DEEP FAKE RESEARCH PROJECT

Emre BÜYÜKYILMAZ Sena YALÇIN İlayda Zeynep KARAKAŞ **Advisor:** Prof. Dr. Nazlı İKİZLER CİNBİŞ

Department of Computer Engineering

Introduction

Deep Fakes use advanced generative models to create highly realistic manipulated images and videos by swapping or altering facial identities. These techniques have enabled novel applications in entertainment, virtual reality, and creative content generation. However, the same underlying methods also cause the spread of misinformation, fraud, and privacy violations. Recent high-profile cases have shown how DeepFakes can undermine public trust.

Despite numerous detection approaches, many methods still struggle with generalization across diverse datasets. Challenges include varying video resolutions, compression artifacts, and adversarial manipulations designed to evade detection. There is a clear need for ensemble-based detection system capable of maintaining high accuracy under real-world conditions. This project aims to address these gaps by evaluating multiple CNN architectures and combining their strengths through a hybrid model.

Dataset and Preprocessing

Our experiments utilize two primary datasets:

- **FaceForensics++**: 5,000 videos (1,000 real, 4,000 DeepFake) across four manipulation methods: FaceSwap, NeuralTextures, Deepfakes, and Face2Face.
- **Celeb-DF**: Used as cross-dataset evaluation to test generalization capabilities.

Preprocessing Pipeline:

- Face detection and alignment using dlib library
- Frame extraction (5-30 frames per video depending on experiment phase)
- Videos with inconsistent face detections were excluded
- Data augmentation: random flips, slight rotations, and color jitter
- Class weighting (5:1) applied to handle dataset imbalance

Methodology

Phase 1: 2D CNN Baseline Models (5 frames per video)

- ResNet50, EfficientNetB0, and XceptionNet with ImageNet weights
- Final fully connected layer modified for binary classification
- Adam optimizer with class-weighted cross-entropy loss
- Trained a hybrid model from scratch by combining multiple architectures into a single network.

Phase 2: Temporal Feature Exploration (10 frames per video)

- Standard 3D CNNs: r3d_18, SlowResNet
- Custom 3D ResNet with additional regularization
- Simple frame aggregation vs. explicit temporal modeling

Phase 3: Advanced Architectures (30 frames per video)

- Frame expansion to capture more temporal information, XceptionNet retrained on expanded dataset ViTs (ViT-16), Modern CNN variants (ConvNeXt-S), Hierarchical transformers (Swin-S), Hybrid architectures (3D ResNet + Transformer, Xception + Transformer)

Cross-Dataset Evaluation:

- Models trained on FaceForensics++ and evaluated on unseen Celeb-DF dataset
- Focus on ROC AUC scores rather than accuracy due to class imbalance in both datasets

2D CNN Performance Analysis

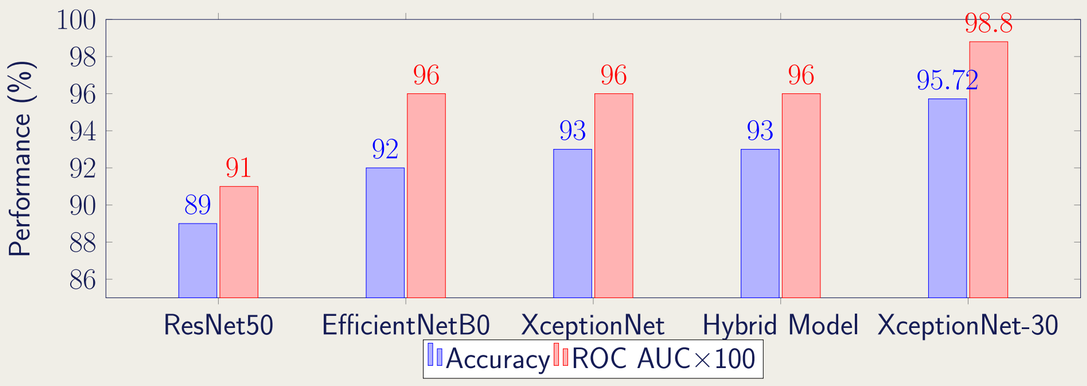


Figure 1. Performance comparison of 2D CNNs. XceptionNet-30 uses 30 frames per video, while others use 5.

Key Findings:

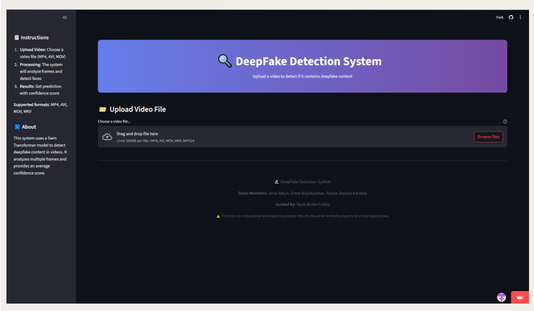
- XceptionNet outperformed other architectures
- EfficientNetB0 achieved competitive performance with significantly lower computational requirements (92% accuracy)
- ResNet50 performed the worst (89% accuracy) despite having the highest parameter count among the three models.
- The hybrid model matched XceptionNet's performance but with reduced variance in predictions
- Increasing frames from 5 to 30 significantly improved XceptionNet performance (95.72% acc.)

Detection Performance by Manipulation Type (XceptionNet-30):

Method	FaceSwap	Deepfakes	Face2Face	NeuralTextures
AUC	0.9921	0.9937	0.9877	0.9785

Interactive Demo Prototype

We also considered packaging our top-performing model into a simple, interactive application. Users would upload a video or an image, which the app automatically splits into frames and feeds into the classifier. The interface would then display a real-time verdict—Deepfake or Real—along with confidence scores.



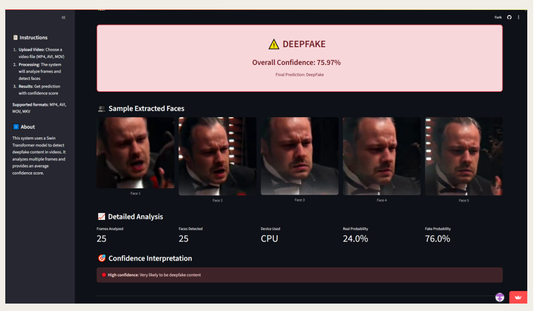


Figure 2. UI for our app

3D CNN and Temporal Analysis

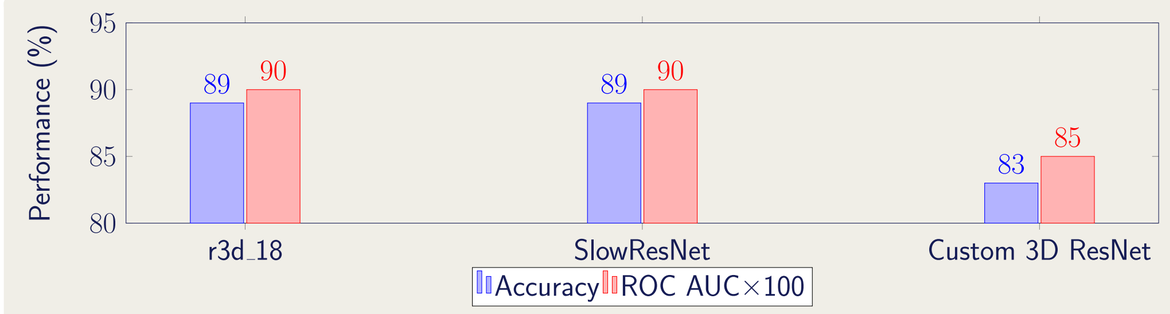


Figure 3. Performance comparison of 3D CNN models using 10 frames per video.

3D CNN Architecture Benefits:

- Capable of capturing temporal inconsistencies across frames
- Potentially more robust to single-frame artifacts
- Direct modeling of motion patterns that may reveal deepfake artifacts

Key Insight: While 3D CNNs theoretically offer advantages for video-based tasks, their performance was constrained by the limited temporal context (10 frames). This finding motivated our expansion to 30 frames in subsequent experiments.

Advanced Architectures and Cross-Dataset Evaluation

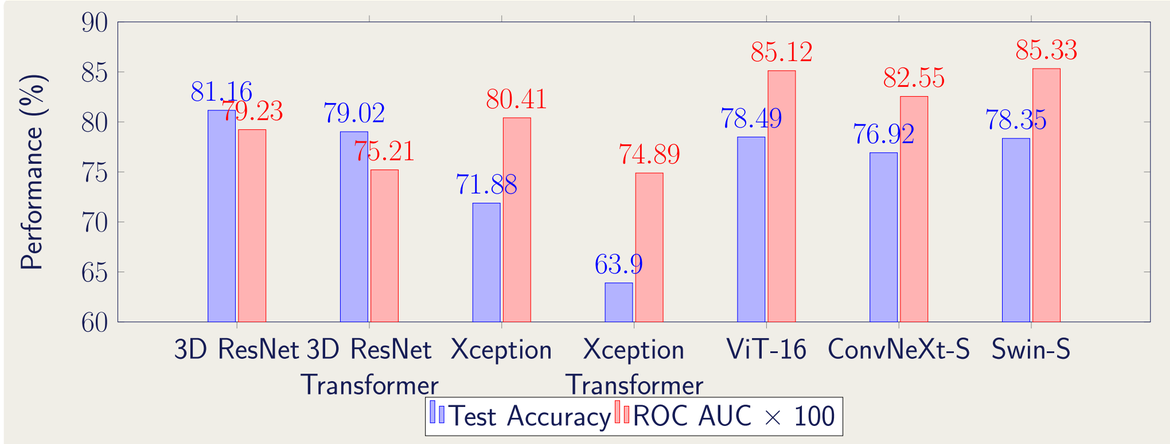


Figure 4. Performance comparison of different models on FaceForensics++ (train) and Celeb-DF (test) using 30 frames per video.

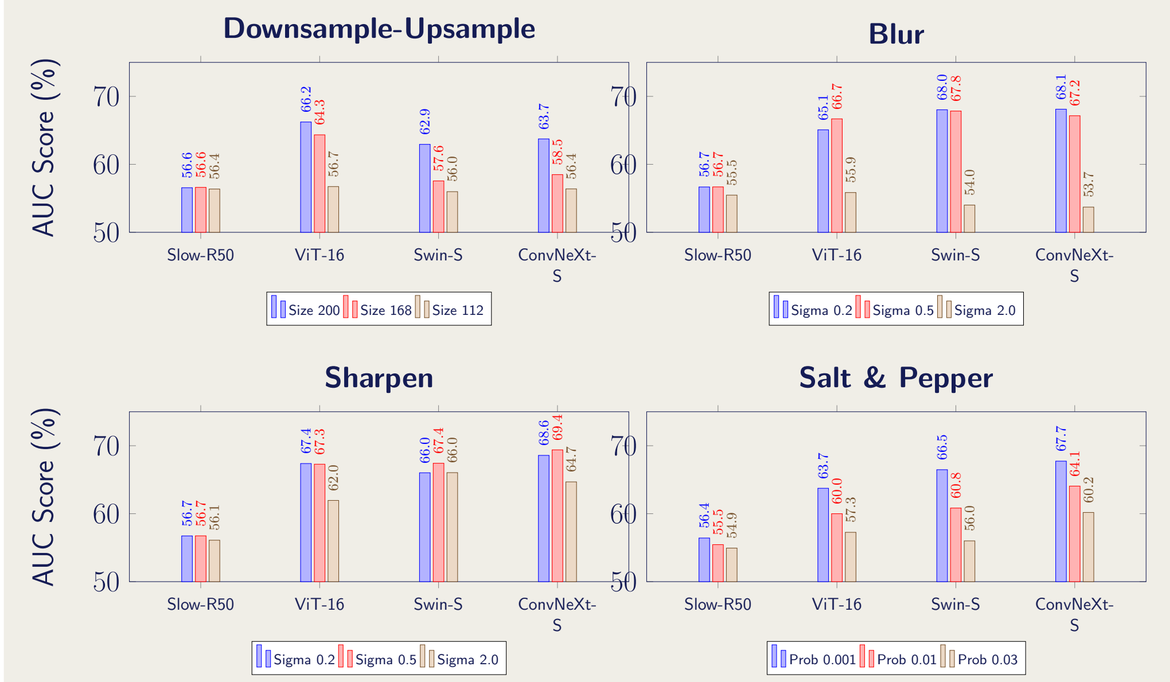
Cross-Dataset Generalization:

- Significant performance drop when testing on unseen Celeb-DF dataset
- Transformer-based models (ViT-16, Swin-S) demonstrated superior generalization
- Swin-S achieved the highest AUC (0.8533) through hierarchical feature learning
- Hybrid approaches combining CNNs with transformers underperformed expectations

Metrics Analysis:

- Accuracy and AUC rankings significantly differ due to dataset imbalance
- 3D ResNet achieved highest accuracy (81.16%) but moderate AUC (0.7923)
- ROC AUC provides more reliable performance assessment for imbalanced classification tasks

Model Performance Under Various Perturbations



Model performance (AUC Score %) under four different types of data perturbation. Each chart shows the robustness of four models to varying intensity of a specific perturbation.

ConvNeXt-S and Swin-S consistently demonstrated the highest robustness against most perturbations. While performance clearly degrades with higher intensity for Downsampling, Blur, and Salt Pepper, the Sharpen filter had a much less pronounced negative impact. This shows that model accuracy is more sensitive to artifacts like noise and resolution loss than it is to sharpening.

References

Zhu, L., B. Liao, Q. Zhang, X. Wang, W. Liu, and X. Wang (2024). *Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model*. URL: https://arxiv

emrebuyukyilmaz@hacettepe.edu.tr senayalcin@hacettepe.edu.tr ilaydakarakas@hacettepe.edu.tr