From Data to Detection: A Machine Learning Approach to Spotting DeepFakes

By:
Shamsvi Balooni Khan
Saraswathi Baskaran
Sanjeev Durge
Mohini Jasthi

Table Of Contents

1. Introduction

In the age of enhanced media manipulation, the rise of deep fakes, realistic but fake media created using deep learning techniques poses enormous social hazards. The threats are serious, ranging from disinformation to slander. Deep Fakes are synthetic media in which a person's likeness is digitally manipulated, most typically via swapping faces in videos. As the technology underlying DeepFakes advances, recognizing them has become a major problem. This project attempts to create a machine learning pipeline that can detect DeepFake films by detecting both spatial distortions and temporal anomalies.

The primary purpose of this research is to determine if a video is genuine or a DeepFake by detecting tiny indicators of alteration. DeepFake models frequently create subtle inconsistencies, particularly across frames that are not always obvious to the naked eye. Our model is intended to detect those cues. We also aim to reduce false positives so that genuine films are not mistakenly labeled as fake, which is critical in sensitive circumstances such as news or court evidence.

2. Dataset Description

For this project, we used a version of the FaceForensics++ dataset obtained from Kaggle. The dataset consists of 400 full-length movies, evenly divided between 200 actual and 200 false videos, making it appropriate for a balanced binary classification problem. Each movie is well labeled to promote guided learning.

The real videos show unmodified footage of people conversing spontaneously, but the false movies were created utilizing cutting-edge face manipulation techniques like FaceSwap, DeepFakes, and FaceToFace. These approaches include varied degrees of visual alteration, ranging from exceedingly realistic to obvious artifacts.

The dataset's diversity is notable: it includes a variety of lighting situations, face emotions, head positions, and camera angles. This unpredictability adds a degree of realism that closely resembles real-world settings, making the process of discriminating between real and false movies more challenging and realistic of the difficulties encountered in actual applications.

Furthermore, the fake videos in the dataset differ in terms of modification quality, with some possessing mild imperfections and others displaying more dramatic distortions. This range of manipulation fidelity makes the dataset ideal for building a strong model capable of recognizing both obvious and convincing deep fakes.

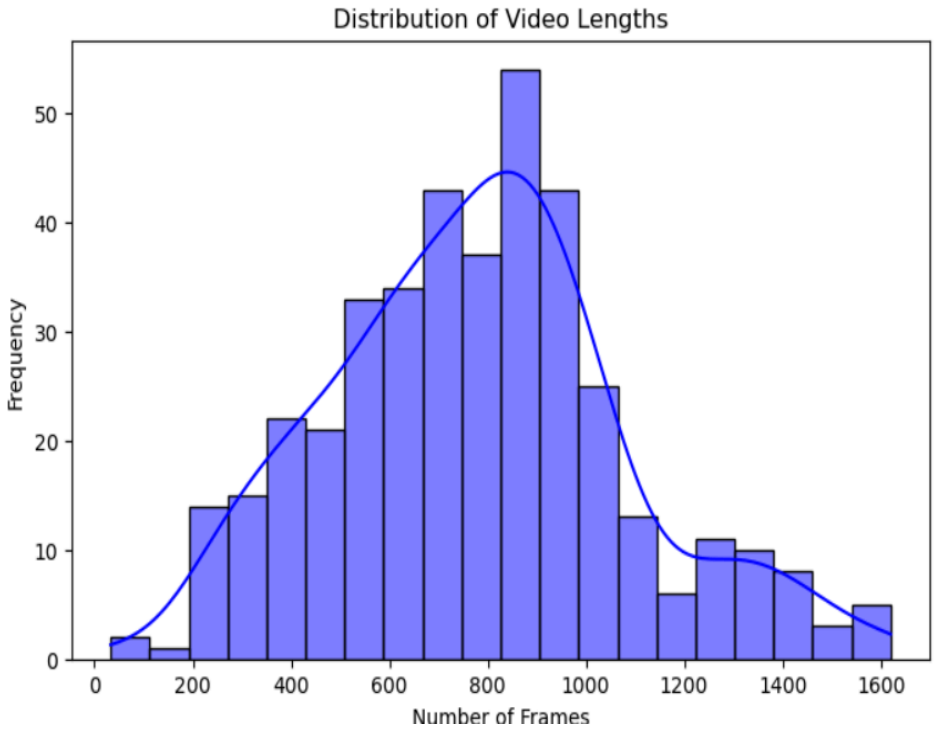

Fig1: Distribution of length of videos

The image illustrates the distribution of video lengths in the dataset, highlighting the frequency of different duration ranges. It provides a clear visual representation of how video lengths are spread across various time intervals.

3. Data Pre-processing

The initial stage of data preprocessing involved a thorough analysis of the dataset. The distribution of video lengths was examined to understand the spread and consistency of the videos across different durations. Following this, the class distribution was analyzed, which revealed a noticeable imbalance between real and fake videos. To gain further insights into the feature space, Principal Component Analysis (PCA) was performed, projecting high-dimensional video features into two dimensions. This helped in visualizing partial clustering, suggesting that real and fake videos occupy somewhat distinct regions in the reduced feature space.

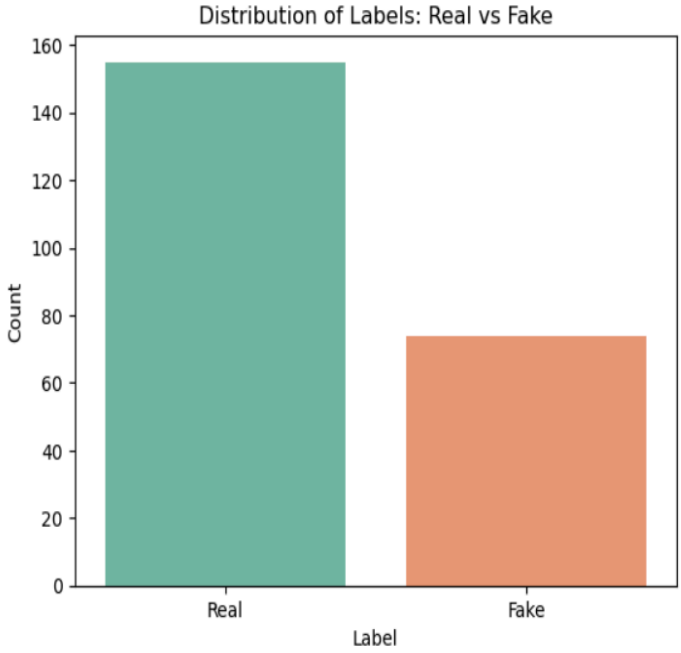

Fig 2: Distribution of classes (Real and Fake)

The image depicts the class distribution, showing an imbalance where one class significantly outnumbers the other. This visual highlights the disparity in the frequency of occurrences between the two classes.

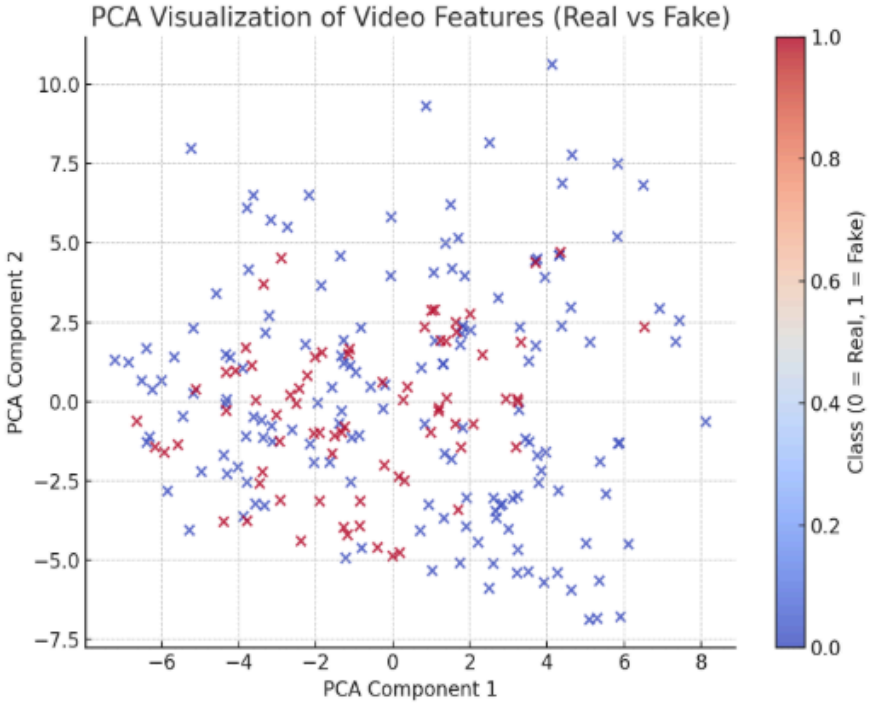

Fig 3: Principal Component Analysis

PCA has been performed to reduce the dimensionality of the high-dimensional video features. Although there is some overlap, the scatter plot reveals partial clustering, indicating that real and fake videos occupy distinct regions within the feature space.

Building on these observations, from each video, 5 to 10 equally spaced frames are extracted. This method captures the temporal dynamics of face emotions and movements without significantly increasing the dataset size, achieving a compromise between computing efficiency and rich temporal information.

Next, we used the Multi-task Cascaded Convolutional Neural Network (MTCNN) to recognize faces. MTCNN is a popular approach for finding face landmarks because of its excellent accuracy under varying lighting and posture situations. Once faces were identified, we clipped the face region from each frame, thus reducing extraneous background noise and directing the model's attention just to facial regions where modifications are most likely to occur.

After cropping, all face photos were scaled to 224 × 224 pixels for homogeneity throughout the dataset and compliance with conventional CNN designs. We then normalized the pixel values, generally scaling them between 0 and 1, to facilitate faster and more stable model training by standardizing the input data distribution.

During preprocessing, we used data augmentation approaches to increase the model's generalizability to new data and imitate real-world settings. These comprised random rotations, brightness changes, and horizontal flips. Such augmentations make the model more resilient to changes that may occur during deployment, such as varied head postures, lighting conditions, and tiny occlusions.

Finally, the preprocessed data was stratified and divided into two sets: 70% for training and 30% for testing, to retain the original class distribution of actual and fraudulent films. This stratified sample is critical for ensuring a fair evaluation environment and avoiding bias against either class during model assessment.

4. Modeling Approaches

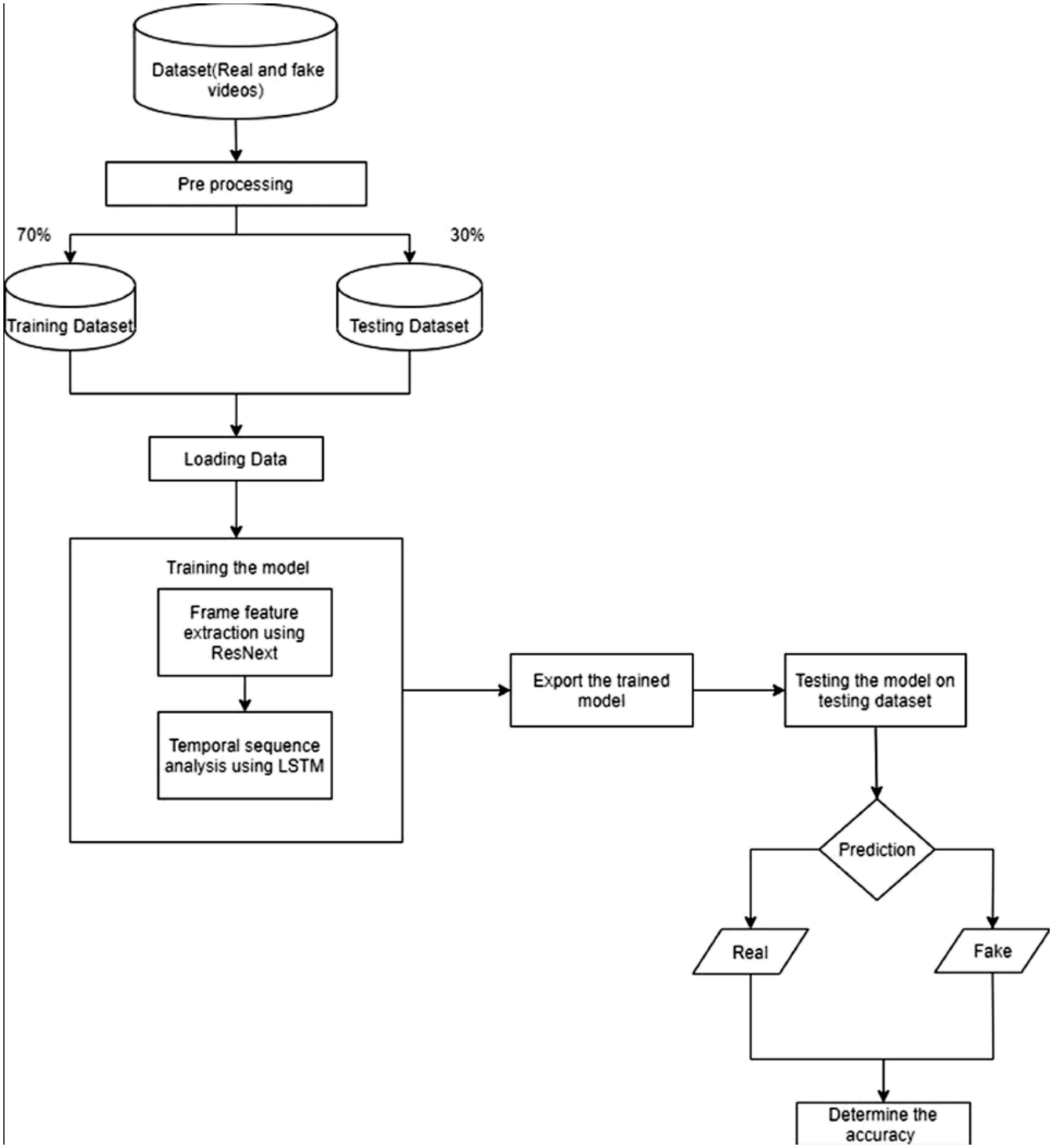

Fig 4: Representation of workflow of the proposed model

4.1 Baseline Model: XceptionNet

The initial baseline for our DeepFake detection pipeline was created with XceptionNet, a deep Convolutional Neural Network (CNN) architecture noted for its superior performance in image classification tasks.

- XceptionNet was used to identify spatial aberrations and pixel-level distortions in video frames. DeepFake manipulations frequently generate small discrepancies within face areas, such as abnormal textures, artificial mixing, or slight color mismatches, which CNNs are quite good at detecting.
- Limitation: XceptionNet functions well on static pictures, but does not consider temporal correlations between frames. As a result, it is unable to identify temporal irregularities, such as inconsistent facial motions or abrupt transitions, which are common among DeepFakes.

4.2 Primary Model: LSTM

To address a significant drawback of static frame analysis, we improved the baseline model by including temporal modeling using an LSTM network.

- **Architecture:**
    - **Feature Extraction:** Frames are processed through a pre-trained MTCNN model to extract feature vectors. This phase guarantees that each frame is represented with a diverse collection of spatial information while considerably lowering dimensionality.
    - **Temporal Modeling:** Frame-level feature vectors are input into a single-layer LSTM with 64 hidden units. The LSTM examines sequential data and learns patterns about how characteristics grow over frames. This allows the model to recognize unnatural transitions or irregularities in face movements that static models would miss.
    - Classification Head: The LSTM's hidden state is transferred via dense (completely connected) layers, ending in an output layer with a Softmax activation function. This layer generates probability for two classes: real and fake.

- **Why LSTM?**
  LSTMs are particularly intended for sequential data and are extremely good at capturing long-term dependencies. In the case of deep fake detection, face modifications frequently produce temporal artifacts such as inconsistent lip movement, odd blinking, or frame-to-frame misalignments that are difficult to identify by examining individual frames alone. By adding an LSTM, our model can grasp not just what is in a frame, but also how face traits change over time, giving a considerably more effective and comprehensive approach to identifying deep fakes.

5. Model Training and Evaluation

5.1 Training setup

To train the DeepFake detection model, we developed a robust configuration that prioritized both convergence efficiency and performance stability:

- **Loss Function:** We used Categorical Cross-Entropy Loss, a typical method for multi-class classification issues (with two classes: real and false). This loss function calculates the difference between the projected probability distribution and the real distribution, which encourages the model to provide high probabilities for the proper class.
- **Optimizer:** The Adam Optimizer, an adaptive learning rate optimization technique, was used to train the model by calculating separate rates for each parameter. Adam combines the benefits of two prominent methods: AdaGrad (for sparse gradients) and RMSProp (for non-stationary settings). Adam achieves quicker and more stable convergence by keeping and updating estimates of the gradients' first (mean) and second (uncentered variance) moments, which is critical given the complexity of the hybrid XceptionNet + LSTM architecture.
- **Evaluation Metric During Training:** During training, the evaluation metric focused on recall for the DeepFake class (class 1). Detecting fake videos (class 1) is crucial, because missing a DeepFake (false negative) might have far-reaching repercussions than erroneously designating a legitimate video as fake.

5.2 Evaluation Metrics

Following training, we performed a thorough review utilizing numerous indicators to acquire a holistic perspective of the model's performance.

- **Classification Report:**
    - We created a comprehensive categorization report that includes essential indicators.
    - **Precision:** Determines how many anticipated fraudulent videos were indeed phony.
    - **Recall:** Determines how many genuine fraudulent videos were properly detected by the algorithm.
    - **F1-Score**: The harmonic mean of accuracy and recall, providing a balanced evaluation of the two.

Classification Report:

	precision	recall	f1-score	support
0	0.85	0.94	0.89	31
1	0.83	0.67	0.74	15
accuracy			0.85	46
macro avg	0.84	0.80	0.82	46
weighted avg	0.85	0.85	0.84	46

Fig 4: Performance metrics for the final model on FF++ dataset

The model achieved an overall accuracy of 85% on the test set. For class 0 (real videos), it showed strong performance with a precision of 0.85, recall of 0.94, and an F1-score of 0.89. For class 1 (fake videos), the model achieved a precision of 0.83, but the recall dropped to 0.67, resulting in an F1-score of 0.74. The macro average F1-score is 0.82, and the weighted average F1-score is 0.84, indicating that while the model performs better on real videos, it still maintains reasonable effectiveness across both classes.

- **Confusion Matrix:**
  The confusion matrix shows the breakdown of true positives, true negatives, false positives, and false negatives. This visualization showed us how successfully the model identified between actual and fraudulent videos, as well as where the mistakes were concentrated.

5.3 Performance Focus

- **Primary Goal:**
  The main objective was to increase recall for the fictitious class (class 1). High recall means that the model successfully recognizes the bulk of bogus movies, despite a somewhat higher false positive rate. In high-risk applications such as news verification or legal evidence, it is more acceptable to incorrectly identify a genuine video than to miss a DeepFake.
- **Additional Focus:**
  We focused on maintaining a balanced F1-score to enhance accuracy (minimizing false positives) and recall (minimizing false negatives). Balancing these criteria ensures that the model is not too biased toward a single class and remains effective in a variety of contexts.

6. Results and Insights

The performance evaluation found significant differences between the baseline and improved models:

6.1 Baseline Model: XceptionNet.

The solo XceptionNet model, trained just on individual frames, does quite well at recognizing DeepFakes. It successfully caught spatial abnormalities such as uneven texturing, odd mixing, and pixel-level distortions within a single frame.However, because it examined each frame separately, it was unable to use sequential or temporal information, such as anomalies in facial animation or odd transitions between frames.
As a result, the baseline model struggled to recognize more sophisticated deepfakes in which individual frames seemed genuine but anomalies emerged only after monitoring sequences over time. This constraint resulted in decreased recall and a considerably worse F1-score for the false class.

6.2 Primary Model: LSTM

The hybrid architecture, which included XceptionNet for spatial feature extraction with LSTM for temporal modeling, showed substantial gains across various assessment metrics:

- **Better Detection of Subtle Manipulation:** The model may identify small manipulations, such as irregularities in blinking, lip movements, or head positions, by analysis of frame sequences.
- **Improved Recall and F1-Score for the Fake Class:** Adding LSTM improved the model's recall and F1-score for detecting false movies. The F1-score also rose, indicating a better balance of accuracy (fewer false positives) and recall (fewer false negatives).
- **Confusion Matrix and Insights: T**he confusion matrices for the models offered clear visual evidence.
    - The LSTM-enhanced model has a greater true positive rate for DeepFake detection.
    - False negatives (DeepFakes incorrectly labeled as real) were substantially lower than in the baseline model.

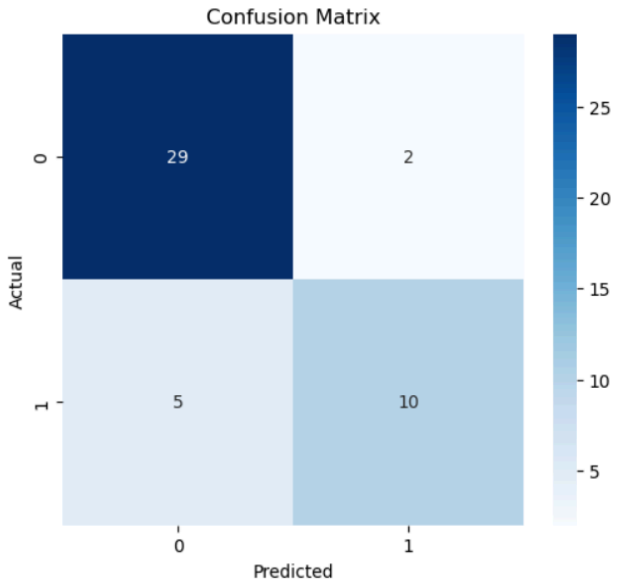

Fig : Confusion matrix for the finalised model

The confusion matrix reveals that the model performs well in detecting fake and real videos. It correctly identified 29 real videos as real and 10 fake videos as fake, yielding a total of 39 correct predictions. However, there were some misclassifications like 2 real videos were incorrectly predicted as fake, and 5 fake videos were incorrectly predicted as real. These errors suggest that while the model shows a good overall performance, there is still room for improvement, especially in distinguishing between real and fake videos.

6.3 Overall Insights

Integrating temporal modeling and spatial analysis was critical to boosting DeepFake detection accuracy.

While static frame analysis using CNNs is useful to some extent, collecting the temporal development of face characteristics improves the model's capacity to detect manipulation errors that occur over sequences of frames, a major element of many recent DeepFakes.

7. Conclusion

We have created a deepfake detection system that balances spatial and temporal analysis to achieve high performance in discriminating between real and fake movies using a structured and carefully constructed methodology.

The system was able to efficiently capture both frame-level artifacts and sequential inconsistencies that are typical of deepfakes by integrating a strong preprocessing workflow, feature extraction using XceptionNet, and temporal modeling via an LSTM network.

7.1 Key Takeaways

- **Frame-based models alone are inadequate:** Deepfake alterations frequently generate small discrepancies that are not visible in individual frames but become obvious when viewed in succession over time. Temporal modeling is thus essential for increasing detection accuracy.
- **Data augmentation dramatically improved generalization:** Random rotations, brightness changes, and horizontal flips helped the model become more resilient to variations in face emotions, lighting, and position, increasing its resilience when exposed to previously unknown data.
- **Prioritizing recall for deepfakes is crucial in real-world deployments:** In practical applications such as media verification, legal evidence authentication, and disinformation avoidance, missing a phony video (false negatives) might have more serious effects than erroneously identifying a legitimate one. Thus, promoting good recall for the phony class is an important design decision.

7.2 Future Work

While the existing technique produced great results, there are various opportunities for additional improvement:

- **Exploring increasingly complicated structures, such as 3D CNNs:** 3D convolutional networks may extract spatial and temporal characteristics directly from video recordings, possibly yielding even more detailed representations than the two-stage CNN + LSTM technique.
- **Integrating Attention Mechanisms:** Using attention-based models, the system might focus more specifically on suspect locations within frames, such as the lips, eyes, or jawline, where modifications are most visible.
- **Expanding the dataset:** Incorporating a broader range of deepfake generating techniques and larger, more diverse datasets will strengthen the model's resilience to newly emerging manipulation methods, assuring its effectiveness against increasingly sophisticated attacks.

References

1. Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). *DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection*. Information Fusion, 64, 131-148. (https://doi.org/10.1016/j.inffus.2020.03.014)

2. Chollet, F. (2017). *Xception: Deep Learning with Depthwise Separable Convolutions*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1251-1258. (https://doi.org/10.1109/CVPR.2017.195)

3. Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). *Joint Face Detection and Alignment Using Multi-task Cascaded Convolutional Networks*. IEEE Signal Processing Letters, 23(10), 1499-1503. (https://doi.org/10.1109/LSP.2016.2603342)

4. Singh, J., & Agarwal, A. (2022). *Deepfake Video Detection Using Deep Learning Techniques: A Review*. Global Transitions Proceedings, 3(1), 98–103. (https://doi.org/10.1016/j.gltp.2022.04.017)