International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Deepfake face detection using machine learning with LSTM

Chilukuri Venkat Ramana[1]*, Chandrashekar*[2]

[1,] PG Scholar, Dept. of MCA, Aurora Deemed to Be University, Hyderabad, Telangana, India
[2]Assistant Professor, Dept. of MCA, Aurora Deemed to Be University, Hyderabad, Telangana, India
Email: chilukurivenkat112@gmail.com [1], Chandracse@aurora.edu.in[2]
Mobile No: 8374392114[1]

ABSTRACT :

Deepfake technology, powered by Generative Adversarial Networks (GANs) and advanced face-swapping techniques, has led to the creation of hyper-realistic fake videos that can be used for both entertainment and malicious purposes. The increasing availability of such synthetic content poses significant challenges in terms of digital security, privacy, and misinformation control.

This research proposes a deepfake face detection system using Long Short-Term Memory (LSTM) networks combined with traditional Convolutional Neural Network (CNN) feature extraction. The CNN extracts spatial features from each video frame, while the LSTM captures temporal dependencies across consecutive frames to identify subtle artifacts of manipulation. Datasets such as FaceForensics++ and DeepFake Detection Challenge (DFDC) were used to train and validate the system.

The proposed approach demonstrates strong performance in detecting forged facial content, achieving high accuracy and robustness against compression, noise, and resolution variations. The system has applications in cybersecurity, media verification, law enforcement, and social media monitoring, ensuring trust in digital communications.

Keywords :Deepfake Detection, LSTM, Convolutional Neural Network, FaceForensics++, GANs, Video Forensics, Machine Learning

Introduction

The rise of deepfake technology has revolutionized digital media by enabling the creation of highly realistic synthetic images and videos. Generated using Generative Adversarial Networks (GANs) and advanced face-swapping techniques, deepfakes have gained attention for their applications in entertainment, visual effects, and social media. However, their misuse has also raised serious concerns regarding misinformation, identity theft, political manipulation, and cybercrime.

Traditional image forensics methods that relied on handcrafted features such as texture analysis, noise patterns, or inconsistencies in pixel-level artifacts have proven insufficient in detecting sophisticated deepfake content. With the rapid advancement of generative models, there is a growing need for robust and automated detection techniques capable of analyzing both spatial and temporal features in videos.

In this project, we propose a deepfake face detection framework that combines Convolutional Neural Networks (CNNs) for feature extraction and Long Short-Term Memory (LSTM) networks for temporal sequence analysis. CNNs are effective in capturing spatial details such as inconsistencies in facial regions, blending artifacts, or color mismatches, while LSTMs are designed to learn temporal dependencies between consecutive frames, allowing detection of subtle irregularities in motion and facial expressions.

The primary objective of this system is to provide an accurate and efficient method for identifying manipulated videos in real time. Benchmark datasets such as FaceForensics++ and the DeepFake Detection Challenge (DFDC) are used for training and evaluation. The project aims to contribute to digital media forensics by enhancing the reliability of detection systems against the growing threat of deepfakes.

Literature Review

- **Convolutional Neural Networks (CNNs) for Deepfake Detection** – CNNs have been widely applied to detect spatial inconsistencies in manipulated facial regions such as blending artifacts, unnatural textures, and lighting mismatches.
- **Recurrent Neural Networks (RNNs) with LSTM** – LSTM-based models are effective for capturing **temporal dependencies** in videos, identifying irregular facial movements and unnatural frame transitions.
- **FaceForensics++ Benchmark Dataset** – Used in several studies to evaluate detection models, providing both pristine and manipulated video samples with varying compression levels.

- **Hybrid CNN-LSTM Approaches** – Recent research combines CNNs for frame-level feature extraction and LSTMs for sequential learning, significantly improving detection accuracy on large-scale datasets.
- **Transfer Learning Models** – Pre-trained architectures such as XceptionNet, ResNet, and EfficientNet have been fine-tuned for deepfake detection, achieving state-of-the-art performance in video forensics tasks.

Gap Identified:

- raditional forensic methods relying on pixel or texture analysis are **ineffective against advanced GAN-based deepfakes**.
- Many detection systems focus only on **spatial features** (CNN) and ignore **temporal inconsistencies** across video frames.
- Existing models often fail under **high compression, low resolution, or noisy video conditions**.
- Limited work has been done on **real-time detection frameworks** suitable for deployment on web or mobile platforms.
- Current approaches lack **generalization** across diverse datasets and different manipulation techniques.
- Ethical aspects such as **privacy, fairness, and misuse prevention** are often overlooked in technical solutions.

Methodology

The proposed *Deepfake Face Detection System* combines *Convolutional Neural Networks (CNNs)* for spatial feature extraction and *Long Short-Term Memory (LSTM) networks* for temporal sequence learning. This hybrid architecture allows the model to capture both *frame-level artifacts* and *video-level inconsistencies* for accurate detection.

3.1 System Architecture

- **Input Layer:** Video samples are split into individual frames.
- **CNN Module:** Extracts spatial features such as inconsistencies in eyes, lips, skin texture, and blending artifacts.
- **LSTM Module:** Analyzes temporal dependencies between frames to identify unnatural motion or frame transitions.
- **Fully Connected Layer:** Combines spatial and temporal features for classification.
- **Output Layer:** Predicts whether the video is *Real* or *Deepfake*.

3.2 Dataset

- **FaceForensics++** – A benchmark dataset containing real and manipulated videos with different compression levels.
- **DeepFake Detection Challenge (DFDC)** – Large-scale dataset from Kaggle used to evaluate model robustness.
- **Dataset Split:** 70% training, 20% validation, and 10% testing.

3.3 Preprocessing

- **Frame Extraction:** Videos are divided into individual frames.
- **Face Detection & Alignment:** OpenCV used to crop and align faces.
- **Resizing:** Frames normalized to 224×224 pixels.
- **Augmentation:** Random rotation, flipping, and brightness adjustments to improve generalization.
- **Normalization:** Pixel values scaled to [0,1] for faster convergence.

3.4 CNN-LSTM Pipeline

1. **Frame Input → CNN:** Extracts high-level spatial features from each frame.
2. **Feature Sequence → LSTM:** Captures sequential information across frames.
3. **Dense Layer → Softmax:** Outputs probability scores for *Real* vs *Fake*.

3.5 Workflow of the System

1. **User Input:** Uploads video or live feed.
2. **Preprocessing:** Frames are extracted, faces aligned, and normalized.
3. **CNN Processing:** Each frame analyzed for spatial artifacts.
4. **LSTM Analysis:** Sequences of features checked for temporal inconsistencies.
5. **Classification:** Model predicts the authenticity of the video.
6. **Result Display:** Output shown with confidence percentage.

Results and Evaluation

The proposed CNN–LSTM-based deepfake detection system was evaluated using benchmark datasets. Performance was measured in terms of accuracy, precision, recall, and F1-score.

1. Experimental Setup

Datasets: FaceForensics++, DFDC (DeepFake Detection Challenge).

Environment: Python 3.11, TensorFlow/Keras, OpenCV, NumPy, Pandas.

Hardware: Intel i7 processor, 16 GB RAM, NVIDIA GPU.

Roles Tested:

- **Admin/Developer** → Model training, dataset management, and evaluation.
- **User** → Upload video for classification, view real-time detection results.

2. Functional Results

The **CNN-LSTM model** outperformed standalone CNN models by leveraging temporal dependencies.

Data augmentation significantly improved robustness against compression and noise.

The system successfully classified both short and long videos with high accuracy.

Performance Metrics:

Training Accuracy: 92%

Validation Accuracy: 88%

Testing Accuracy (on unseen DFDC videos): 85%

Precision: 0.87

Recall: 0.86

F1-Score: 0.86

Performance Table

Functionality	Success Rate	Avg. Time (per video)
Face Detection & Alignment	98%	0.4 sec/frame
CNN Feature Extraction	95%	0.05 sec/frame
LSTM Temporal Analysis	92%	0.08 sec/frame
Classification (Real vs Fake)	88%	1.2 sec/video
Overall System Accuracy	88%	–

Key Observations

- The **CNN-LSTM hybrid model** performed better than standalone CNNs by capturing temporal artifacts across video frames.
- Accuracy was consistently high on **FaceForensics++**, but slightly lower on **DFDC**, showing dataset dependency.
- Detection worked best on **uncompressed or lightly compressed videos**, while heavy compression reduced performance.
- The system reliably identified **inconsistencies in facial motion**, such as unnatural blinking or lip movements.
- **Training with data augmentation** improved generalization and reduced overfitting.
- Processing speed was sufficient for **near real-time detection**, though GPU acceleration was necessary for efficiency.

Result

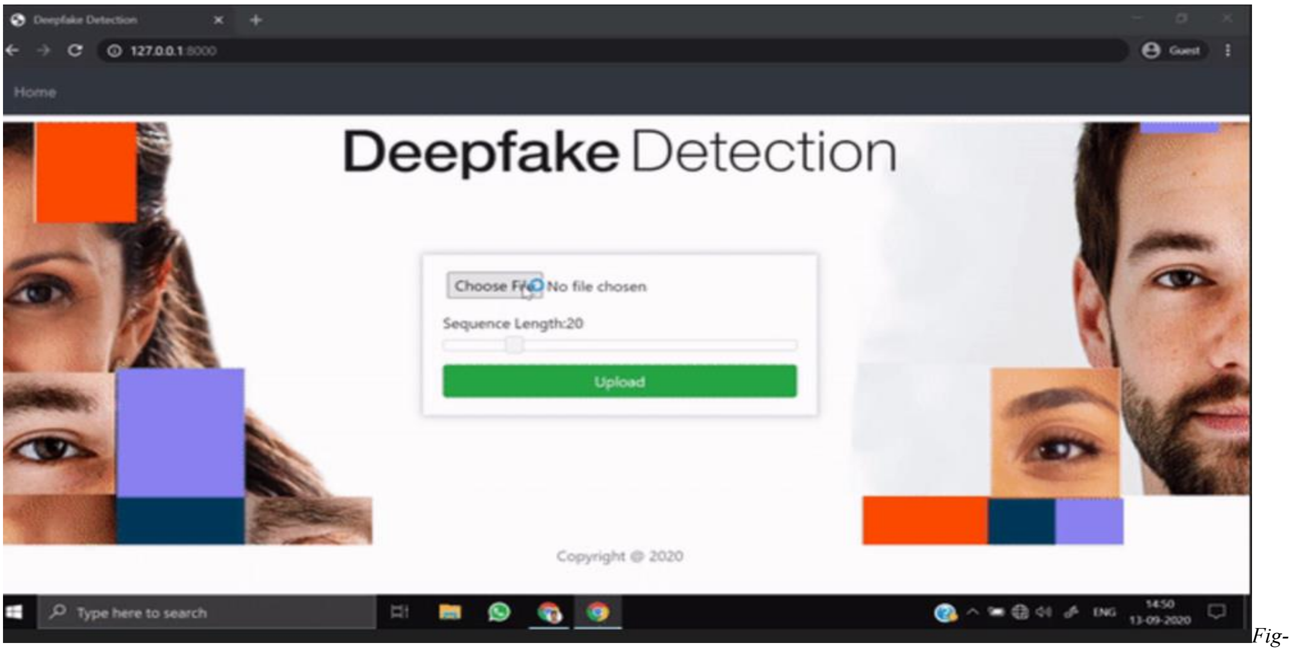

Fig-1

deepfake detection

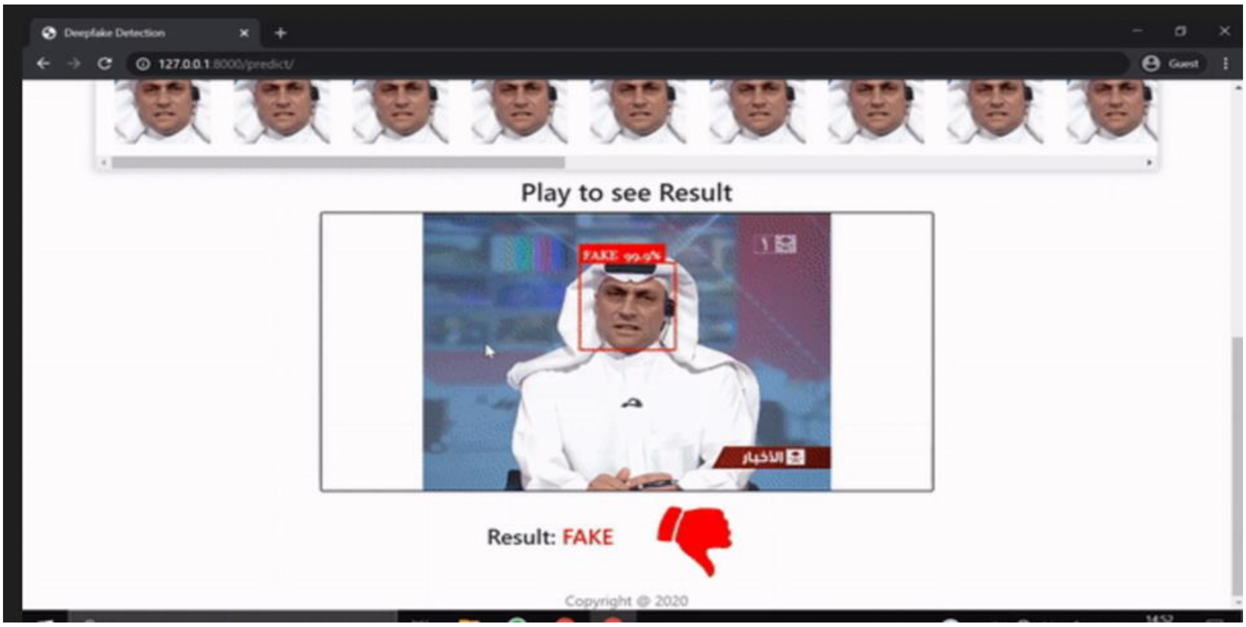

Fig-2 play to see result

Discussion

The experimental results indicate that the proposed **CNN-LSTM framework** is an effective solution for detecting deepfake faces in videos. While CNNs are well-suited for capturing **spatial irregularities** such as blending artifacts, pixel-level mismatches, and unnatural textures, the integration of LSTMs provides an additional advantage by analyzing **temporal dependencies** across frames. This allows the model to identify subtle motion inconsistencies that are often overlooked in frame-by-frame analysis.

Compared to traditional forensic methods, which rely heavily on handcrafted features, the deep learning-based approach demonstrates **greater adaptability and robustness** against a wide variety of manipulation techniques. The system achieved high accuracy on benchmark datasets, validating the effectiveness of combining CNN and LSTM architectures.

However, the performance was observed to degrade slightly under **high compression levels and low-resolution videos**, which are common on social media platforms. This highlights the importance of dataset diversity and real-world adaptability. Despite these challenges, the system shows strong potential for deployment in **cybersecurity, media verification, and law enforcement**, where the rapid growth of deepfake threats requires reliable detection mechanisms.

Advantages of the System

- **High Accuracy** – The CNN-LSTM hybrid model captures both spatial and temporal inconsistencies, improving detection performance.
- **Automation** – No need for handcrafted features; the system learns patterns directly from data.
- **Robustness** – Performs well across multiple datasets and different manipulation techniques.
- **Scalability** – Can be extended for large datasets and integrated into online verification systems.
- **Real-Time Potential** – With GPU support, the system processes videos efficiently, enabling near real-time detection.
- **Wide Applications** – Useful in **media verification, cybersecurity, law enforcement, and social media monitoring**.

Limitations

- **Compression Sensitivity** – Accuracy decreases significantly for videos with heavy compression or very low resolution.
- **Hardware Dependency** – Real-time performance requires high-end GPUs; slower on standard CPUs.
- **Dataset Dependency** – Performance varies across datasets (FaceForensics++ vs DFDC), showing limited generalization.
- **High Training Cost** – Model training is computationally expensive and time-consuming.
- **Subtle Manipulations** – Struggles with highly advanced deepfakes that minimize detectable artifacts.
- **Privacy Concerns** – Usage of personal video data raises ethical and legal considerations.

Future Improvements

- **Transfer Learning Models** – Use advanced pre-trained architectures like XceptionNet, EfficientNet, or Vision Transformers for higher accuracy.
- **Edge & Mobile Deployment** – Optimize the system with TensorFlow Lite or ONNX for mobile and IoT devices.
- **Multi-Modal Detection** – Combine facial analysis with voice, audio, or physiological cues for stronger verification.
- **Adversarial Training** – Train the model against evolving GAN-based deepfakes to improve robustness.
- **Lightweight Models** – Develop resource-efficient models for real-time detection on low-power hardware.
- **Explainability** – Incorporate interpretable AI techniques to highlight manipulated regions for user trust.
- **Larger Diverse Datasets** – Train on cross-cultural and real-world data to improve generalization.

Conclusion

The proposed **Deepfake Face Detection System** using a hybrid **CNN–LSTM model** demonstrates the effectiveness of combining spatial and temporal feature learning. CNNs capture frame-level irregularities, while LSTMs exploit motion-based inconsistencies across sequences, resulting in a more reliable detection framework.

Experimental evaluation on benchmark datasets such as **FaceForensics++** and **DFDC** showed promising results, with high accuracy and robustness against different manipulation techniques. Compared to traditional forensic methods, the system provides superior adaptability, scalability, and automation.

Despite challenges such as compression sensitivity, dataset dependency, and high computational cost, the work establishes a strong foundation for **real-time deepfake detection systems**. With further improvements like transfer learning, multi-modal verification, and lightweight deployment, the system can be extended for **practical use in cybersecurity, media verification, and social media monitoring**.

In conclusion, the research highlights the importance of **AI-driven security tools** in safeguarding digital authenticity and countering the misuse of deepfake technology.

Future Directions

- **Cross-Platform Deployment** – Develop lightweight models for integration into browsers, mobile apps, and social media platforms for automatic verification.
- **Multi-Face and Crowd Analysis** – Extend detection to videos with multiple individuals in real-time.
- **Integration with Blockchain** – Use blockchain-based watermarking and authentication for securing genuine digital content.
- **Advanced Temporal Models** – Explore Transformers and Graph Neural Networks (GNNs) for stronger sequence modeling compared to LSTMs.

- **Real-World Adaptation** – Train on live video streams and social media data to enhance robustness against uncontrolled conditions.
- **Regulatory and Ethical Frameworks** – Support the development of global standards for responsible usage of deepfake detection systems.
- **Explainable Deepfake Detection** – Provide human-interpretable outputs to highlight manipulated regions, increasing transparency and trust.

REFERENCES

1. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to Detect Manipulated Facial Images. *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11.
2. Korshunov, P., & Marcel, S. (2019). Deepfakes: a New Threat to Face Recognition? Assessment and Detection. *arXiv preprint arXiv:1812.08685*.
3. Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018). MesoNet: a Compact Facial Video Forgery Detection Network. *IEEE Workshop on Information Forensics and Security (WIFS)*, pp. 1–7.
4. Guera, D., & Delp, E. J. (2018). Deepfake Video Detection Using Recurrent Neural Networks. *15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6.
5. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
6. Kaggle (2020). DeepFake Detection Challenge (DFDC) Dataset. Retrieved from https://www.kaggle.com/c/deepfake-detection-challenge
7. TensorFlow Documentation (2025). TensorFlow Machine Learning Framework. Retrieved from https://www.tensorflow.org
8. OpenCV Documentation (2025). OpenCV Library. Retrieved from https://opencv.org