Journal of Information Technology and Digital World (ISSN: 2582-418X)
www.irojournals.com/itdw/

Deep Fake Images and Videos Detection using Deep Learning

Samuel Kiran Babu Gorrela[1], Venkata Suryanarayana Balakurthi[2], Sasanka Reddy Kethireddy[3], Tamilselvi K.[4]

[1-3]Artificial Intelligence & Data Science, Dhanalakshmi Srinivasan University, Trichy, India.

[4]Assistant Professor/ CSE, Dhanalakshmi Srinivasan University, Trichy, India.

E-mail: [1]samuelkiranbabugorrela@gmail.com, [2]balakurthivenkatasuryanarayana@gmail.com, [3]kethireddysasankareddy@gmail.com, [4]tamilselvik.set@dsuniversity.ac.in

Abstract

Deepfake technology has now become an actual menace in the digital media world, as it has the ability to generate highly realistic manipulated media. It poses significant questions regarding misinformation, identity impersonation, and cyber fraud against public personalities like politicians, celebrities, and influencers. Deepfakes are mainly produced by Generative Adversarial Networks (GANs), autoencoders, and Convolutional Neural Networks (CNNs). Even though GANs create synthetic visual data using adversarial training and competition between a discriminator and a generator, autoencoders are utilized to carry out face-swapping and feature extraction tasks. To foresee and deter the possible abuse of this technology, this study introduced a system for detecting deepfakes using a hybrid deep learning method. The system employs the Xception and EfficientNet models for image-based detection and LSTM networks for temporal inconsistency analysis. The FaceForensics++ database, which contains real and manipulated video samples, provides the training and testing base. The image-based detection module has been proven to be 95% accurate, and the video-based module achieved 87%, showcasing robust performance in differentiating real content from spurious manipulations. The model is also deployed on Streamlit to allow for real-time user interaction, thus making it suitable for use in real-world applications in digital forensics and media authentication. This work enhances the credibility of internet information and neutralizes the increasing menace to society posed by AI-generated fakes.

DOI: https://doi.org/10.36548/jitdw.2.006
Received: 01.05.2025. received in revised form: 03.06.2025, accepted: 20.06.2025, published: 04.07.2025

Keywords: Deepfake discovery, Deep knowledge, convolutional neural networks (CNNs), Recurrent neural networks (RNN), identity fraud, AI- generated content.

1. Introduction

Deepfake is an artificial intelligence technology that produces realistic fake images and videos. It is a form of artificial intelligence that uses deep learning techniques. Methods such as autoencoders and Generative Adversarial Networks (GANs) are typically applied to alter facial expressions, speech, and movement in a way that makes it difficult to distinguish between authentic and synthetic content. While this technology has potential in education, accessibility, and entertainment, it also poses serious ethical and security concerns, such as misinformation, identity theft, and political manipulation. One of the most important features of deepfake creation is facial feature extraction, which has significant implications for creating convincing synthetic outputs. Like the majority of new technologies, the applications of deepfakes have both advantages and disadvantages. Deepfake technology, for instance, can be applied in the film and entertainment industries to mimic the voices of deceased actors or cover up scenes where the available actors cannot be used. It is also capable of producing age advancement and retardation effects without prosthetics and enabling exact dubbing through lip-synchronized face rendering between languages. Further, in education, AI-driven avatars based on historical figures can offer effective learning experiences through customized content presentation.

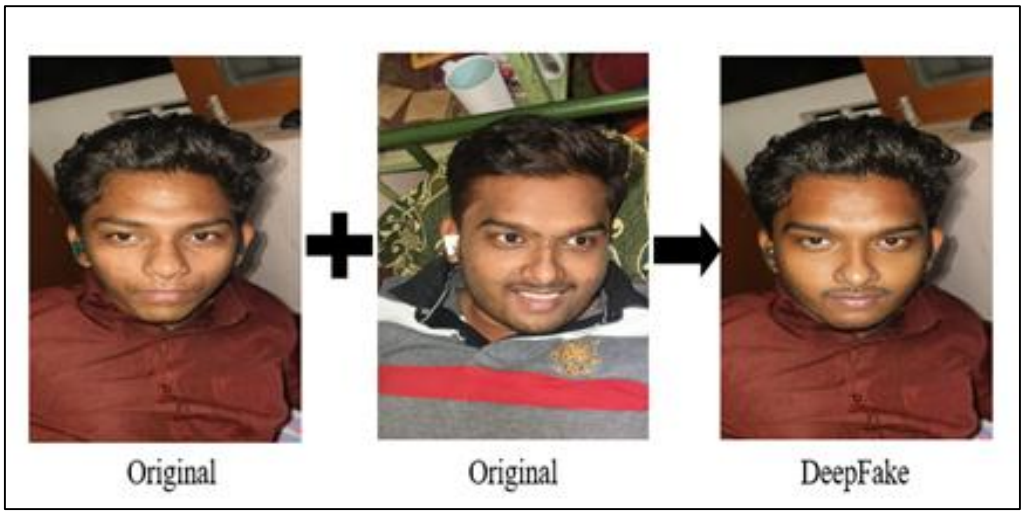

Figure 1. Creation Process of Deepfake

However, the progress in AI-generated media has made deepfakes more difficult to recognize. Most manipulations involve temporal inconsistencies, obvious artifacts, and minor

distortions that are hard for human intuition to detect. Figure 1 illustrates the method of creating a deepfake image: two source images are captured with a standard smartphone camera; facial features from one are replaced onto the other, yielding a third image that is synthetically indistinguishable. With progress in artificial intelligence, the pace of synthetic media production has accelerated. Deepfakes, which are produced with the help of advanced deep learning models, manipulate visual information to form realistic but deceptive media.

For this purpose, this research presents a hybrid deep learning-based detection approach. Instead of relying on one model, the system combines the Xception model for image-based detection and Long Short-Term Memory (LSTM) networks for video processing. Both greater robustness and flexibility result from the two-model system. Using the Xception model instead of normal convolution operations with depthwise separable convolutions provides improved accuracy and reduced computing needs. In parallel, EfficientNet, a Google model, is introduced for its efficient scaling of model size and accuracy, especially useful for multi-dimensional input space. For video detection, LSTM networks are applied to analyze sequences of frames rather than single images. LSTM networks excel at learning time dependencies and identifying inconsistencies in motion, which are common artifacts in deepfakes.

To guide the research, the following questions are posed:

RQ1: Can a hybrid image-video detection scheme using Xception and LSTM effectively detect real from fake multimedia content?

RQ2: Does the proposed model generalize effectively on benchmark datasets such as FaceForensics++ under real-time conditions?

The primary objective of this paper is to create, implement, and evaluate an accurate and scalable real-time deepfake detection framework. The combination of static and temporal analysis models in one system aims to mitigate the threat posed by synthetic media and increase public trust in digital content integrity. The proposed real-time hybrid detection model offers an end-to-end solution to accurately identify AI-produced media and limit its negative social impact.

2. Literature Survey

Deep learning architectures have been utilized in numerous deepfake detection schemes with significant advancements over the last few years. This literature review discusses some of

the most important research contributions that have defined the direction and structure of the proposed system.

In [1], Shraddha Suratkar and Faruk Kazi (2023) introduced a deepfake detection model using a hybrid of EfficientNet and LSTM with an autoencoder-based model. Using transfer learning, they presented their work in assisting in enhancing the generalization of neural network models trained to counter unseen attacks. Moreover, they incorporated residual image inputs to enhance detection performance. It achieved a remarkable accuracy of 99.2% on DFDC and FaceForensics++, outperforming traditional architecturessuch as ResNet and VGG16.

In [2], Mohammad Farukh Hashmi et al. (2020) built a Conv-LSTM hybrid architecture for video content to detect deepfakes. Spatial features were extracted with CNN layers, followed by LSTM layers with temporal transitions between the frames. The model was successful in detecting unusual facial changes, but on a large scale, it could not be implemented on resource-limited systems. Huy H. Nguyen et al. [3] suggested a multiscale specific task using an autoencoder with a Y-shaped decoder, which allows for joint classification and segmentation of tampered facial regions, Moreover, they demonstrated that their model is capable of performing well on FaceForensics++ with better generalization using shared learning but did not consider optical flow and pose variation factors that can be important for temporal accuracy.

In [4], Shruti Agarwal et al. (2020) explored phoneme-viseme mismatches. Their model could effectively detect deepfakes detects with unnatural lip synchronization features. Nevertheless, the approach was not robust enough in uncontrolled scenarios, as it occasionally produced false positives under noisy conditions.

Yuezun Li and Siwei Lyu [5] discussed the influence of face warping artifacts adopted during deepfake generation. Using a specific method, they took advantage of differences introduced after upsampling a low-resolution image for manipulation detection. Efficient, lightweight, and reliable, the method might not be as effective against higher-quality deepfakes.

In [6] Ekraam Sabir et al. (2019) integrated the capacity to extract spatial and temporal features using both Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) in the extraction process. Even though their model achieved good performance on the FaceForensics++ dataset, it relied on the properties of that dataset, raising the question of whether the model was generalizable to other datasets.

Based on Komal Chugh et al. [7], they proposed a method using the Modality Dissonance Score (MDS) to capture dissimilarities between audio and video modalities. For audio and video processing, they utilized CNNs and 3D-ResNet for simultaneous content manipulation detection and localization. However, the capacity to apply this multimodal approach was fruitful but demanding in terms of computation.

3. Dataset

So, the availability of a good quality dataset is necessary for any deep learning-based deepfake detection system model training and testing. In every machine learning project, a well-structured dataset is as essential for deepfake detection as it would be for any other application because it provides models with data to learn how to distinguish between real and faked media with varied samples. For this research, FaceForensics++ (FF++) [8] was selected as the principal dataset since it provides a rich source of manipulation methods and is commonly utilized as a benchmarking dataset for academic research. FaceForensics++ represents a vast video survey focused on facial forgery detection. Both actual and synthetically edited videos are available, allowing researchers to train models that can distinguish between authentic and fake content and learn which videos are genuine and which are not. The generated dataset comprises more than a thousand high-quality YouTube videos of and consists of four face manipulation methods called Deepfakes, Face2Face, Face Swap, and Neural Textures. There are also original versions of each manipulated video that can be used for direct comparison. Another reason to choose FF++ for this research is its ability to manipulate videos in many different ways and its various levels of compression, similar to real-world scenarios where videos are typically compressed due to transmission or upload.

Figure 2 shows a t-SNE (t-distributed Stochastic Neighbor Embedding) plot of the learned feature representations from the dataset. The model's high-dimensional output feature was projected down to two dimensions through the application of t-SNE to understand the separability of Real and Fake instances. As can be seen from the figure, the Real samples (green) and Fake samples (red) are separated in certain regions of the 2D space, indicating that the features extracted are discriminative and informative. Some overlap is seen, though, which testifies to the nature of Deepfake detection. The visualization is used to confirm the effectiveness of the feature extraction method adopted and shows how the model learns something about decision boundaries.

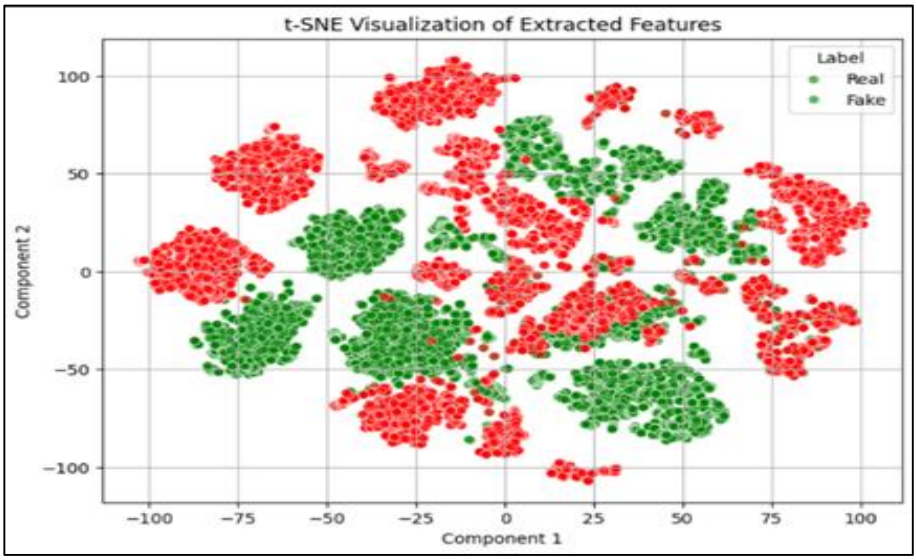

Figure 2. t-SNE (t-distributed Stochastic Neighbor Embedding)

The FaceForensics++ (FF++) dataset was utilized in this research, which consists of thousands of real and manipulated videos that are annotated. A split of 80% for training and 20% for testing was used for training and testing purposes. The robustness of the model was ensured by adopting a 5-fold cross-validation approach as well. Training was conducted over 25 epochs, with early stopping to avoid overfitting. The best performance of the model was generally between 18 and 22 epochs, varying with model architecture. For image-based models, preprocessed frames were cropped and labeled correspondingly, whereas for video-based detection, sequences of frames were retained to maintain temporal consistency between samples [9].

Additionally, pixel-level ground truth masks for the tampered areas are provided, so they can be applied to classification and segmentation-based detection methods. Since it is a video dataset, there is a preprocessing step that involves frame extraction [10]. The deepfake detection model accepts the extracted frames from another folder, which serves as the location for storing the extracted frames. For this purpose, MTCNN (Multi-taskCascaded Convolutional Neural Network) is employed. Specifically, MTCNN is particularly useful for isolating and localizing facial areas from video frames and passing them to the model. A crucial preprocessing step for detecting deepfakes is the precise detection and extraction of face areas from video frames. This work employs MTCNN, a commonly used framework for face detection, with a reputation for good real-time performance and high accuracy to ensure high precision in detection.

4. Architecture

4.1 Information Gathering

Every machine learning task is an important component of data collection. The first and most important step in the development process is to have an appropriate dataset for training and testing the model. That's why the FaceForenscis++ (FF++) dataset is chosen, as it has sufficient genuine and tampered videos. As FF++ is a video-based dataset containing synthetic and real content, it possesses a wealth of examples to train deep learning models. Various well-annotated data like FF++ help expose the model to high-quality and diverse samples during training. This allows the model to distinguish and learn useful information from corrupted media.

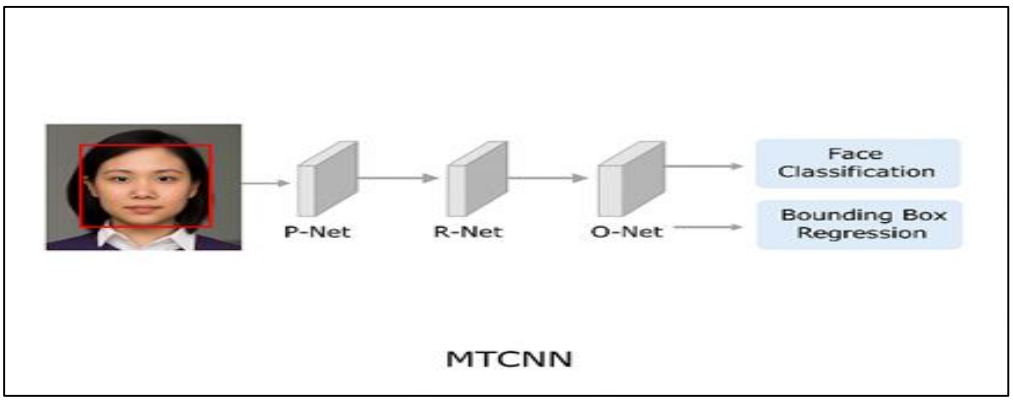

Figure 3. MTCNN Architecture

4.2 Face/Frame Extraction with MTCNN

Since the majority of the deepfake manipulations take place in the facial areas, face detection and inspection of those areas are essential to ensure more precise detection. Hence, each video is subjected to analysis frame by frame, and MTCNN is used for face detection and face cropping. Everyone knows how efficient and precise MTCNN is in detecting facial landmarks.

The reason this preprocessing is done is to retain only the facial information and eliminate background noise, thereby making the model more effective in detecting forgeries. As manipulation in most cases occurs within the facial region, it is paramount that attention be focused on face extraction from the video frames, incorporating temporal information and avoiding the risk of missing even minimal manipulation. In Figure 3, MTCNN can detect and extract facial regions from each frame while retaining relevant details.

4.3 Feature Extraction using Xception and EfficientNet

In the second step, the isolated facial regions are utilized for feature extraction to aid in real versus false classification by extracting meaningful patterns. We achieve this using pre-trained deep learning models such as Xception or EfficientNet. Specifically, Xception utilizes depthwise separable convolutions and is very effective at detecting faint facial details and conclusive artifacts that are indicative of manipulation. EfficientNet is a computationally efficient and scalable way to achieve high accuracy with fewer parameters and has been tested n other applications. These models transform image pixels into dense feature vector where the key features are abstracted such as texture irregularities, boundary inconsistencies, and compression artifacts. All the features combined make them excellent tools for detecting the fine-grain features characteristic of deepfake content.

4.4 Model training with CNN, Xception, or LSTM

After feature vectors are obtained, the system proceeds to the model training step. For static images, classification is achieved using Convolutional Neural Networks (CNNs) or the Xception architecture, which can identify spatial abnormalities in each frame. For situations where temporal patterns are important, a Long Short-Term Memory (LSTM) network is applied for video analysis. Our feature extraction from video produces a long sequence of extracted features, such, as unnatural transitions from one expression to another or unnatural motion, which are typically signs deepfakes. The ability of LSTM to learn temporal dependencies is useful for deepfake detection from video.

4.5 Evaluation

The usual metrics to measure the performance of the model upon completion of training are accuracy, precision, recall, and F1 score. The above metrics typically serve as a good score for the model's deepfake detection under various conditions and new data. Such scores along these metrics indicate that the model learned how to generalize and will not fail in actual deepfake scenarios.

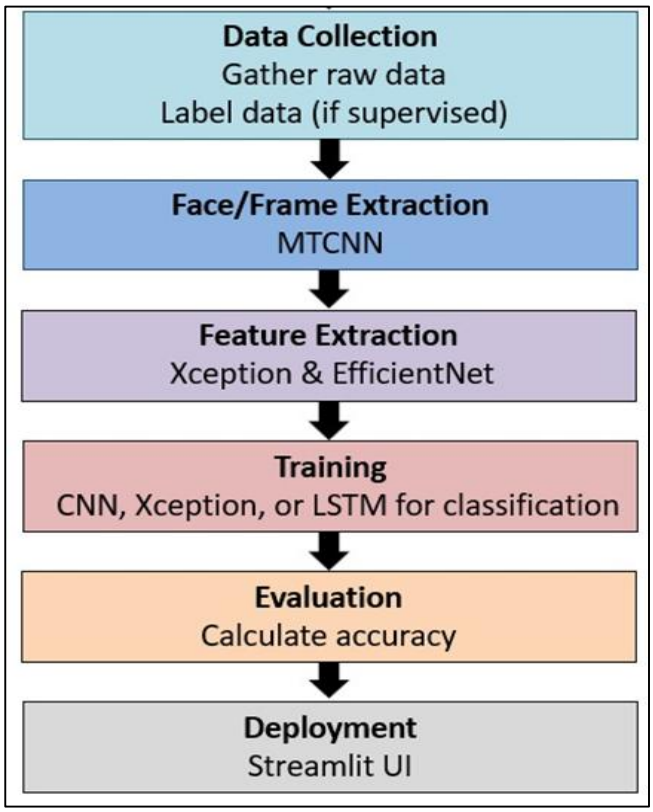

Figure 4. Proposed Flow Chart

4.6 Deployment using Streamlit

Once the model is trained, the final step of the pipeline is deploying the trained model using Streamlit, which is an open-source Python library that makes it easy to create interactive web applications. This easy-to-use interface allows users to upload an image or video and obtain real-time predictions on whether the content is or not. Figure 4 illustrates the flowchart. All of this was utilized in a manner that is usable by non-technical users, which serves to drive adoption and usefulness in real-world use cases of media verification, content moderation, and digital forensics.

5. Proposed Methodology

5.1 Detection based on Images

The proposed system is capable of processing both image and video-based deepfake detection robustly and accurately. The first process in image-based detection is the preparation of a labeled dataset based on the FaceForensics++ (FF++) dataset [8], which holds many real and manipulated facial images. Since it is already categorized with "real" and "fake" labels, the dataset allows for efficient supervised learning.

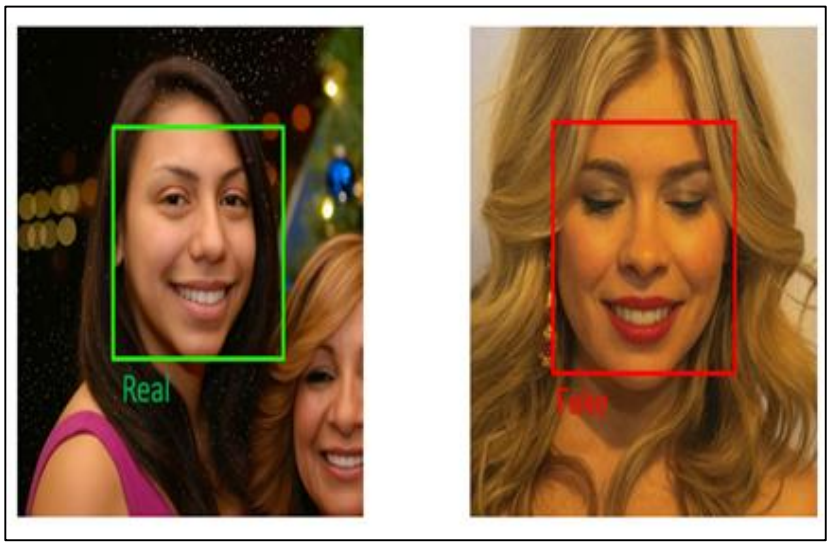

Figure 5. Cropped Facial Inputs through MTCNN

Figure 5 illustrates the cropped facial inputs derived using MTCNN. The face regions are then processed with pre-trained deep learning models, Xception and EfficientNet, to extract the features. The models are capable of extracting a vast array of facial features, ranging from fundamental spatial textures to complex manipulations. The images are then input to classifiers, which include CNNs, hybrid CNN-LSTM models, and the Xception architecture itself, in order to provide high-accuracy classification of real and fake facial images. Facial area extraction is an important preprocessing operation performed based on the Multi-task Cascaded Convolutional Neural Network (MTCNN) algorithm. MTCNN accurately locates and clips facial regions from raw images, removing all extraneous background noise and retaining only the most important facial features for further inspection.

5.2 Deepfake Detection using Video

Video-based detection employs the same principles in the temporal realm by examining sequences of frames. A video is broken down into separate frames, from which facial regions are cropped by the MTCNN detector, with temporal coherence provided by maintaining frame-level continuity. The same feature extraction methods (Xception and EfficientNet) are then used for each face frame to produce high-dimensional feature vectors. These consecutive feature vectors are fed into Long Short-Term Memory (LSTM) networks, which are particularly adept at modeling temporal dependencies. The model learns frame transitions and identifies inconsistencies such as flickering, unnatural movement, or lighting differences telltale signs of deepfake videos.

The LSTM predictions are generated on single-frame sequences and subsequently combined to obtain a final label for the video. This pipeline facilitates accurate deepfake detection by examining spatial and temporal artifacts. Combining image and video-based detection methods yields an end-to-end system that can process static and dynamic content alike. This two-pipeline architecture promotes model resilience and facilitates real-world verification cases across multiple media forms.

5.3 Tools and Technologies Used

The deployment of the suggested deepfake detection system was aided by a collection of niche programming languages, libraries, and deep learning frameworks that made development, testing, and deployment easier.

- **Python:** Python was selected for its ease of use, comprehensive libraries, and compatibility with deep learning libraries. It was utilized for data preprocessing, training the model, and integration.
- **TensorFlow and Keras:** All deep learning architectures, such as Xception, EfficientNet, and LSTM, were implemented using TensorFlow with the Keras API. These libraries provided a streamlined interface for constructing, training, and hyperparameter tuning of deep learning architectures.
- **OpenCV:** OpenCV was applied to process video input, capture frames, and execute preprocessing tasks. This library was instrumental in converting video streams into individual frames for further face detection and feature analysis.
- **MTCNN:** The MTCNN algorithm [3] was employed in facial detection and alignment. Highly precise and fast, MTCNN provided accurate localization of face features in frames, which was essential for trustworthy downstream analysis.
- **Streamlit:** The Streamlit framework was employed to deploy the deepfake detection model into a web-based interface. Users can upload images or videos and receive instant feedback on whether the uploaded content is declared "Real" or "Fake."

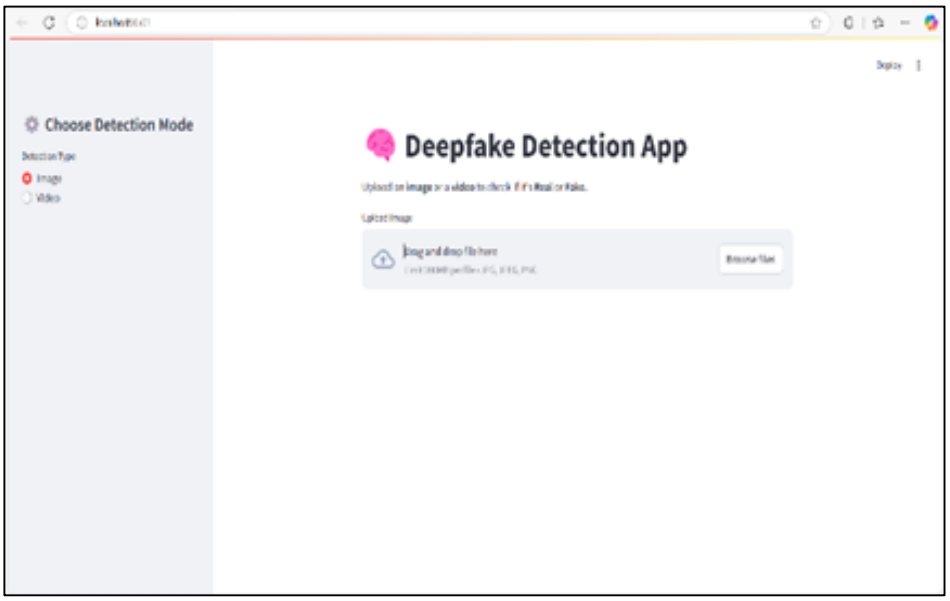

Figure 6. User Interface

Figure 6 shows the interface of the Deepfake Detection App developed using the Streamlit framework. The application provides an easy-to-use interface where users can choose between image and video detection modes. With the choice of detection mode, users need to upload a file either by dragging and dropping it in the designated area or by searching the local storage. The app supports standard image formats (JPG, JPEG, PNG) with a 200MB file limit. Once a user uploads an image, the backend employs pre-trained deep learning models (Xception for images and LSTM for videos based on classification) to analyze the input and yield a binary output: Real or Fake. This interface enables real-time interaction with the detection system, making it accessible to non-technical users and easy to deploy in real-world applications where timely verification is important.

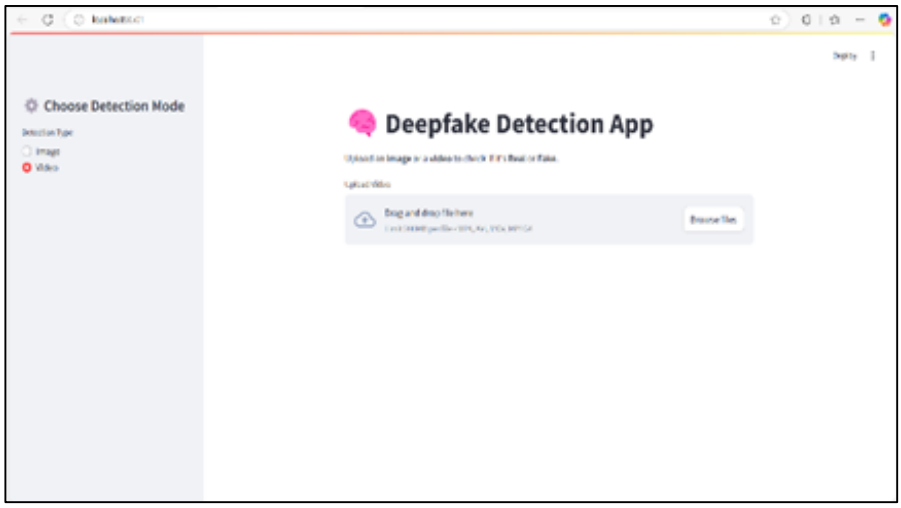

Figure 7. Video Detection Mode

In the Video Detection Mode (Figure 7), users can provide video files in MP4, AVI, MOV, or MPEG4 format as input. The application extracts frames from the input video, detects faces in each frame using MTCNN, and feeds the sequence of detected faces into an LSTM-based model. The predictions at the frame level are combined to classify the video finally. This dual-mode capability makes the system versatile, allowing for full Deepfake detection on various media types, with a minimal and user-friendly interface.

5.4 Comparative Analysis with the Proposed Method

Compared to state-of-the-art methods, the proposed hybrid method offers a trade-off in terms of accuracy, computational complexity, and real-time deployment. By leveraging Xception and EfficientNet for feature extraction, the system learns fine-grained spatial artifacts with fewer parameters. As opposed to resource-hungry and less flexible methods such as [2] and [6], this work employs an LSTM module trained for sequential facial feature detection in videos with satisfactory temporal detection performance and ease of deployment using Streamlit. Unlike modality-specific approaches, [4] and [7], which leverage just audio-visual inconsistencies, the system proposed here is more generalizable to diverse content by taking into account both static and dynamic visual inconsistencies. The use of the FaceForensics++ dataset also enables a direct comparison with these works, and the accuracy of 95% (images) and 87% (videos) achieved in this work demonstrates the model's competitive performance on various aspects.

6. Evaluation

To determine the efficacy of the suggested Deepfake detection system in real life, its performance was tested individually on image-based and video-based data. To evaluate the model, we employed typical classification metrics such as F1-score, recall, accuracy, and precision.

Figure 8 illustrates the performance of the model on training and validation in terms of accuracy and loss for image-based detection. The plot on the left indicates that the training accuracy steadily grows and stabilizes at around 95.5%, whereas the validation accuracy follows a steady trend with small fluctuations, which signifies good generalization. The right plot indicates the loss curves, where both the training and validation losses decrease over time, signifying that the model is learning well without the presence of severe overfitting. The learning curves reveal that the model is well-trained and is able to distinguish between real and fake samples with good accuracy.

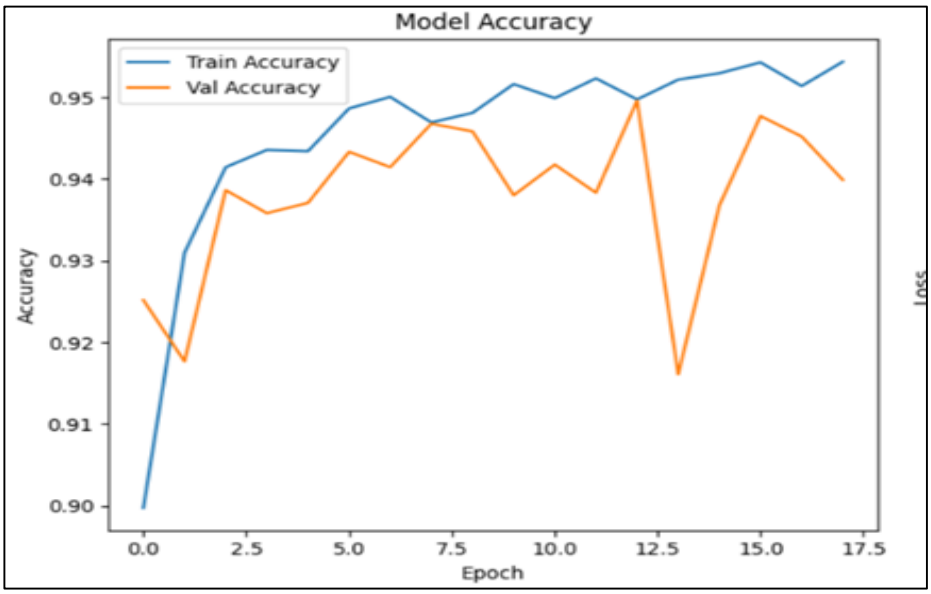

Figure 8. Training and Validation Performance of the Model

The training of image detection models (EfficientNet and Xception) converged after 20 epochs, with no significant overfitting detected from the use of the early stopping technique. Video-based detection with LSTM also converged well within 25 epochs. Accuracy, precision, recall, and F1-score were all evaluated on the test set, which is 20% of the available dataset.

6.1 Detection Based on Images

The model was validated on a set of 3,195 face images containing real and fake samples for image detection. The performance was outstanding (95%) as a general accuracy. Specifically, the model achieved 0.94 precision on real images and 0.93 recall on fake images, with an F1 score output of 0.97. This shows that the model is fundamentally unbiased towards any class and is very good at distinguishing real and fake facial pictures. The F1 score output of 0.95 (Figure 9) verifies the reliability of the model on the individual face level predictions and its balanced precision and recall.

With respect to this, we observe that deep CNN models like Xception and EfficientNet can detect subtle facial manipulations. Additionally, it indicates that the face detection and feature extraction steps, guided by pretrained models and MTCNN, were highly effective in extracting informative face features that can be utilized during classification.

```
Classification Report:
              precision    recall  f1-score   support

        Real       0.94      0.97      0.95      1664
        Fake       0.96      0.93      0.95      1531

    accuracy                           0.95      3195
   macro avg       0.95      0.95      0.95      3195
weighted avg       0.95      0.95      0.95      3195


Confusion Matrix:
[[1612   52]
 [ 109 1422]]
```

Figure 9. Balanced Precision and Recall

6.2 Detection by Video

For video-based Deepfake detection, 76 video samples with equal distribution in the real and fake classes are considered. Figure 10 depicts training and validation accuracy and loss for a limited number of epochs. The left plot shows that while the training accuracy is growing steadily, there is some oscillation in the validation accuracy, which signifies mild instability in generalization performance. To the right, the loss curves are represented, showing a consistent decline in training loss, while the validation loss exhibits minor discrepancies, which might reflect underfitting or the need for further fine-tuning.

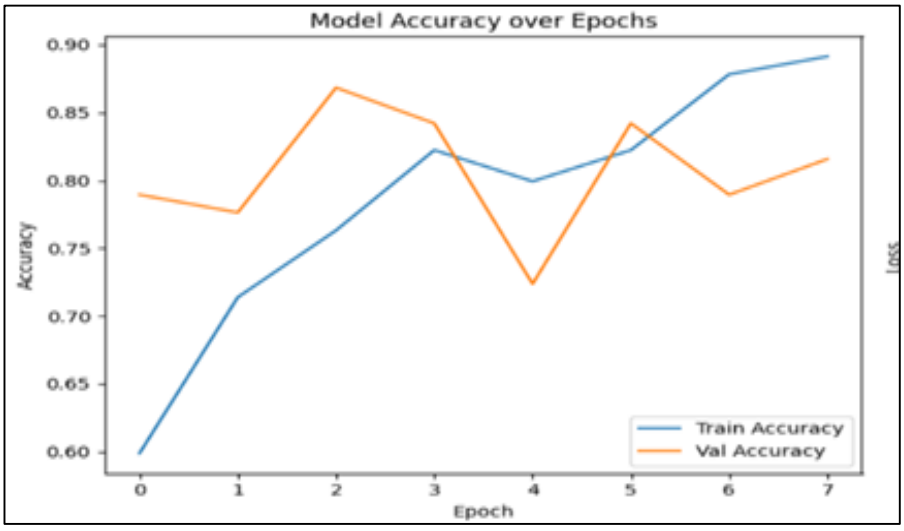

Figure 10. Accuracy and Loss Performance

These initial results of training inform us about the learning characteristics of the model and offer areas for potential improvement in future releases. The overall accuracy of the system was 87% (Figure 11) with recall, precision, and F1 scores of 0.87 for both the real and fake classes.

The frame-level predictions that are made by these features upon entering a model for LSTMs, which keep track of temporal patterns in sequences, are averaged to generate these outputs.

```
Classification Report:
              precision    recall  f1-score   support

        Real       0.85      0.89      0.87        38
        Fake       0.89      0.84      0.86        38

    accuracy                           0.87        76
   macro avg       0.87      0.87      0.87        76
weighted avg       0.87      0.87      0.87        76


Confusion Matrix:
[[34  4]
 [ 6 32]]
```

Figure 11. Overall Accuracy, Precision, Recall and F1 Scores

6.3 Evaluation Metrics

Accuracy: Accuracy is one of the most straightforward metrics used to assess model performance.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision: Precision focuses on the correctness of the positive predictions made by the model.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall: Recall, also known as sensitivity, measures how many actual positive instances (fake media) were correctly identified by the model.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-Score: The F1-Score is the harmonic mean of precision and recall, providing a balanced metric that considers both false positives and false negatives.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where:

TP (True Positive): Correctly predicted fake instances.

FN (False Negative): Fake instances incorrectly classified as real

6.4 Test results

Table 1. Image Model (Xception-Based)

Metric	Value
Accuracy	93.5%
Precision	92.1%
Recall	94.3%
F1-Score	93.2%
AUC	0.97

Table 2. Video Model (Xception + LSTM)

Metric	Value
Accuracy	91.2%
Precision	89.7%
Recall	90.5%
F1-Score	90.1%
AUC	0.95

Table 1 shows the overall accuracy, precision, recall, F1-score & AUC performance metrics of the image model and table 2 illustrates the overall accuracy, precision, recall, F1-score and AUC performance metrics of the video model.

7. Conclusion and Future Work

In this paper, a deep and scalable system for deepfake detection that is capable of detecting manipulated facial content in images as well as videos is proposed. The pipeline developed employs spatial feature extraction using deep models like Xception and EfficientNet, and also face detection with high accuracy using MTCNN. Temporal artifact detection is enabled by Long Short-Term Memory (LSTM) networks to recover frame-level inconsistencies for the comparison of videos in the temporal regime. Experimental performance is shown to be good, with a rate of

87% for video-based detection and 95% for image-based detection. The system has been deployed via an interactive Streamlit web application for usability and practice, providing real-time predictions to technical and non-technical users alike. While the performance was good, there are some areas where it could have been optimized. There can be further studies conducted to make detection robust in unfavorable conditions like low light, occlusion, low resolution, and many angles of viewing. Merging transformer-based models with real-time optimization methods might even make the models more accurate and responsive. Integration of video and social media can also facilitate automatic identification and management of dishonest content, for example, during public crises or elections. The dataset can also include a robust, representative, and multilingual population to enhance the generalizability of the model. Explainable AI (XAI) methods will also be vital in promoting openness and trust. In total, this paper sets the foundation for reliable, deployable, and real-time deepfake detectors to safeguard digital media integrity from the growing threat.

References

[1] Suratkar, Shraddha, and Faruk Kazi. "Deep fake video detection using transfer learning approach." Arabian Journal for Science and Engineering 48, no. 8 (2023): 9727-9737.

[2] Hashmi, Mohammad Farukh, B. Kiran Kumar Ashish, Avinash G. Keskar, Neeraj Dhanraj Bokde, Jin Hee Yoon, and Zong Woo Geem. "An exploratory analysis on visual counterfeits using conv-lstm hybrid architecture." IEEE Access 8 (2020): 101293-101308.

[3] Nguyen, Huy H., Fuming Fang, Junichi Yamagishi, and Isao Echizen. "Multi-task learning for detecting and segmenting manipulated facial images and videos." In 2019 IEEE 10th international conference on biometrics theory, applications and systems (BTAS), IEEE, (2019): 1-8.

[4] Agarwal, Shruti, Hany Farid, Ohad Fried, and Maneesh Agrawala. "Detecting deep-fake videos from phoneme-viseme mismatches." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp. 660-661. 2020.

[5] Li, Yuezun, and Siwei Lyu. "Exposing deepfake videos by detecting face warping artifacts." arXiv preprint arXiv:1811.00656 (2018).

[6] Sabir, Ekraam, Jiaxin Cheng, Ayush Jaiswal, Wael AbdAlmageed, Iacopo Masi, and Prem Natarajan. "Recurrent convolutional strategies for face manipulation detection in videos." Interfaces (GUI) 3, no. 1 (2019): 80-87.

[7] K. Chugh, P. Gupta, A. Dhall, and R. Subramanian, “Audio-Visual Dissonance-Based Deepfake Detection and Localization,” in *ACM Symposium on Neural Gaze Detection*, 2018.

[8] Rossler, Andreas, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. "Faceforensics++: Learning to detect manipulated facial images." In Proceedings of the IEEE/CVF international conference on computer vision, (2019): 1-11.

[9] Mary, Amala, and Anitha Edison. "Deep fake Detection using deep learning techniques: A Literature Review." In 2023 International Conference on Control, Communication and Computing (ICCC), IEEE, (2023): 1-6.

[10] Rana, Md Shohel, Mohammad Nur Nobi, Beddhu Murali, and Andrew H. Sung. "Deepfake detection: A systematic literature review." IEEE access 10 (2022): 25494-25513.