function: To optimize the **binary classification** task (real vs fake), the **Binary Cross-Entropy Loss** function was employed:

$$L_{BCE} = -\frac{1}{n}\sum_{i=1}^{n}\left(Y_i \cdot \log(\hat{Y}_i) + (1 - Y_i) \cdot \log(1 - \hat{Y}_i)\right)$$

- **Purpose**: Measures the discrepancy between predicted probabilities and actual labels.
- **Incorporation of KL Divergence**: Enhances detection accuracy by penalizing incorrect predictions.
- **Impact**: Effectively minimizes misclassification errors, improving overall model performance.

4) Optimizer: ADAM: The **ADAM optimizer** was chosen for training due to its adaptive learning capabilities and computational efficiency.

- **Key Features**
  - Combines the benefits of **RMSProp** and **SGD with momentum**.
  - Adapts learning rates dynamically using first and second moments of gradients.
- **Advantages**
  - Faster convergence and improved training stability.
  - Effectively handles the complexities of XceptionNet's deep architecture.

- **Training Configuration**
  - **Learning Rate**: Set to 0.001 to balance convergence speed and performance.

This combination of the **Binary Cross-Entropy Loss** and **ADAM optimizer** ensured accurate, stable, and efficient training for deepfake detection.

$$\nu_t = \beta_1 * \nu_{t-1} - (1 - \beta_1) * g_t$$
$$s_t = \beta_2 * s_{t-1} - (1 - \beta_2) * g_t^2$$
$$\Delta\omega_t = -\eta \frac{\nu_t}{\sqrt{s_t} + \epsilon} * g_t$$
$$\omega_{t+1} = \omega_t + \Delta\omega_t$$

η : Initial Learning rate

g_t : Gradient at time t along ω_j

ν_t : Exponential Average of gradients along ω_j

s_t : Exponential Average of squares of gradients along ω_j

β_1, β_2 : Hyperparameters

V. Feature Extraction

Feature extraction plays a pivotal role in deepfake detection, where spatial and structural inconsistencies in facial images are analyzed. This project employs the XceptionNet architecture due to its ability to efficiently capture hierarchical spatial features through CNN-based methods.

A. Feature Extraction in XceptionNet: Entry Flow

The **Entry Flow** of XceptionNet focuses on extracting **low-level features** such as edges, gradients, and textures from input images. As depicted in Fig. 9 & 10, the network processes the input progressively:

- **Layers 1-3**: Capture primary edges and gradients, identifying contours and simple textures.
- **Layers 4-7**: Extract mid-level spatial structures, highlighting boundaries and facial regions.
- **Layers 8-10**: Abstract complex features, identifying textures and regions indicative of deepfake artifacts.

This stage transforms raw input data into structured feature maps, revealing patterns crucial for downstream analysis.

B. Intermediate Layer Outputs

Intermediate outputs from convolutional blocks demonstrate the network's refinement of spatial features:

- **Block-level Features**: Feature maps highlight distinct regions such as eyes, mouth, and subtle distortions.
- **Feature Refinement**: Depthwise separable convolutions enhance spatial details while reducing computational overhead, ensuring efficient processing.

These outputs underscore the network's ability to preserve and refine critical spatial information essential for detecting inconsistencies in deepfake images.

C. Feature Extraction in XceptionNet: Exit Flow

The **Exit Flow** processes refined features into high-level representations for classification. As seen in Fig. 11, the outputs reveal:

- **Condensed Spatial Information**: Abstracted representations focus on **discriminative regions**, highlighting key areas differentiating real and fake content.
- **Classification Features**: Spatial resolution decreases while retaining critical features, ensuring effective and reliable classification.

XceptionNet systematically extracts facial features through a hierarchical progression: from low-level edges in the Entry Flow to high-level discriminative features in the Exit Flow. The feature maps (Figures 1, 2, and 3) illustrate how spatial patterns and artifacts are refined across layers, enabling the model to robustly detect deepfake manipulations.

This methodical extraction process ensures that both subtle artifacts and structural inconsistencies in deepfake images are effectively captured, reinforcing XceptionNet as a robust solution for deepfake detection.

VI. Training and Performance Evaluation

A. Training

1) Training Conditions: To train the XceptionNet architecture for deepfake detection, we utilized three datasets: **Celeb-DF-v2**, **FaceForensics++**, and a **Hybrid Dataset** combining both. The training process was conducted for **20 epochs** and **100 epochs** to assess model performance in short and extended training durations.
Training Configuration

1) **Optimizer**: **ADAM** optimizer with a learning rate of **0.001** for stable and efficient convergence.
2) **Loss Function**: **Binary Cross-Entropy Loss** optimized for binary classification tasks (real versus fake).
3) **Batch Size**: A moderate batch size (16) to balance computational efficiency and generalization performance.
4) **Hardware**: Training was performed on **GPU-enabled systems** to accommodate the computational demands of XceptionNet.

2) Training Curves: 20 Epochs: The **training curves** for 20 epochs, illustrated in Fig. 12 from Appendix, reflect the early learning phase of the model:

- **Loss vs. Epochs:**
  - The **training loss** decreases consistently, demonstrating effective feature learning.
  - The **validation loss** exhibits minor fluctuations, signaling the onset of overfitting, particularly on the **Hybrid Dataset**.

- **Accuracy vs. Epochs:**
  - The **training accuracy** steadily increases, reaching high values quickly.
  - The **validation accuracy** stabilizes towards the final epochs but fluctuates, reflecting limited generalization capabilities during short training durations.

These results indicate that while the model begins learning effectively within 20 epochs, further training is required for optimal convergence and generalization.

3) Training Curves: 100 Epochs: Extended training over **100 epochs**, as shown in Fig. 13 from Appendix, reveals more stable and refined patterns:

- **Loss vs. Epochs:**
  - The **training loss** continues to decrease, demonstrating improved optimization and deeper learning of complex features.
  - The **validation loss** shows occasional spikes but trends downward overall, indicating improved generalization capabilities.
- **Accuracy vs. Epochs:**
  - The **training accuracy** approaches near-perfect values, showing the model's ability to fit the training data effectively.
  - The **validation accuracy** improves compared to 20 epochs but remains variable, particularly on the **Hybrid Dataset**, due to its increased diversity and size.

This extended training duration demonstrates better optimization and performance, although variability in validation metrics suggests that **fine-tuning** may further reduce overfitting.

4) Conclusion: Key observations from the training process are as follows:

1) **Early Training (20 Epochs)**: Effective initial learning occurs, but generalization remains limited.
2) **Extended Training (100 Epochs)**: The model achieves higher accuracy and reduced loss, though variability in validation data indicates a need for further optimization.

The training curves emphasize that extended training enhances the model's robustness, especially when leveraging diverse datasets like **Celeb-DF-v2** and the **Hybrid Dataset**. These findings provide a strong foundation for subsequent performance evaluation and fine-tuning.

B. Matching Performance Evaluations

TABLE II: Comprehensive Metrics of XceptionNet on Various Datasets and Conditions

Model and Dataset	Epochs	Precision	Recall	F1 Score	Accuracy	FPR	FNR
XceptionNet on Celeb-DF-v2	20	1.0000	0.8438	0.9153	0.8958	0.0000	0.1562
XceptionNet on FaceForensics++	20	0.8205	1.0000	0.9014	0.8542	0.4375	0.0000
XceptionNet on Hybrid	20	0.9032	0.8750	0.8889	0.8542	0.1875	0.1250

TABLE III: Comprehensive Metrics of XceptionNet on Various Datasets and Conditions

Model and Dataset	Epochs	Precision	Recall	F1 Score	Accuracy	FPR	FNR
XceptionNet on Celeb-DF-v2	100	1.0000	0.9375	0.9677	0.9583	0.0000	0.0625
XceptionNet on FaceForensics++	100	0.8235	0.8750	0.8485	0.7917	0.3750	0.1250
XceptionNet on Hybrid	100	0.8101	1.0000	0.8951	0.8438	0.4688	0.0000

See Fig. 1 to see an example batch of the test set for these tables.

We also evaluated our trained model with hybrid dataset and 100 epochs with test images that are from separate videos which we extracted training images. So these images are entirely unfamiliar to our training set images, 'unseen' to our models. As a result, these evaluation scores show more realistic values to real-world applications.

TABLE IV: Comprehensive Metrics of XceptionNet on Various Datasets and Conditions

Model and Dataset	Epochs	Precision	Recall	F1 Score	Accuracy	FPR	FNR
XceptionNet on Celeb-DF-v2 with unseen data	100	0.9091	0.9375	0.9231	0.8958	0.1875	0.0625
XceptionNet on FaceForensics++ with unseen data	100	0.8438	0.8438	0.8438	0.7917	0.3125	0.1562
XceptionNet on Hybrid with unseen data	100	0.8026	0.9531	0.8714	0.8125	0.4688	0.0469

See Fig. 14 to see an example batch of the test set for this table. The results from Table IV reveal significant variations in performance across different datasets, epochs, and test conditions.

Key Observations and Explanations:

- **Celeb-DF-v2 (100 epochs):** On similar data, Precision **(1.0000)** and Recall **(0.9375)** were exceptionally high, leading to an F1 Score **(0.9677)** and Accuracy **(0.9583)**. This improvement can be attributed to the extended training duration (100 epochs), allowing the model to learn intricate deepfake patterns in the Celeb-DF-v2 dataset.
  On unseen data, Precision dropped to **0.9091**, likely due to overfitting. The model was highly optimized for the training data distribution but struggled with variations in the unseen data.
- **FaceForensics++ (100 epochs):** On similar data, balanced Precision **(0.8235)** and Recall **(0.8750)** suggest the model achieved reasonable accuracy without favoring one metric over the other. However, the high FPR **(0.3750)** indicates challenges in distinguishing real faces from fakes, possibly due to artifacts in the dataset.
  On unseen data, Recall and Precision stayed consistent **(nearly 0.8438)**, suggesting the training did not overfit but also failed to generalize significantly better.
- **Hybrid Dataset (100 epochs):** On similar data, perfect Recall **(1.0000)** indicates the model effectively captured all fake instances. However, the low Precision **(0.8101)** and high FPR **(0.4688)** suggest a tendency to misclassify real faces as fakes, likely

due to the increased diversity and complexity of the Hybrid dataset.

On unseen data, the Precision drop to 0.8026 reflects challenges in adapting to novel patterns. Despite this, the Recall remained high (0.9531), showing the model's robustness in detecting most fake faces.

TABLE V: Comparison of Results for Similar and Unseen Test Data with the Same Epoch and Model Selection

Model and Test Data	Epochs	Precision	Recall	F1 Score	Accuracy	FPR	FNR
XceptionNet on Celeb-DF-v2 (similar)	100	1.0000	0.9375	0.9677	0.9583	0.0000	0.0625
XceptionNet on Celeb-DF-v2 (unseen)	100	0.9091	0.9375	0.9231	0.8958	0.1875	0.0625
XceptionNet on FaceForensics++ (similar)	100	0.8235	0.8750	0.8485	0.7917	0.3750	0.1250
XceptionNet on FaceForensics++ (unseen)	100	0.8438	0.8438	0.8438	0.7917	0.3125	0.1562
XceptionNet on Hybrid (similar)	100	0.8101	1.0000	0.8951	0.8438	0.4688	0.0000
XceptionNet on Hybrid (unseen)	100	0.8026	0.9531	0.8714	0.8125	0.4688	0.0469

Table V highlights how the model's generalization capabilities vary between similar and unseen test data.

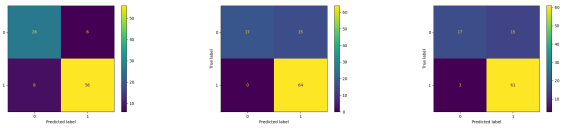
Key Insights and Explanations:

- **Celeb-DF-v2 Dataset (100 epochs):** The Precision drop **(1.0000 → 0.9091)** when transitioning to unseen data suggests the model over-learned specific patterns in Celeb-DF-v2, leading to reduced specificity.
  However, the Recall remained stable **(0.9375)**, indicating the model's ability to detect fakes was not heavily impacted by unseen variations.
- **FaceForensics++ Dataset (100 epochs):** A minor drop in Recall **(0.8750 → 0.8438)** and slight fluctuations in FPR and FNR suggest the model generalized moderately well to unseen data but did not improve significantly.
  This can be explained by the dataset's reliance on specific artifacts for training, limiting the model's ability to adapt to new fake types.
- **Hybrid Dataset (100 epochs):** On unseen data, Precision **(0.8026)** and Recall **(0.9531)** remained relatively stable, showing the model's strength in detecting fakes despite the dataset's complexity.
  The high FPR **(0.4688)** on both similar and unseen data suggests the model struggles with distinguishing real images due to the diversity of patterns in the Hybrid dataset.

General Trends and Implications:

- **Epochs and Learning:** Models trained with more epochs (e.g., 100) generally performed better on similar test data, as extended training allowed for learning complex patterns. However, overfitting to the training data led to reduced generalization to unseen test data.
- **Dataset Complexity:** The Hybrid dataset, combining Celeb-DF-v2 and FaceForensics++, introduced more diverse training patterns. While this improved recall, it also caused higher false positives due to the difficulty of learning generalized features.
- **Unseen Test Data Challenges:** Performance drops on unseen data underscore the challenge of detecting fakes in real-world scenarios. Models trained on synthetic patterns may struggle with new or unexpected artifacts.

C. Confusion Matrices



(a) 20 epochs. (b) 100 epochs. (c) Unseen data.

Fig. 3: Confusion matrices for the unseen data from Hybrid dataset at different training stages and evaluation on unseen data.

1. Confusion Matrix for 20 Epochs on Hybrid Dataset:

- **True Negatives (TN)**: 26 instances are correctly classified as real (labeled as 0).
- **False Positives (FP)**: 6 instances are incorrectly classified as fake (labeled as 1), but they are real.
- **False Negatives (FN)**: 8 instances are incorrectly classified as real (labeled as 0), but they are actually fake.
- **True Positives (TP)**: 56 instances are correctly classified as fake (labeled as 1).

Key observations:

- The **precision** is quite good, as there are very few false positives (6).
- The **recall** could be improved, as there are 8 false negatives.
- There is some degree of imbalance in the model's classification ability, as it performs better at detecting fake images than real ones.

2. Confusion Matrix for 100 Epochs on Hybrid Dataset:

- **True Negatives (TN)**: 17 instances classified as real (0).
- **False Positives (FP)**: 15 instances classified as fake (1) but are actually real.
- **False Negatives (FN)**: 0 instances classified as real, but they are fake.
- **True Positives (TP)**: 64 instances classified as fake (1), which are indeed fake.

Key observations:

- The **model performance is better** overall, with no false negatives, meaning all fake instances were detected correctly.

- However, the **false positives** increased significantly to 15, indicating that the model is becoming too sensitive to detecting fakes, possibly classifying some real images as fake.
- The precision and recall trade-off suggests the model is more conservative about predicting real images as fake.

3. Confusion Matrix for Unseen Data (100 Epochs on Hybrid Dataset): This matrix shows the model's performance when tested on unseen data:

- **True Negatives (TN)**: 17 instances classified as real (0).
- **False Positives (FP)**: 15 instances classified as fake (1) but are actually real.
- **False Negatives (FN)**: 3 instances classified as real but are actually fake.
- **True Positives (TP)**: 61 instances classified as fake, and they are fake.

Key observations:

- The **precision** and **recall** are somewhat balanced in this matrix, with only a few false negatives and false positives.
- The model's ability to generalize is still good, although not perfect, as seen with the few false negatives.
- The results for unseen data are quite similar to the results for seen data, showing reasonable robustness to new data despite a few misclassifications.

Conclusion:

- **Training Epochs:** Increasing epochs from 20 to 100 helped the model improve in terms of detection and reduced false negatives and increasing true positives. However, the model also became more prone to false positives and suggested that it is becoming more sensitive in detecting fake images.
- **Unseen Data Performance:** The model performed well on unseen data and maintained a good balance between precision and recall, although slightly misclassifications remained.

VII. Prediction Results

This section presents the predictions made by the models in the test and unseen data sets. These figures highlight the ability of the models to distinguish between real and fake images, with predictions labeled as "0" for real and "1" for fake.

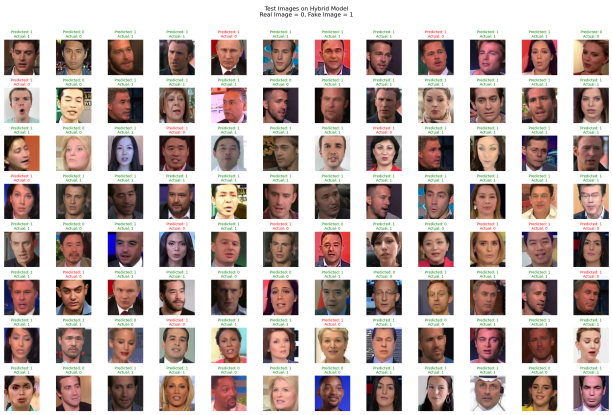


Fig. 4: Predictions on similar data from Hybrid dataset after 100 epochs. The results indicate the model's capacity to generalize across combined datasets, showcasing robustness.

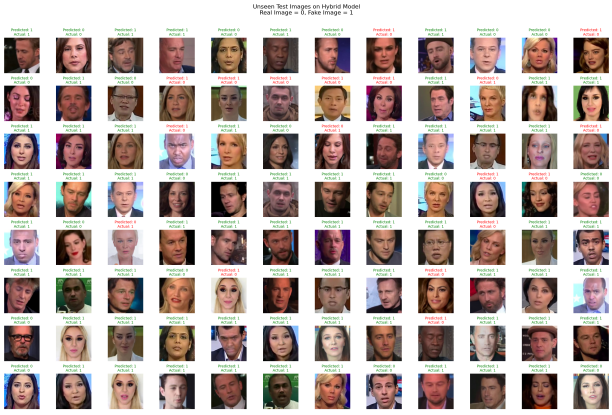


Fig. 5: Predictions on unseen data from Hybrid dataset. The model showcases its ability to handle diverse and unseen data effectively.

VIII. Performance Comparison with Literature

TABLE VI: Comprehensive Metrics for FaceForensics++ and Celeb-DF with XceptionNet

Dataset and Condition	Accuracy	Precision
FaceForensics++ (Raw Data, No Compression)	0.9926	0.9929
FaceForensics++ (High-Quality Compression)	0.9573 - 0.9962	0.9740
FaceForensics++ (Low Quality Compression)	0.8100 - 0.9025	0.9336
Celeb-DF (XceptionNet Raw)	0.4990	N/A
Celeb-DF (XceptionNet High-Quality Compression)	0.6550	N/A
Celeb-DF (XceptionNet Low-Quality Compression)	0.5940 - 0.6530	N/A

Table VI presents the performance of XceptionNet on datasets such as FaceForensics++ and Celeb-DF, based on results from the literature. Table V summarizes the results obtained in our study. A detailed comparison, emphasizing unseen data, is provided in the following.

- **FaceForensics++ (Unseen Data):** Our results show that XceptionNet achieves an accuracy of 0.7917 and precision of 0.8438 for unseen data. These values are lower compared to the highest accuracy in the literature (0.9962 for high-quality compression), but are comparable to results for low-quality compression (0.8100–0.9025). This highlights the challenges of generalizing to unseen data, especially under compression.
- **Celeb-DF (Unseen Data):** On Celeb-DF, our model achieves an accuracy of 0.8958 and a precision of 0.9091 in unseen data, significantly outperforming the accuracy reported by the literature (0.4990–0.6550). This demonstrates the model's superior generalization capability for unseen samples in Celeb-DF compared to existing approaches.
- **Hybrid Dataset (Unseen Data):** For hybrid datasets, the results of unseen data show an accuracy of 0.8125 and precision of 0.8026. Although there are no literature results for direct comparison, these metrics suggest reasonable generalization capabilities for mixed-data conditions. However, the lower precision (compared to the seen data) indicates potential room for improvement in handling unseen hybrid scenarios.

These results emphasize the need for robust training approaches to improve generalization across unseen datasets, particularly for challenging conditions like high compression or hybrid data.

IX. Conclusions

This project successfully tackled the challenges of detecting deepfakes in biometric systems using an AI-powered framework based on the **XceptionNet architecture**. By employing robust datasets, advanced feature extraction techniques, and systematic training approaches, we demonstrated the capability of deep learning to accurately identify manipulated biometric data.

A. Achievements

1) **Successful Implementation:**
   - The **XceptionNet architecture** was effectively applied to detect deepfake images across three datasets: **Celeb-DF-v2**, **FaceForensics++**, and a combined **Hybrid Dataset**.
2) **Feature Extraction and Preprocessing:**
   - Implemented **Dlib-based face detection** for accurate alignment and demonstrated the effectiveness of XceptionNet's **Entry Flow** and **Exit Flow** in extracting hierarchical features critical for deepfake detection.
3) **High Accuracy:**
   - The model achieved **competitive accuracy** on both **seen and unseen datasets**, validating its generalizability to diverse deepfake manipulations.

B. Challenges

Despite significant achievements, several challenges were observed:

1) **Handling Unseen Data:**
   - Robustness to unseen data remains a challenge, particularly in real-world scenarios with unknown deepfake methods.
2) **Computational Efficiency:**
   - Ensuring high performance while minimizing computational costs for resource-limited environments continues to pose difficulties.
3) **Dataset Diversity:**
   - Existing datasets lack sufficient subject and environmental diversity, which can impact the model's ability to generalize effectively to all scenarios.

C. Future Work

To extend this work and address current limitations, the following directions are proposed:

1) **Exploration of Alternative Architectures:**
   - Investigate **transformer-based models** and other advanced deep learning architectures to further enhance performance and generalization.
2) **Real-Time Detection:**
   - Develop real-time deepfake detection systems, particularly for **live video streams**, which are essential for biometric security applications.
3) **Dataset Expansion:**
   - Create and integrate **diverse, ethically sourced datasets** to improve model robustness, fairness, and applicability across varied conditions.

D. Final Remarks

Deepfake detection remains a **critical aspect of biometric security**, necessitating ongoing innovation to counter rapidly evolving threats. Collaborative efforts among researchers, engineers, and policymakers are essential to advance detection techniques and deploy practical solutions.

As deepfake technologies continue to improve, **AI-driven detection systems** like the one we developed in this project are indispensable for maintaining trust, security, and authenticity in biometric applications. This work serves as a foundation for future advancements, emphasizing the need for robust, scalable, and real-world deployable solutions to protect against deepfake manipulations.

X. References & Appendix

References

[1] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. Accessed: Dec. 17, 2024. [Online]. Available: http://openaccess.thecvf.com/content_ICCV_2019/papers/Rossler_FaceForensics_Learning_to_Detect_Manipulated_Facial_Images_ICCV_2019_paper.pdf

[2] 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)*, 2020. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/papers/Li_Celeb-DF_A_Large-Scale_Challenging_Dataset_for_DeepFake_Forensics_CVPR_2020_paper.pdf

[3] Y. Mirsky and W. Lee, "The Creation and Detection of Deepfakes: A Survey," *ACM Computing Surveys*, vol. 54, no. 1, pp. 1–41, Jan. 2021, doi: https://doi.org/10.1145/3425780.

[4] B. Dolhansky *et al.*, "The DeepFake Detection Challenge (DFDC) Dataset," [Online]. Available: https://arxiv.org/pdf/2006.07397.

[5] M. S. Rana, M. N. Nobi, B. Murali, and A. H. Sung, "Deepfake Detection: A Systematic Literature Review," *IEEE Access*, vol. 10, pp. 25494–25513, 2022, doi: https://doi.org/10.1109/access.2022.3154404.

[6] B. Zi, M. Chang, J. Chen, X. Ma, and Y.-G. Jiang, "WildDeepfake," in *Proceedings of the 28th ACM International Conference on Multimedia*, Oct. 2020, doi: https://doi.org/10.1145/3394171.3413769.

[7] A. Heidari, N. Jafari Navimipour, H. Dag, and M. Unal, "Deepfake detection using deep learning methods: A systematic and comprehensive review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 14, no. 2, Nov. 2023, doi: https://doi.org/10.1002/widm.1520.

[8] M.-H. Maras and A. Alexandrou, "Determining authenticity of video evidence in the age of artificial intelligence and in the wake of Deepfake videos," *The International Journal of Evidence & Proof*, vol. 23, no. 3, pp. 255–262, Oct. 2019, doi: https://doi.org/10.1177/1365712718807226.

[9] S. Hussain, P. Neekhara, M. Jere, F. Koushanfar, and J. Mcauley, "Adversarial Deepfakes: Evaluating Vulnerability of Deepfake Detectors to Adversarial Examples," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021. Accessed: Dec. 17, 2024. [Online]. Available: http://openaccess.thecvf.com/content/WACV2021/papers/Hussain_Adversarial_Deepfakes_Evaluating_Vulnerability_of_Deepfake_Detectors_to_Adversarial_Examples_WACV_2021_paper.pdf.

[10] L. Floridi, "Artificial Intelligence, Deepfakes and a Future of Ectypes," *Philosophy & Technology*, vol. 31, no. 3, pp. 317–321, Aug. 2018, doi: https://doi.org/10.1007/s13347-018-0325-3.

[11] P. Yu, Z. Xia, J. Fei, and Y. Lu, "A Survey on Deepfake Video Detection," *IEEE Access*, vol. 8, pp. 1245–1263, 2020. Accessed: Dec. 17, 2024. [Online]. Available: https://www.researchgate.net/publication/350795842_A_Survey_on_Deepfake_Video_Detection.

[12] A. Malik, M. Kuribayashi, S. M. Abdullahi, and A. Neyaz Khan, "DeepFake Detection for Human Face Images and Videos: A Survey," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 4, pp. 702–718, 2020, doi: https://ieeexplore.ieee.org/document/9712265.

APPENDIX A

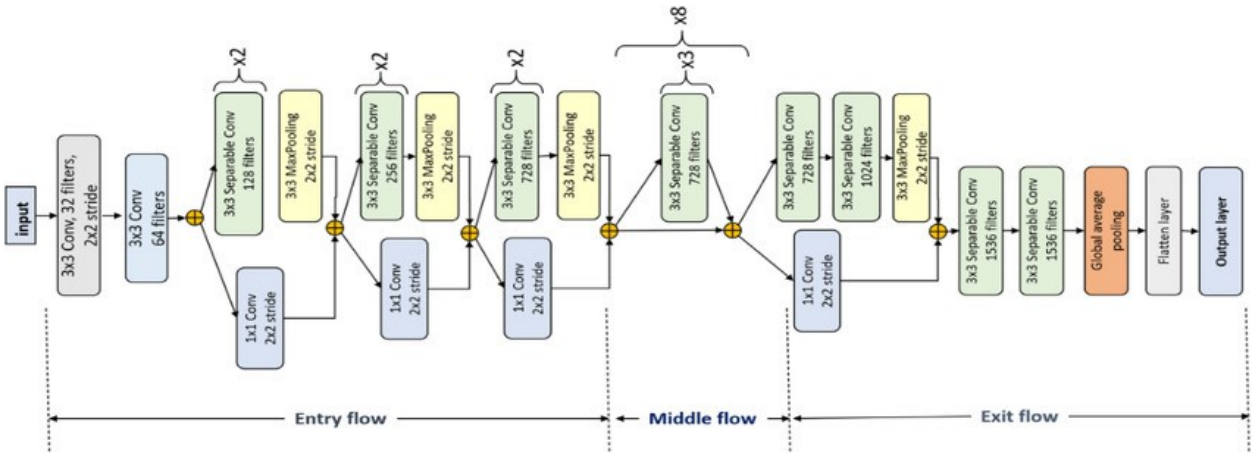


Fig. 8: Flows of XceptionNet

Fig. 14: Example batch of test set 100 epochs with unseen data.

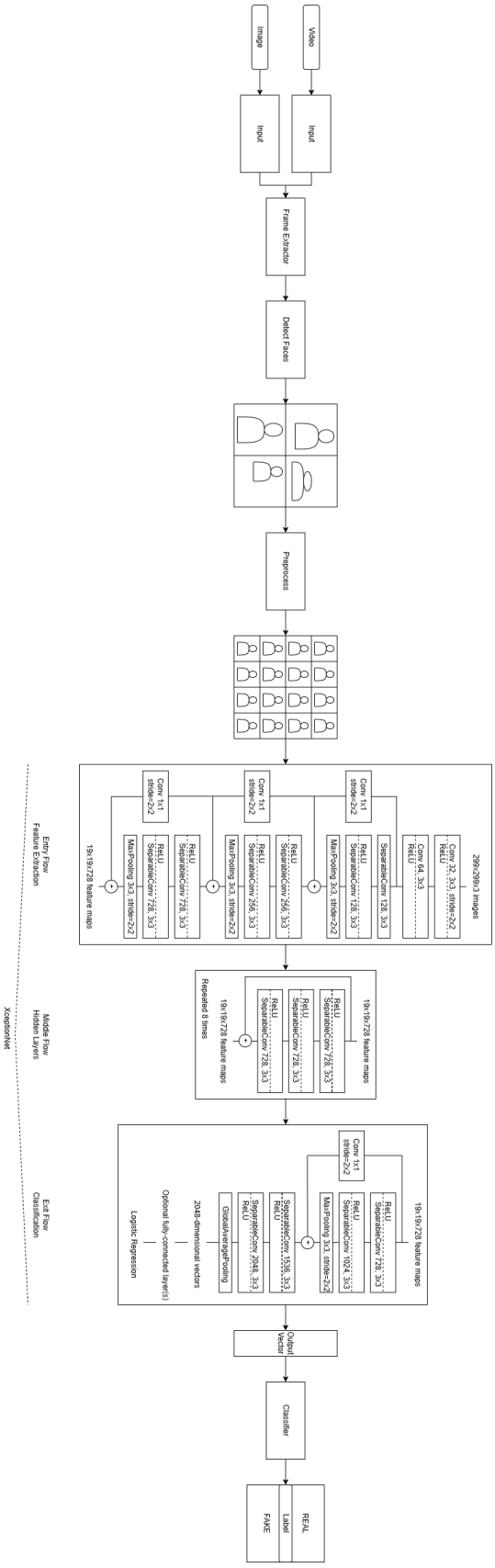


Fig. 6: System Architecture for AI-Powered Deepfake Detection system overview

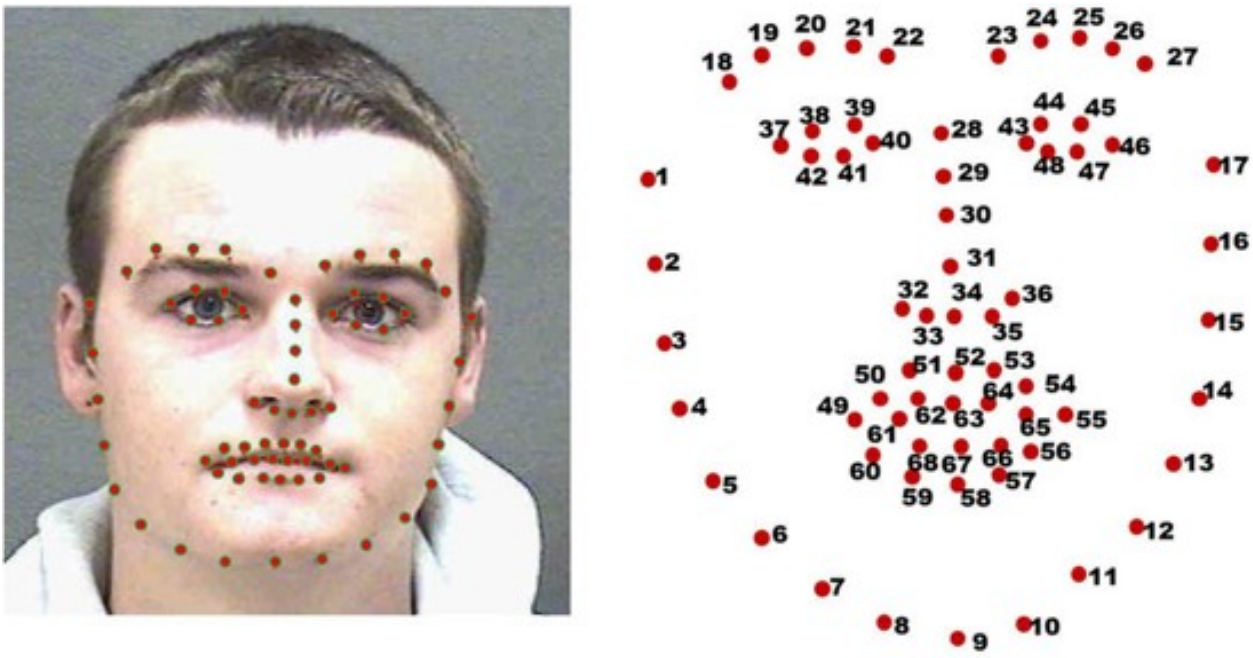


Fig. 7: Dlib Face Detection Library process example.

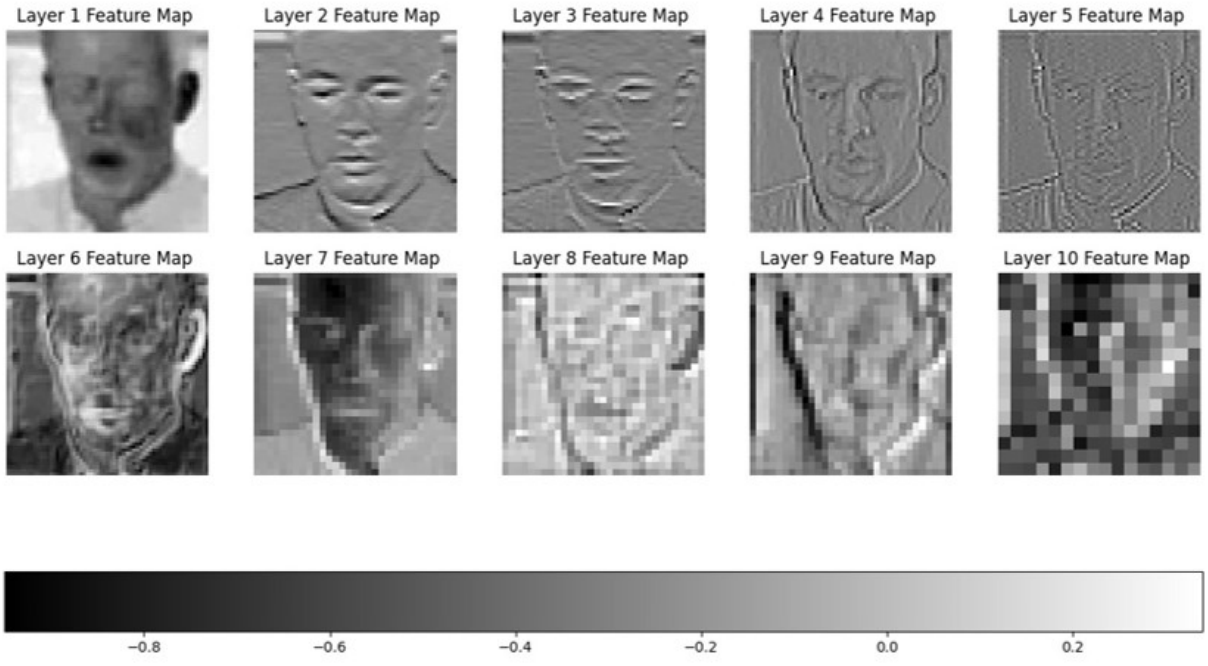


Fig. 9: Example for the first 10 layers of XceptionNet's entry flow.

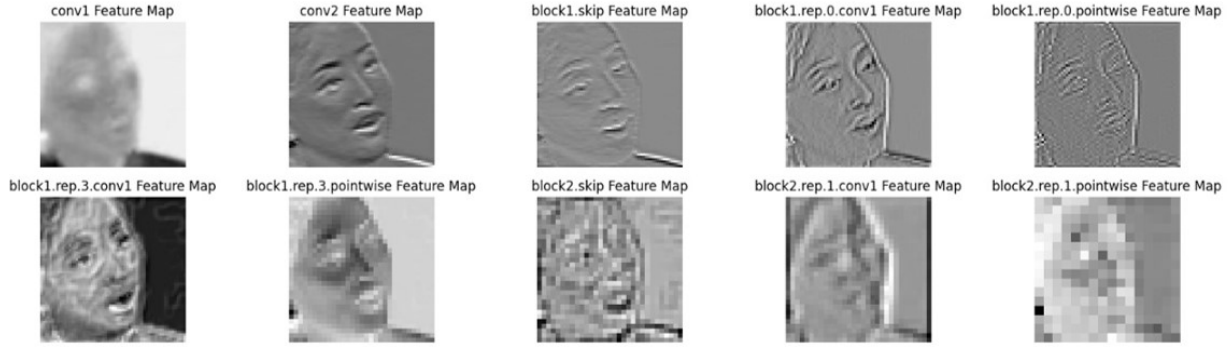


Fig. 10: Another example for the first 10 layers of XceptionNet's entry flow.

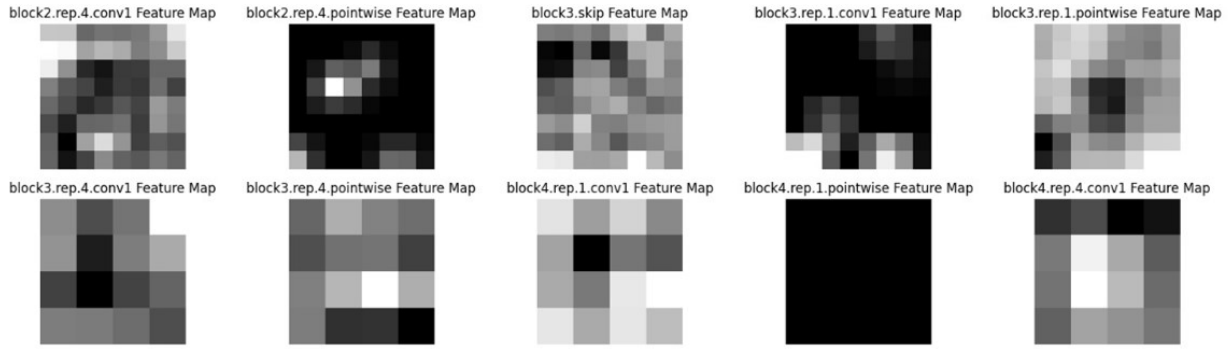


Fig. 11: Example for the last 10 layers of XceptionNet's exit flow.

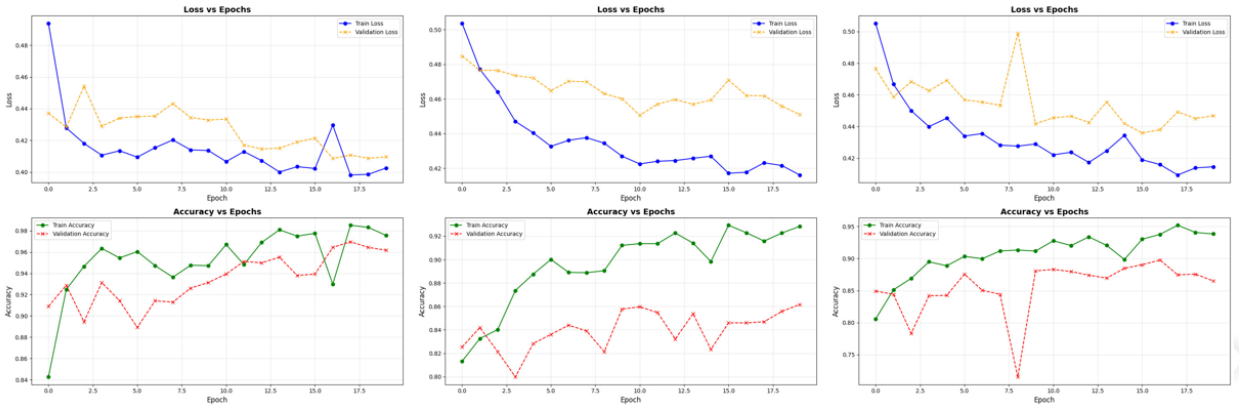


Fig. 12: Training curves for training 20 epochs with mentioned datasets; Celeb-DF-v2, FaceForensics++, Hybrid Dataset;respectively.

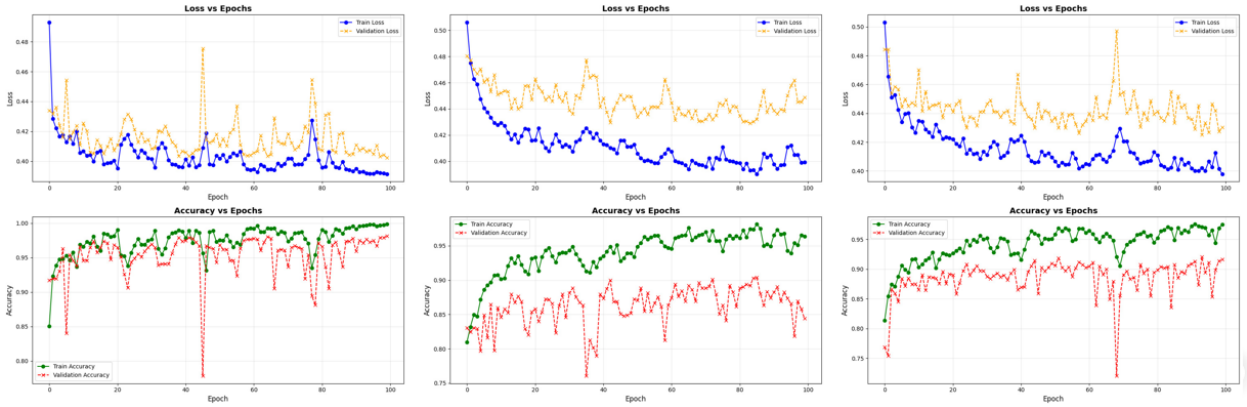


Fig. 13: Training curves for training 100 epochs with mentioned datasets; Celeb-DF-v2, FaceForensics++, Hybrid Dataset;respectively.