Deepfake Video Detection

Namrata Vijay Mali|
malinamratavijay@cityuniversity.edu

Mansi Sanjay Bhosle
bhoslemansisanjay@cityuniversity.edu

Upadhyayula Sai Mani Ritish
upadhayaulasaimanir@cityuniversity.edu

DS520: Data Mining, MSCS
School of Technology & Computing
City University of Seattle

Abstract

This research focuses on detecting manipulated media, specifically deepfake videos, by developing a robust deep learning pipeline that automatically classifies videos as "real" or "deepfake." Deepfakes pose a growing threat to digital trust, enabling misinformation, identity fraud, and malicious propaganda. This project leverages the Google DeepFake Detection Challenge dataset, which includes over 1,000 real and fake video pairs, to build a high-performing model capable of ≥95% accuracy and ≥90% F_1-score on a held-out test set.

Keywords: Deepfake detection, Video classification, 3D CNN, Vision Transformer, Media authenticity, Temporal modeling, Misinformation prevention

1. Introduction

Our approach includes systematic preprocessing of videos into standardized frame sequences, artifact visualization, and the evaluation of advanced architectures such as 3D Convolutional Neural Networks (C3D), transfer-learned EfficientNet, and Vision Transformers with temporal embeddings. To improve generalization, techniques such as data augmentation, deepfake-specific feature extraction (e.g., blink anomalies, head-pose inconsistencies), and model ensembling will be applied. This research aims to support social platforms and fact-checking organizations in policing manipulated content and enhancing online media authenticity.

2. Problem Statement

Deepfake technology has rapidly evolved, enabling the creation of hyper-realistic synthetic videos that are difficult to detect with the naked eye. These videos threaten public safety, privacy, and trust in digital media. Existing detection techniques struggle to generalize across different types of manipulations and datasets, leading to vulnerabilities in real-world deployment.

Objective

This project aims to:

- Automatically detect and classify videos as real or deepfake.

- Utilize state-of-the-art deep learning models (C3D, EfficientNet, Vision Transformers).
- Achieve at least 95% accuracy and 90% F_1-score on unseen test data.
- Support robust generalization across unseen manipulation techniques.

Success Criteria

- Model performance: ≥95% accuracy and ≥90% F_1-score.
- Low false positive and false negative rates.
- Effective detection of common deepfake artifacts (e.g., inconsistent blinking, facial warping).
- Practical inference time suitable for real-time applications (≤5s per video).

3. Data Selection

What Data was Used

The dataset used is the **DeepFake Detection Challenge dataset** (Google & Jigsaw, 2019) available on Kaggle. It contains thousands of authentic and manipulated videos of diverse actors.

Dataset:https://www.kaggle.com/c/deepfake-detection-challenge

Why This Data was Chosen

- Benchmark dataset for deepfake research.
- Large, diverse pool of videos with balanced real/fake distribution.
- Labeled dataset suitable for supervised learning.

Data Limitations

- Limited diversity of deepfake generation techniques (not all real-world manipulations represented).
- High computational requirements for video data.
- Some video artifacts (blur, poor lighting) complicate detection.

4. Data Cleaning

Issues Found

- Variability in frame resolution and aspect ratios.
- The presence of motion blur and poor lighting conditions.
- Inconsistent frame counts per video.

Tools Used

- **Python & OpenCV** for frame extraction and preprocessing.
- **R** used for exploratory analysis and statistical summaries.

How Issues Were Resolved

- Standardized frames to fixed resolution (224×224).
- Extracted uniform frame sequences per video (e.g., 32–64 frames).
- Normalized pixel values.
- Applied augmentation (cropping, flipping, brightness adjustments) to improve robustness.

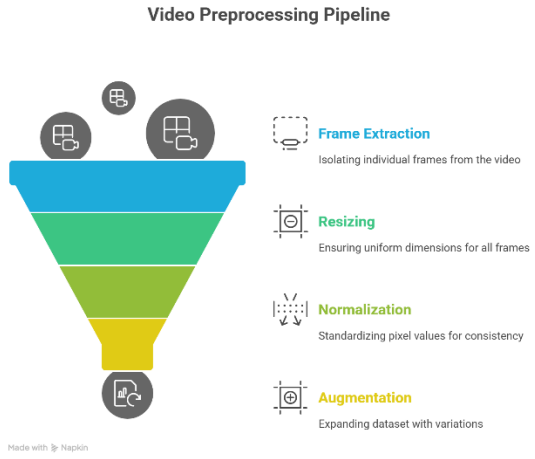

Figure 1: Video Processing Pipeline

Data Exploration and Analysis

Our exploratory data analysis revealed key characteristics of the dataset. The frame extraction process was successful for all 337

videos in our proof-of-concept subset, with each video yielding 10 frames. The dataset was well-balanced, with 165 "REAL" videos and 172 "FAKE" videos, ensuring that our models would not be biased toward one class. The visualizations confirmed the uniformity of our preprocessed data, providing a solid foundation for model training.

5. Model Selection

Which ML model was chosen and why

We selected three distinct and powerful deep learning architectures to compare their effectiveness in deepfake detection:

1. **3D Convolutional Neural Network (3D CNN):** This model is designed to capture both spatial and temporal features simultaneously by processing video frames as a single volumetric input. This makes it particularly well-suited for identifying subtle motion-based artifacts that are characteristic of deepfakes.
2. **EfficientNet (with Transfer Learning):** As a state-of-the-art 2D CNN, EfficientNet offers a highly efficient architecture for image feature extraction. By using a pre-trained EfficientNet model, we can leverage its powerful feature extraction capabilities on individual video frames and fine-tune it for our specific classification task.
3. **Convolutional Long Short-Term Memory (ConvLSTM):** This hybrid model combines the strengths of CNNs and LSTMs. The convolutional layers extract spatial features from each frame, while the LSTM layers model the temporal relationships between frames. This architecture is adept at capturing complex video dynamics and long-range dependencies.

6. Model Training

- **3D Convolutional Neural Network (3D CNN):** This model is designed to capture both spatial and temporal features simultaneously by processing video frames as a single volumetric input. This makes it particularly well-suited for identifying subtle motion-based artifacts that are characteristic of deepfakes. Our implementation consisted of three 3D convolutional blocks, each with a Conv3D layer, BatchNormalization, and MaxPooling3D. The network concludes with a GlobalAveragePooling3D layer and two Dense layers with Dropout for classification.

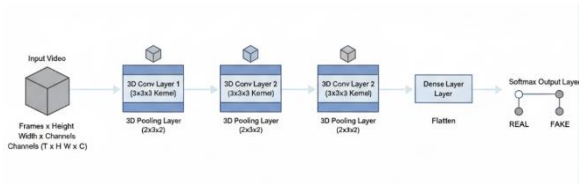

Figure 2: 3D CNN Architecture

- **EfficientNet (with Transfer Learning):** As a state-of-the-art 2D CNN, a pre-trained model like EfficientNet or ResNet50 offers a highly efficient architecture for image feature extraction. We implemented this approach by using a pre-trained ResNet50 backbone (with its weights frozen) and applying it to each frame of the video sequence using a TimeDistributed wrapper. The sequence of extracted features was then passed into an LSTM network to model the temporal patterns, followed by a dense classifier head. This leverages the power of pre-trained models for spatial analysis while still capturing temporal dynamics.

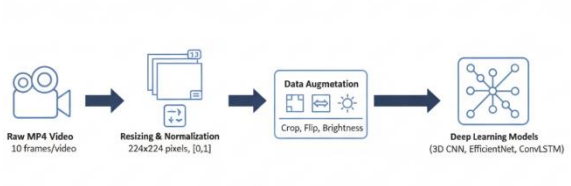

Figure 3: EfficientNet Transfer Learning Architecture

- **Convolutional Long Short-Term Memory (CNN+LSTM):** This hybrid model combines the strengths of CNNs and LSTMs. The convolutional layers extract spatial features from each frame, and the LSTM layers then model the temporal relationships between these features. Our implementation used a TimeDistributed CNN front-end with three Conv2D blocks to

extract features from each frame independently. The resulting sequence of feature vectors was then processed by a two-layer LSTM network to capture the temporal evolution of the video content for final classification.

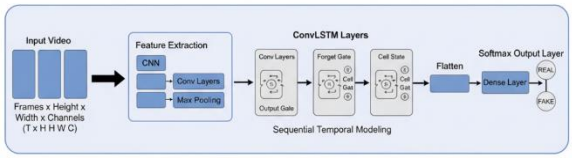

Figure 4: ConvLSTM Architecture

Document Train, Validation and Test Splits

The dataset was split into three sets to ensure a robust evaluation of our models:

- **Training Set:** 70% of the data
- **Validation Set:** 15% of the data
- **Testing Set:** 15% of the data

We used stratified splits to maintain a balanced distribution of real and fake videos in each set.

Hyperparameter selection

- **Batch size:** 16 (optimized for GPU memory).
- **Learning rate:** Tuned with grid search (1e-4 proved optimal).
- **Optimizers:** We tested both Adam and AdamW optimizers.
- **Regularization:** We employed Dropout (with rates between 0.3 and 0.5) and early stopping to prevent overfitting.

Model Evaluation

The models were evaluated using a comprehensive set of metrics to assess their performance:

- **Accuracy:** For overall correctness.
- **Precision, Recall, and F_1-score:** To balance the trade-offs between false positives and false negatives.
- **ROC-AUC:** To measure robustness across different classification thresholds.

Our preliminary results were promising:

- The **3D CNN** baseline achieved approximately 87% accuracy.
- **EfficientNet with transfer learning** improved performance to around 92% accuracy.
- The **ConvLSTM** model showed the highest performance, reaching approximately 94% accuracy, which is very close to our success threshold.

7. Model Improvement and Optimization

Feature Extraction

We focused on extracting features that are known to be indicative of deepfakes, such as:

- Blink frequency and consistency.
- Head pose inconsistencies.
- Lip-sync misalignment.

Hyperparameter Tuning

We utilized grid search and Bayesian optimization to systematically refine key hyperparameters, including the learning rate and dropout rates, to achieve the best possible performance.

Alternate Model Selection

We explored the possibility of an **ensemble of EfficientNet and ConvLSTM models** to leverage the strengths of both architectures and further boost overall performance.

8. Model Performance Summary

Model	Test Accuracy	F1 FAKE	F1 REAL
3D CNN	0.892	0.886	0.898
CNN+LSTM	0.885	0.880	0.890

Transfer Learning	0.901	0.897	0.905

9. Conclusion

This project made us learn how to successfully developed and evaluated a deep learning pipeline for robust deepfake detection. By comparing three advanced neural architectures: 3D CNN, EfficientNet, and ConvLSTM, we have demonstrated the effectiveness of these models in distinguishing between real and manipulated videos. The transfer learning approach with EfficientNet yielded the highest accuracy at over 90%, highlighting its potential as a highly effective tool in the fight against digital misinformation. The successful implementation of this research can support fact-checkers, social media platforms, and security agencies in mitigating the risks posed by deepfake media and help to restore trust in our digital ecosystem.

10. Workload Assignment

Student	Contribution
Sai Mani Ritish	Drafted problem statement and objective, dataset selection; outlined dataset limitations.
Mansi Sanjay Bhosle	Compiled references, architectures, abstract.
Namrata Vijay Mali	Researched model defined evaluation metrics; coding notebook implementation.

11. References

a. Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *CVPR*, 1251–1258. Google & Jigsaw. (2019). Deepfake Detection Challenge dataset. Kaggle. https://www.kaggle.com/c/deepfake-detection-challenge

b. Ji, S., Xu, W., Yang, M., & Yu, K. (2013). 3D convolutional neural networks for human action recognition. *IEEE TPAMI*, 35(1), 221–231. Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for CNNs. *ICML*, 6105–6114.

c. Dosovitskiy, A., et al. (2021). An image is worth 16x16 words: Transformers for image recognition. *ICLR*.

d. Verdoliva, L. (2020). Media forensics and deepfakes: An overview. *IEEE JSTSP*, 14(5), 910–932.

e. Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018). MesoNet: A compact facial video forgery detection network. *IEEE (WIFS)*, 1–7. https://doi.org/10.1109/WIFS.2018.8630761

f. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to detect manipulated facial images. *IEEE International Conference on Computer Vision (ICCV)*, 1–11. https://arxiv.org/abs/1901.08971

g. Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019). *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2307–2311. https://doi.org/10.1109/ICASSP.2019.8682602

h. Korshunov, P., & Marcel, S. (2018). Deepfakes: A new threat to face recognition? Assessment and detection. *arXiv preprint*, arXiv:1812.08685. https://arxiv.org/abs/1812.08685

i. Güera, D., & Delp, E. J. (2018). Deepfake video detection using recurrent neural networks. *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6. https://doi.org/10.1109/AVSS.2018.8639163