CS22B1081 Neha Kantheti
CS22B2030 Sri Manaswini Velide
ME22B2035 Bhargavi Antham

Deep Fake Detection

1. Project Overview

Deepfakes represent a new frontier in synthetic media generation, using artificial intelligence to manipulate facial expressions and voices in videos. As this technology advances, the detection of deepfakes has become increasingly crucial to prevent misinformation, identity theft, and digital forgery. This project explores deep fake detection using three state-of-the-art deep learning approaches:

- ResNet18 combined with LSTM
- Vision Transformer (ViT)
- Xception Network

Each model is evaluated on its ability to classify video clips from the Celeb-DF v2 dataset as real or fake.

2. Problem Statement

With the increasing accessibility of deepfake generation tools, detecting manipulated videos has become an urgent concern for digital media platforms, news agencies, and cybersecurity professionals. The problem addressed in this project is to accurately classify deep fake and real videos using robust deep learning models that can generalize across realistic and compressed video formats.

3. Dataset

The Celeb-DF v2 dataset is used for training and evaluation. It contains over 5,600 videos (real and fake) of celebrities speaking in diverse environments and lighting conditions. Each video is labeled as **real (0)** or **fake (1)**. The dataset is known for its high quality and subtle manipulations, making it a strong benchmark for detection models.

Find the dataset here.

4. Model Architecture

4.1 Model Architecture Overview

- ResNet18 + LSTM:
    - ResNet18 is used to extract 512-dimensional spatial features from each frame.
    - A single-layer LSTM learns the temporal dependencies between the 15 extracted frames.

    - Output from the final LSTM timestep is passed to a linear layer and sigmoid for prediction.

- Vision Transformer (ViT):
    - Frames are split into patches (16×16), flattened and linearly embedded.
    - A learnable class token is prepended, and positional encodings are added.
    - Multiple transformer encoder layers with self-attention mechanisms process the sequence.
    - Final classification is done using the class token embedding.

- Xception Network:
    - Utilizes depth wise separable convolutions, which reduce model size and improve efficiency.
    - Pretrained on ImageNet, then fine-tuned for binary classification.
    - Input shape: 224×224×3 per frame.
    - Final layers replaced with global average pooling, dropout, and sigmoid output.

5. Data Preprocessing

5.1 Frame Extraction

- 15 uniformly sampled frames per video.
- RGB conversion via OpenCV.
- Resizing to 224×224 (Xception, ResNet), and 256×256 (ViT).

5.2 Batching and Loading

- Normalization: mean=[0.5, 0.5, 0.5], std=[0.5, 0.5, 0.5].
- For LSTM: tensors reshaped to [Batch, Time, Channels, Height, Width].
- Efficient shuffling and batching for training.

6. Training Process

6.1 Loss Function

- All models use Binary Cross-Entropy with Logits (BCEWithLogitsLoss).

6.2 Optimization

- Optimizer: Adam

- Learning Rate: 1e-4

- Batch Size: 16 (Xception, ViT), 8 (ResNet18+LSTM)

- Epochs: 10–20 depending on model stability

6.3 Evaluation Metrics

- Accuracy
- Precision
- Recall
- F1-Score
- ROC AUC
- EER

7. Evaluation and Results

7.1 Metrics

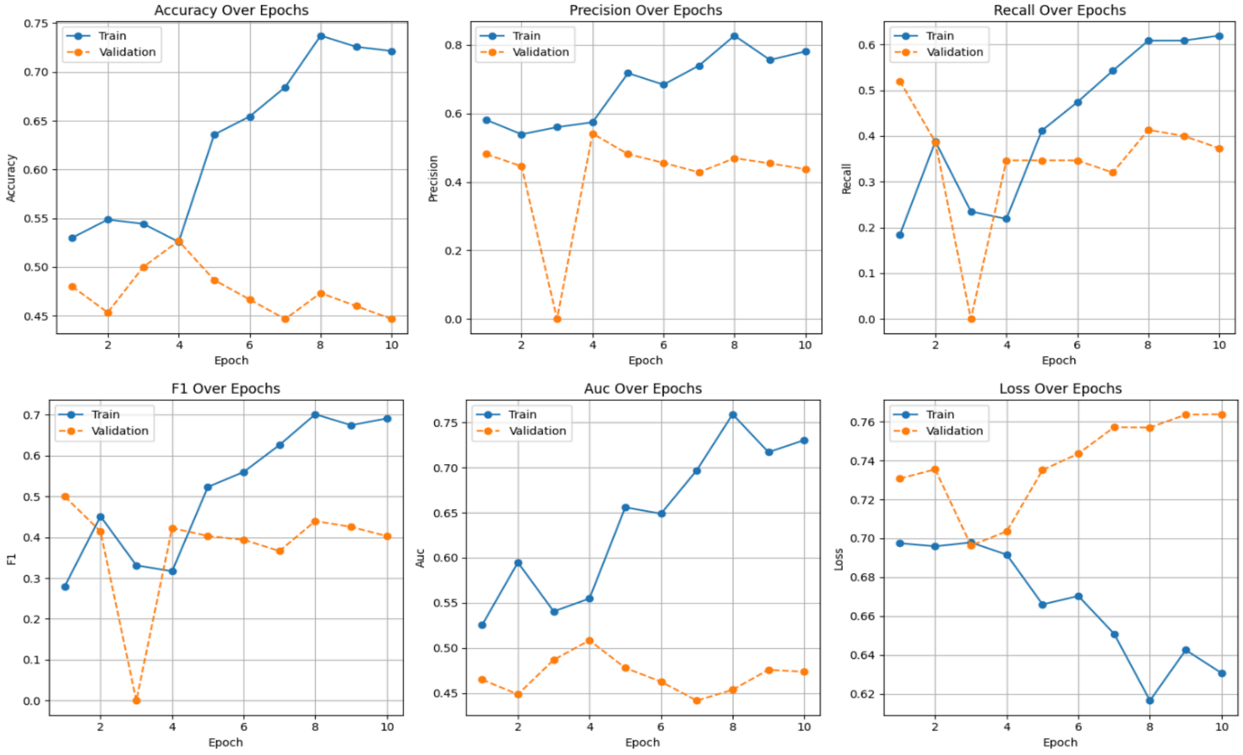
Metric	ResNet-18	Vision Transformer	XceptionNet
Accuracy	48.67%	77.88%	94.00%
Precision	48.44%	83.33%	94.00%
Recall	41.33%	85%	100.00%
F1 Score	44.60%	84.16%	97.00%
AUC	0.4933	0.8312	0.8570
EER	N/A	N/A	0.3027

7.2 Visualization

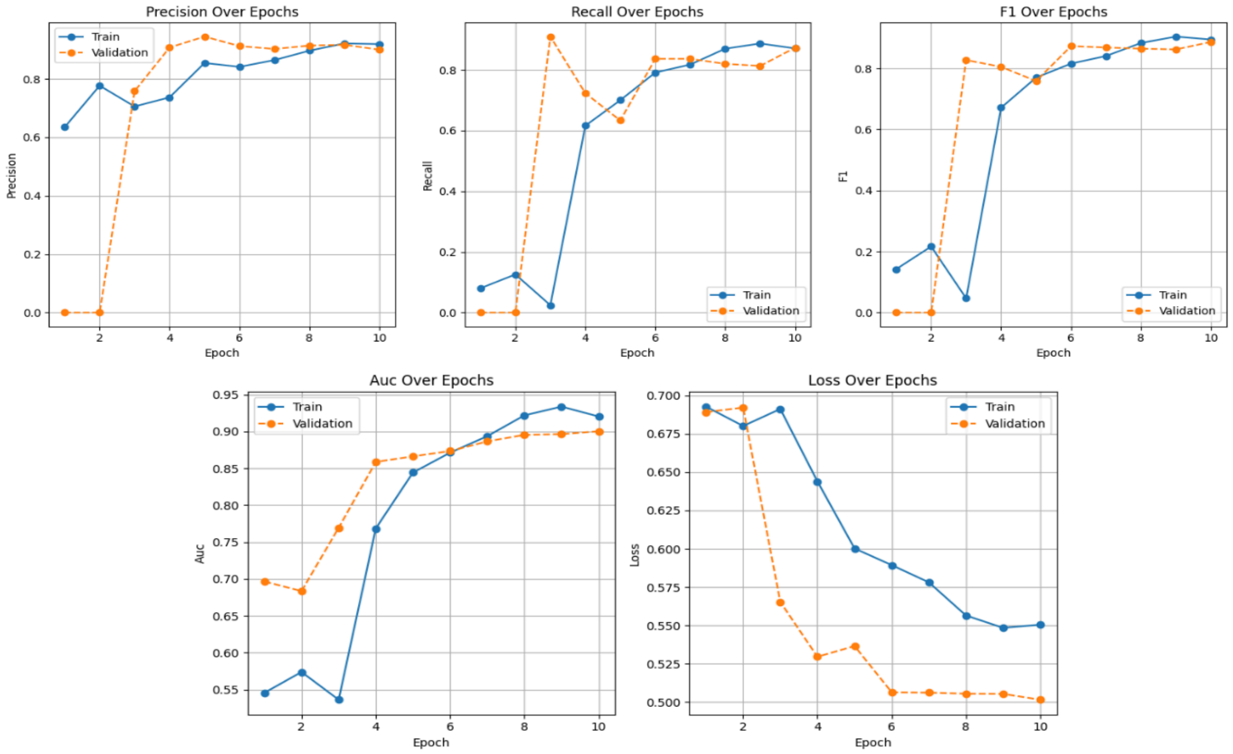
- Confusion matrices:
  - ResNet18 + LSTM: Significant misclassification.
  - ViT: Slight confusion, especially with compressed fake videos.
  - Xception: Clear separation between real and fake.

- Prediction inspection: Display of image frames with predicted and actual labels.

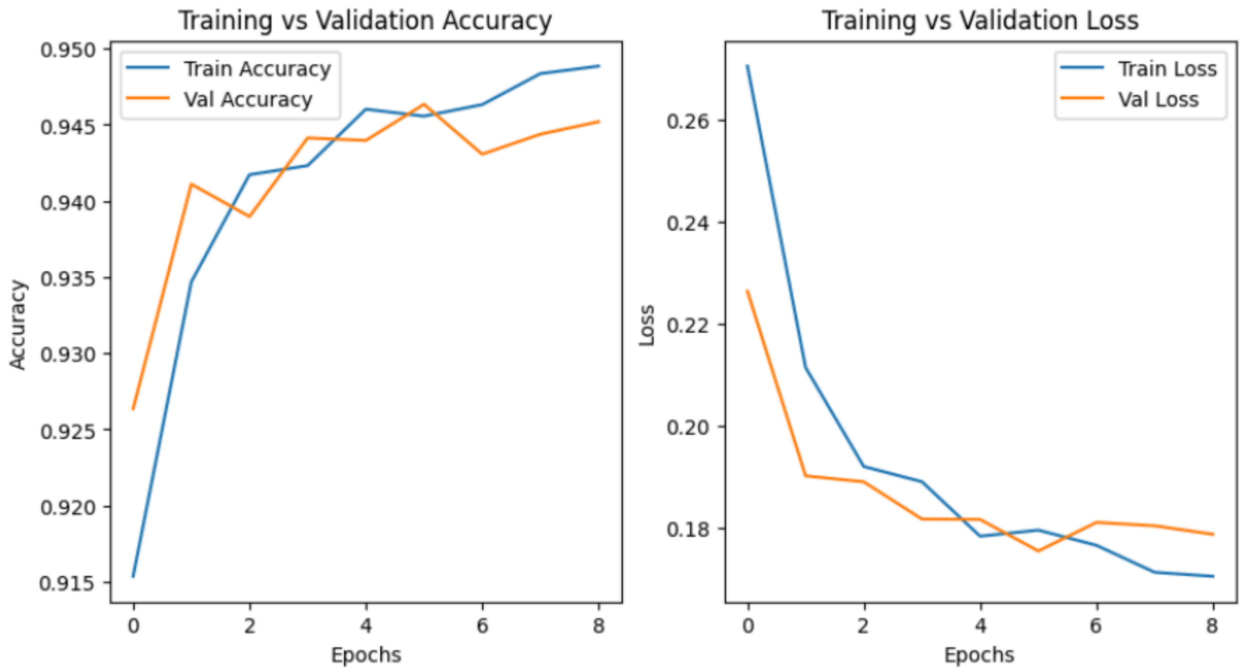
ResNet+LSTM:



Vision Transformer:



XceptionNet:


Training vs Validation Accuracy

Training vs Validation Loss

7.3 Results

- ResNet18 + LSTM:
  - Struggled with low accuracy (48.6%).
  - Temporal modeling was insufficient with short clips.
  - Confusion matrix showed bias toward misclassifying fake as real.
- Vision Transformer:
  - Performs well on spatial consistency but sensitive to compression noise.
  - Requires more data and computation.
- Xception:
  - Achieved the best results due to deep convolutional feature extraction.
  - Fast convergence due to pretrained weights.

8. Challenges and Future Work

8.1 Challenges

- LSTM underperformance due to limited sequence modeling.
- Transformers require high-quality data and GPU resources.
- Overfitting in smaller models due to limited samples.

8.2 Future Work

- Try 3D CNN architectures (e.g., SlowFast, I3D) for better temporal modeling.
- Integrate face alignment and attention mechanisms.
- Combine visual with audio modalities.
- Improve class balance and explore hard-negative mining.

9. Conclusion

This project evaluated three deep learning-based approaches for deepfake detection. The Xception model provided the most consistent results, followed by the Vision Transformer. ResNet18 with LSTM showed the challenges of temporal modeling with limited data. Future improvements could leverage advanced temporal networks and multimodal inputs to enhance performance further.