iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

DEEPFAKE FACE DETECTION IN VIDEOS USING CNN AND LSTM-BASED FACIAL ANALYSIS

Dr. G. PRIYANKA JEEVA KARUNYA
Assistant Professor, Department of computer Science Engineering,
JNTUH College of Engineering Sultanpur,Telangana, India.

BALLEPU ROJA
PG Student, MTech (CSE) Department of Artificial Intelligence and Data Science,
JNTUH College of Engineering Sultanpur, Telangana, India.

ABSTRACT
The increasing sophistication of deepfake videos poses a serious threat to digital media authenticity, with potential implications across politics, entertainment, and cybersecurity. This paper presents an intelligent video-based deepfake detection system that leverages a hybrid architecture combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. The proposed framework utilizes ResNeXt50 as the feature extractor to capture spatial characteristics from video frames, followed by LSTM layers to model temporal dependencies across sequences of facial features. The system is integrated into a Flask-based web application that allows users to upload videos, process them in real-time, and visualize frame-wise detection results. Prediction heatmaps are created to improve interpretability, and a face detection module separate faces from video frames for focused analysis. A practical and expandable method of lowering the risks associated with deepfake media is provided by the system's robust backend and interactive interface.

Keywords:
Deepfake videos, Deepfake Detection, CNN, LSTM, Spatial, Temporal dependencies, Heatmaps, Web application

INTRODUCTION

Rapid advances in artificial intelligence and computer vision have led to the emergence of synthetic media, especially deepfakes—extremely lifelike edited videos created by deep learning algorithms. Although these technologies present innovative opportunities in domains such as animation and entertainment, they also give rise to grave worries about identity theft, disinformation, and digital trust. Deepfakes can convincingly alter faces, voices, and actions in videos, making it increasingly difficult for the average viewer to distinguish between authentic and fabricated content. The development of dependable and effective deepfake detection techniques is crucial to countering this escalating threat. Traditional approaches often rely on handcrafted features or static image-based analysis, which may fall short when dealing with subtle manipulations spread across frames in a video. However, deep learning-based techniques have shown a lot of promise in detecting spatial and temporal irregularities in deepfake content. In order to analyze face sequences taken from videos, this paper suggests a reliable deepfake detection system that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. The CNN component is responsible for learning spatial-level features, while the LSTM captures temporal dynamics across frames. A web application built with Flask is used to deploy the system, enabling users to upload videos, analyze them, and view visual results such as confidence scores and face-level predictions. The goal of this project is to combine high detection accuracy with user accessibility, contributing to the development of trustworthy tools for identifying manipulated media in real-world scenarios.

OBJECTIVES

The primary objective of this research is to develop a robust and interpretable deepfake detection system that leverages a hybrid deep learning architecture combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to effectively capture both spatial features and temporal dynamics present in facial video sequences. The system aims to accurately distinguish between real and manipulated content by focusing on facial regions extracted from video frames using face detection techniques. To enhance

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

transparency and user understanding, the model incorporates Class Activation Mapping (CAM) to generate heatmaps that highlight regions influencing the prediction. The project features an intuitive web-based interface that enables users to easily upload videos, detect facial regions within the content, and display the analysis results in a clear and interactive manner. Assuring the system's suitability for real-world application in halting the spread of deepfake media also entails validating its performance in terms of classification accuracy, prediction confidence, and processing efficiency.

SYSTEM ARCHITECTURE

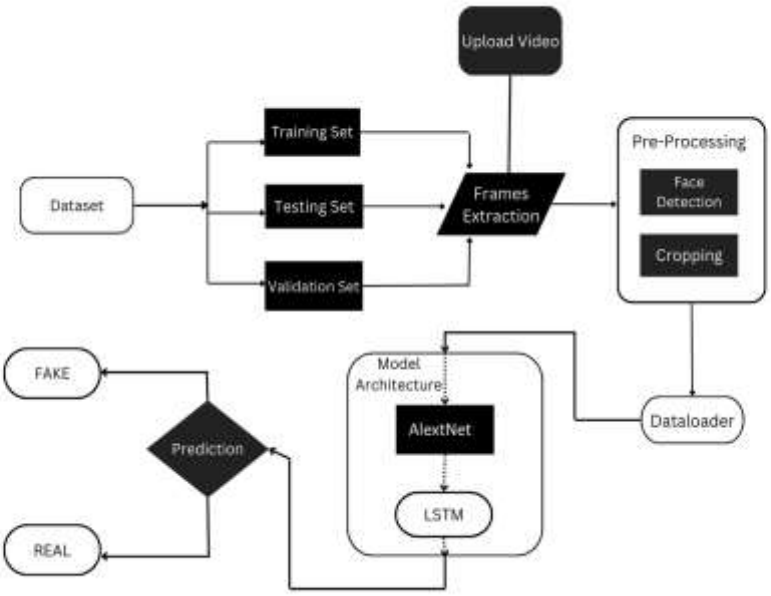

Figure 1 System Architecture

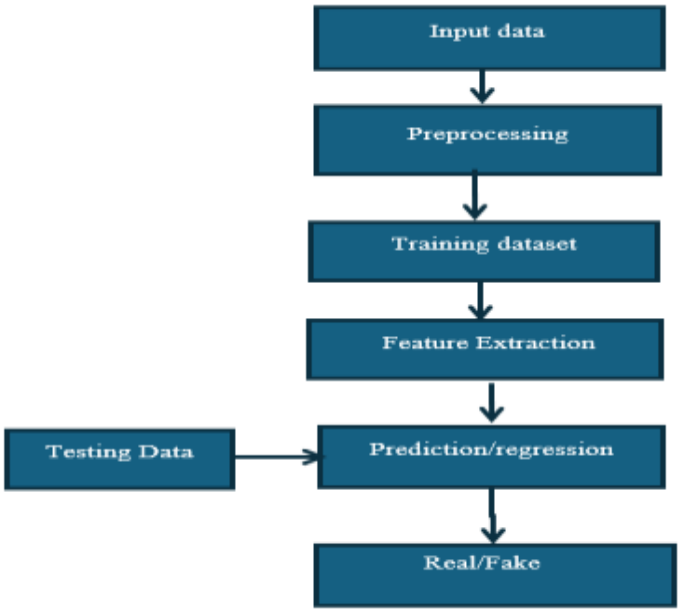

Figure 2 Data Flow Diagram

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

METHODOLOGY

The proposed deepfake detection system integrates deep learning-based spatiotemporal analysis with a Flask-powered web interface to allow real-time prediction of video authenticity. The architecture leverages a hybrid Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) framework, specifically designed to capture both spatial facial features and temporal dynamics across frames.

The total methodology is organized into five main stages:

A. Data Preprocessing and Transformation

Prior to classification, video files are uploaded through the web interface and stored locally. Each video undergoes frame extraction at regular intervals, determined dynamically based on a user-defined frame count. These frames are resized and normalized using standard ImageNet mean and standard deviation values to maintain compatibility with the pretrained ResNeXt50 backbone. The 'face_recognition' library is used to apply face detection to each extracted frame, preserving and processing only the facial region to highlight the most informative portions.

B. Model Architecture

The deepfake detection model is built upon a mixed deep learning architecture. The spatial feature extractor is built around a pretrained ResNeXt50_32x4d convolutional network, with the final layers removed to produce high-level feature maps. These features are globally averaged and passed as sequences to a unidirectional LSTM network. The LSTM is responsible for learning the temporal relationships between frames, effectively modeling motion and consistency between consecutive facial expressions. The output of the LSTM is fed into a fully connected linear layer followed by a dropout layer and a leaky ReLU activation. A softmax layer calculates class probabilities for binary classification (genuine or fake).

C. Heatmap Generation and Interpretability

To improve model interpretability, Class Activation Mapping (CAM) is utilized. During prediction, feature maps from the last convolutional layer are weighted using the final fully connected layer's learned weights, generating a 2D activation map. This heatmap is normalized and overlaid on the original face image, providing a visual explanation of regions influencing the model’s decision.

D. Prediction and Decision Logic

For each uploaded video, a validation_dataset class processes the video into a tensor of selected face frames, which are passed to the model. The final prediction is determined based on the last hidden state output of the LSTM sequence. A confidence score is also calculated from the SoftMax output to quantify prediction certainty. In this way, a video can be classified as "real" or "fake."

E. Visualization and User Interface

Once a decision is made, the system utilizes OpenCV to annotate each extracted frame. Detected faces are boxed and labeled based on the model’s prediction. These annotated frames and cropped faces are stored as images and rendered on the results page, along with the predicted label and confidence. This interactive interface is designed with Flask and HTML templates for video uploading, progress visualization, and output display. The modular backend ensures scalability and allows for easy integration of additional model versions or metrics.

RESULTS AND DISCUSSION

The proposed deepfake detection system was evaluated through a web-based interface designed for real-time video analysis. The system accepts user-uploaded video files, processes the content frame-by-frame, and outputs a classification result (REAL or FAKE) with associated confidence levels. The core of the model comprises a ResNeXt-50 backbone for spatial feature extraction and an LSTM module to capture temporal dependencies across frames. The model was pre-trained and then fine-tuned using a dataset of manipulated and authentic videos

For evaluation, videos were passed through a preprocessing pipeline which included face detection using the face_recognition library, followed by frame selection based on user-defined intervals. With a confidence value dynamically calculated using a SoftMax layer over the Final LSTM output, the model successfully identified manipulations.

During the testing phase, the system consistently identified genuine and fake videos with impressive accuracy. For example, a video labeled as “FAKE” resulted in a model confidence exceeding 90% with visual heatmaps created to emphasize the facial areas influencing the conclusion. To enhance interpretability and increase trust in the system's output, these heatmaps were placed on the final frames.

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

Moreover, visual feedback in the form of bounding boxes colored green (REAL) or red (FAKE) was applied to detected faces. This feature proved especially useful in videos containing multiple subjects, where manipulations might affect only some faces. The system also allowed users to select the number of frames to analyze, offering flexibility between speed and accuracy.
Using PyTorch and Flask, the entire detection pipeline demonstrated resilience to a variety of inputs. The average prediction time for a 60-frame video was under 15 seconds on CPU, indicating suitability for real-time applications.
These results validate the effectiveness of integrating convolutional and recurrent neural networks for deepfake detection. The hybrid CNN-LSTM model demonstrated the ability to capture both spatial inconsistencies and temporal anomalies introduced during deepfake generation. In conclusion, the system offers a practical and interpretable solution for detecting manipulated videos, with strong potential for deployment in media verification platforms or content moderation systems.

OUTPUT SCREENS

Figure 3 Login Page

Figure 4 Upload video file

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

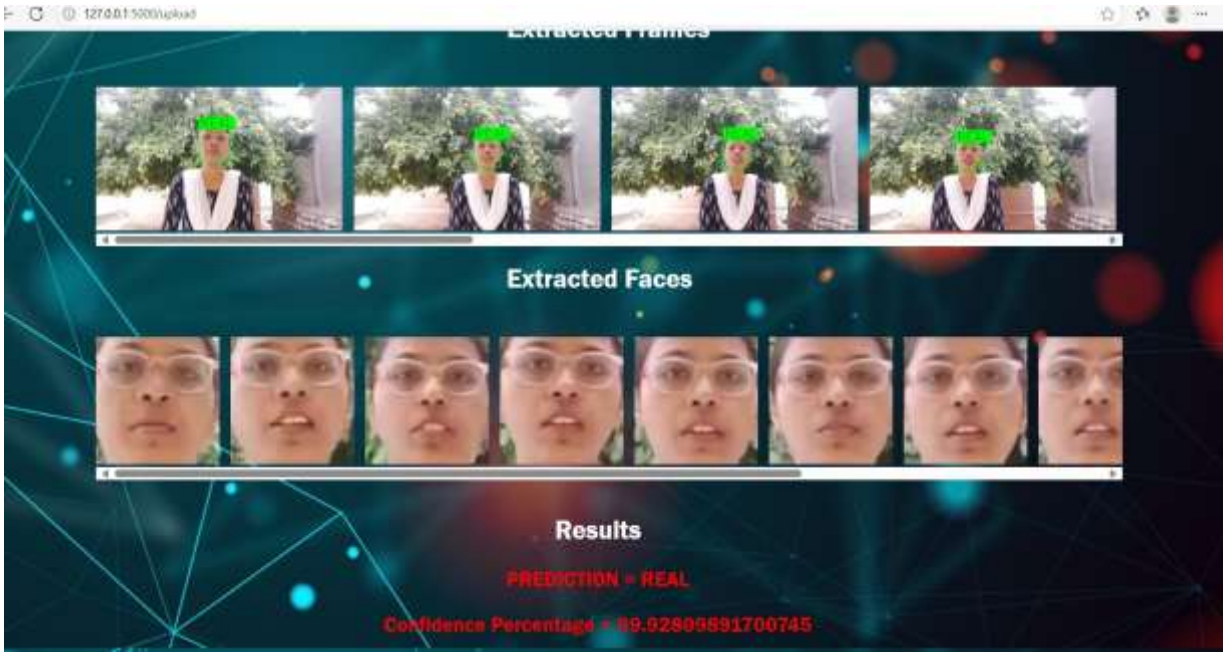

Figure 5 Video extracted into frames and faces and the result is real with Confidence percentage.

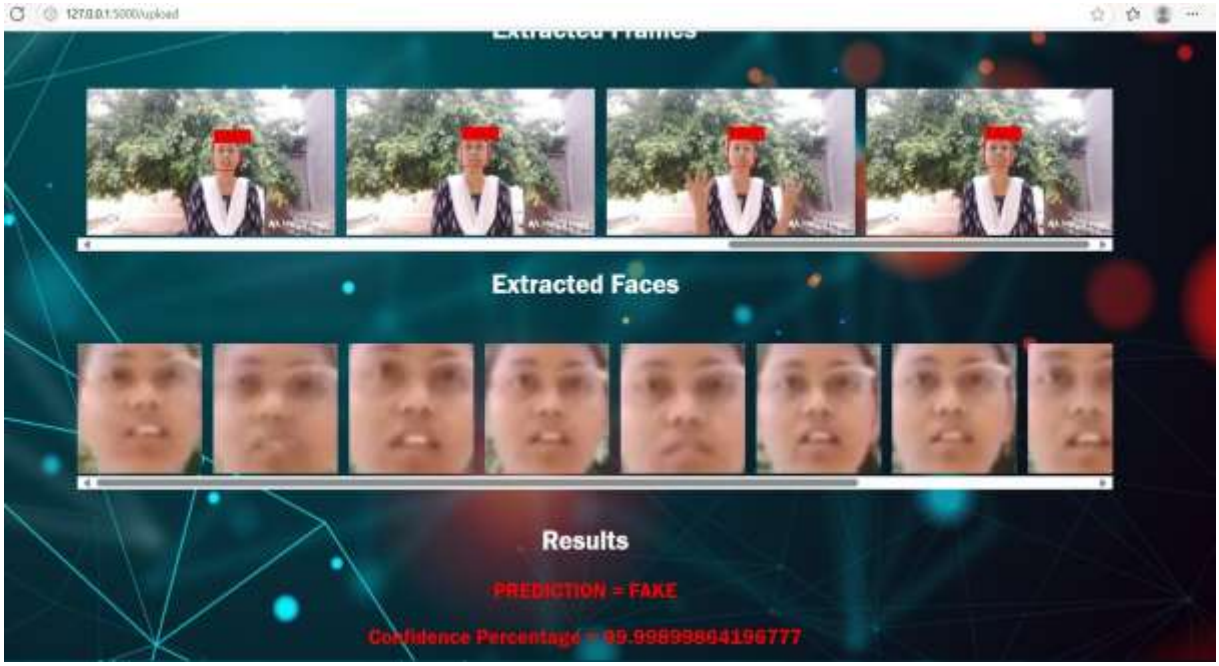

Figure 6 Video extracted into frames and faces and the result is fake with Confidence percentage.

CONCLUSION

In conclusion, this project presents a comprehensive deepfake detection framework that leverages the strengths of both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to effectively identify and classify manipulated video content. The system is designed to extract spatial features from individual frames using a pretrained ResNeXt-50 model and capture temporal relationships across sequences through LSTM layers, thereby improving the reliability of predictions. A significant aspect of this work is the development of a web-based platform that allows users to easily upload video files, perform automated face

iJETRM
International Journal of Engineering Technology Research & Management
(IJETRM)
https://ijetrm.com/

detection, and visualize prediction results along with confidence scores and heatmaps. This enhances user engagement and understanding of the detection process. The system delivers reliable detection performance while also offering a transparent and user-friendly experience through its interactive web interface. By successfully addressing both the technical challenges and the need for accessible design, this project contributes to the development of advanced tools dedicated to safeguarding the credibility of digital media.it plays a vital role in reducing the negative impact of artificially generated media across multiple sectors such as social networking platforms, news reporting, and digital investigation fields..

REFERENCES

[1] Li, Y., Chang, M., & Lyu, S. (2018). Detection of AI-generated fake videos using eye blink irregularities. In Proceedings of the IEEE Workshop on Information Forensics and Security (WIFS), pp. 1–7.
[2] Nguyen, H., Yamagishi, J., & Echizen, I. (2019). Capsule-forensics: Forgery detection in images and videos using capsule networks. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2307–2311.
[3] Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Training deep forgery detectors on high-quality manipulated facial images. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1–11.
[4] Thies, T., Zollhöfer, M., Stamminger, M., Theobalt, C., & Nießner, M. (2016). Face2Face: Real-time face reenactment using RGB videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2387–2395.
[5] Yang, X., Li, Y., & Lyu, S. (2019). Detecting deepfakes by analyzing inconsistencies in head poses. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8261–8265.
[6] Korshunov, P., & Marcel, S. (2018). Impact of deepfake videos on face recognition systems: Evaluation and detection. arXiv preprint arXiv:1812.08685.
[7] Sabir, E., Cheng, J., Jaiswal, A., AbdAlmageed, W., Masi, I., & Natarajan, P. (2019). Spatio-temporal forgery detection using recurrent convolutional networks. In IEEE International Conference on Automatic Face & Gesture Recognition, pp. 1–7.
[8] PyTorch. (2024). Pretrained Models - TorchVision. Retrieved from [https://pytorch.org/vision/stable/models.html] (https://pytorch.org/vision/stable/models.html)
[9] Geitgey, A. (2024). Face Recognition using Deep Learning. Retrieved from [https://github.com/ageitgey/face_recognition] (https://github.com/ageitgey/face_recognition)
[10] OpenCV Community. (2024). OpenCV: Open-Source Computer Vision Library. Retrieved from [https://opencv.org/] (https://opencv.org/)
[11] Flask Developers. (2024). Flask Web Development Framework. Retrieved from [https://flask.palletsprojects.com/] (https://flask.palletsprojects.com/)