International Journal of advancement in electronics and computer engineering (IJAECE)
Volume 15, Issue 1,Jan 2026, pp.106-111, ISSN 2278 -1412

Video-Based Deepfake Detection Using Hybrid Residual And Temporal Methods

[1]Jaya Raj, [2]Swati Khanve, [3]Nitya Khare
[1]M.tech Scholar, [2]Asst. Prof. CSE - SIRTE, [3]HOD CSE - SIRTE
[1]Dept. of Computer Science and Engineering,
[1]Sagar Institute of Research and Technology - Excellence, Bhopal, India

Abstract- *The aim of this study is to evaluate the performance of a hybrid ResNeXt-50 + LSTM deepfake detection model trained on a composite dataset consisting of FaceForensics++, DFDC and Celeb-DF. While ResNeXt-50 generates 2048-dimensional spatial feature vectors and the LSTM layer processes temporal dependencies, this design is optimized for lightweight inference rather than minimal parameter size. In particular, the hybrid framework reduces computational cost during prediction by using a fixed-length sequence of frame embeddings rather than processing full-resolution videos end-to-end. The objective is to create a model capable of generalizing to unseen video manipulations, noisy environments and cross-dataset conditions. The paper presents the methodology, preprocessing pipeline, classification metrics, baseline comparisons and limitations. Future enhancements, including dimensionality reduction and improved temporal modeling, are also proposed.*

Keywords: *Deepfake Detection, ResNeXt, Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) Spatiotemporal Analysis, Frame-level feature extraction.*

I. Introduction

Most of the existing deepfake detection models are made up of standalone CNN or RNN. Both CNNs and RNNs have their own benefits and when combined they can significantly improve the performance of a model [1]. An experiment was done by creating a hybrid model of these two networks and the results were recorded. This paper discusses the process or methodology of the experiment [2] and the results obtained by training the model.

There are many existing models for deepfake detection [3]. Those models were tested against our dataset and their results were recorded and compared with our model's accuracy. The accuracy of the models that performed exceptionally under a single dataset deteriorated when tested on the dataset created by us as that dataset is a combination of three different datasets and contains all kinds of videos.

This shows that the existing models [4] do not perform well on unknown data and our model overcomes that drawback. This hybrid model is designed to work on uncertain data that other models do not do. A further refined version of this model could be used in areas where a tighter level of security is required. This is an important milestone in the field of deepfake detection technology [5].

II. Literature Survey

Deepfake video detection has become a critical research area due to the rapid advancements in generative models such as GANs [6] and diffusion-based architectures. Early approaches to deepfake forensics primarily focused on CNN-based spatial analysis, leveraging models like XceptionNet, ResNet, and EfficientNet performs strongly on FaceForensics++, yet its performance sharply declines under compression or cross-dataset conditions due to its reliance on frame-level features.

Although temporal methods can detect unnatural head movements, blink irregularities, inconsistent lighting, lip-synched mismatch, they often suffer from weak spatial representations and are sensitive to noise and low resolution videos. In addition, most important approaches require stable face tracking and alignment, limiting their usability in the wild scenarios where camera motion or occlusion is present.

DOI Member 168.225.482

To improve robustness, several studies integrated temporal modeling using RNNs, GRUs or LSTMs. Hybrid architectures combining CNN and RNN models have shown promise in balancing feature richness and computational efficiency[7]. Studies have shown that hybrid models outperform standalone CNN or RNN frameworks. Many models perform well under controlled conditions but their accuracy drops by 15 to 30% when evaluated on unfamiliar datasets. Therefore, there is a need for cross-dataset training and feature fusion strategies to enhance resilience.

The shift from GAN, best models to diffusion models has produced the videos with fewer visual artefacts, better temporal algorithms, and highly realistic textures. This has made traditional artifact-based detection less effective. As a result, new research directions include multi model detection, physiological signals, cross model, consistency checks.

The proposed hybrid ResNeXt-50 + LSTM framework directly addresses several of these gaps by combining strong, spatial and temporal analysis while being optimized for real-time deployment or on diverse and merged datasets[8].

III. Proposed Methodology

The proposed deepfake detection system integrates both spatial and temporal information from video sequences and temporal information from video sequences using a hybrid architecture combining ResNeXt-50 for feature extraction and LSTM for sequential pattern learning. This hybrid approach address the limitations of standalone CNN or RNN models by learning texture-level artifacts (spatial) and motion inconsistencies (temporal) simultaneously. The methodology consists of five major stages:

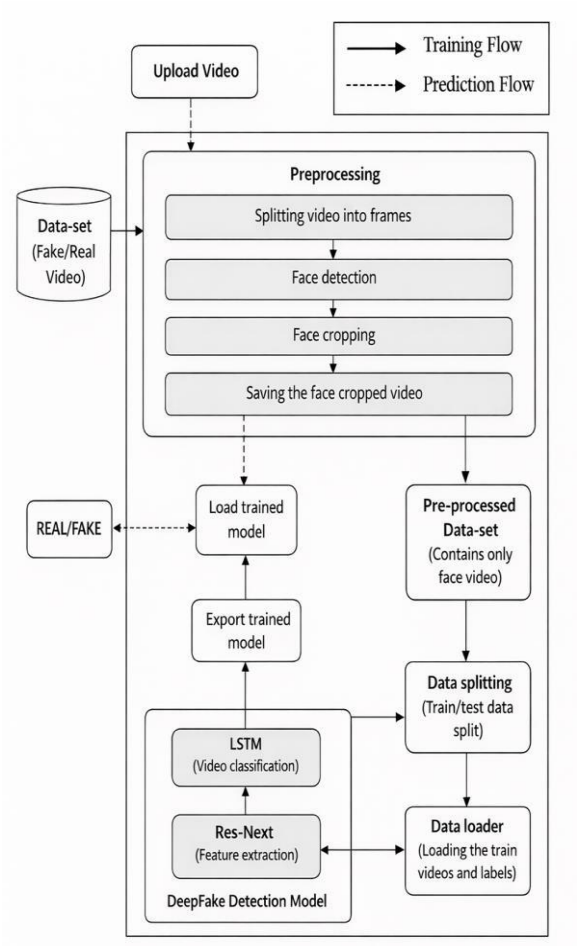

Figure 1 Proposed System Architecture

- **Video Pre-processing and Frame Extraction**

The first step is to create a dataset on which our model will be trained and tested. This model is expected to work well on real-time data, unknown data, low-resolution data and basically data from almost any source. For achieving this level of versatility, the model needs to be trained on a versatile dataset. Therefore, a dataset was created by combining video samples from three different datasets namely FaceForensics++ [9], DFDC [10] and Celeb-DF [11] in the ratio shown in Table 1.

Table 1 Dataset Distribution

Dataset	**Real Videos**	**Fake Videos**	**Total Videos**
FaceForensics++	1000	1000	2000
DFDC	1500	1500	3000

Celeb-DF	500	500	1000
Total	**3000**	**3000**	**6000**

The DFDC dataset contains videos with some noise which could not be used for the experiment, so a python script was applied to remove such videos. The rest of the video samples were included for the experiment. The resulting dataset consists of a combination of different types [12] of videos now.

The next step is preprocessing of the video samples collected. Each input video undergoes a preprocessing pipeline before being passed into the model. From each 10 second video, 15 evenly spaced frames are extracted. Uniform sampling ensures consistent temporal spacing and avoids bias caused by motion density. 70% videos of this dataset is kept for training and 30% for testing.

MTCNN (Multi-Task Cascaded Convolutional Networks) is used to detect faces, crop and align regions and normalize lighting pose variations. Each extracted frame is resized to 224×224 pixels, normalized and standardized for CNN input.

- **Spatial Feature Extraction using ResNeXt-50**
  A pre-trained ResNeXt-50 (32×4d) CNN was deployed for learning from spatial inconsistencies [13]. The spatial component uses ResNeXt-50, a highly efficient CNN architecture based on grouped convolutions. The ResNeXt-50 extracts high-level patterns, identify pixel-level inconsistencies and learns manipulation noise patterns. It generates a 2048-dimesional embedding vector for each frame.

  The output of this module is a feature matrix:
  [15 frames × 2048 features]

- **Dimensionality Reduction Layer**
  To reduce computational complexity a fully connected layer is inserted to compress features . This layer reduces the memory footprint of the LSTM, increases inference speed and helps mitigate overfitting.

  $$2048 \rightarrow 256$$

- **Temporal Pattern Learning using LSTM**
  The LSTM layer receives the sequence of feature vectors in their original time order. LSTMs are effective for capturing frame-to-frame movement, lip-sync inconsistencies, eye-blink irregularities, subtle motion artifacts and temporal coherence violations.

  The LSTM configuration used over here is 2048 units with a dropout rate of 0.4 at 't' seconds with a frame of 't-n' seconds.

- **Classification Layer**
  The final stage uses fully connected dense layer and a softmax activation layer, which generates the output as real or fake based on the confidence scores. An adam optimizer was used for hyper-tuning, which had an adaptive learning rate of le-05, weight decay value of 0.001 and a batch size of 4.

  The output is a binary classification:
  Real (0) or Fake (1)

The pseudocode of the algorithms used in this experiment are discussed in detail below:

Algorithm 1: Deepfake Video Classification
Input: Video V
Output: Class label (Real/Fake)

1. Extract N uniformly spaced frames from V
2. For each frame Fi in the extracted frames:
   a. Detect face region using MTCNN
   b. Align and resize Fi to 224×224 resolution
   c. Compute embedding Fi = ResNeXt50(Fi)
3. Form sequence S = [E1, E2, …, EN]
4. If dimensionality reduction enabled:
   S = Dense_Layer (S)
5. Pass sequence S into the LSTM network
6. Obtain temporal representation H
7. Pass H through a fully connected layer
8. Apply softmax to get probability scores P
9. If P(fake) > P(real) → return Fake
   Else→ return Real

Algorithm 2: Training Procedure
Input: Training dataset D, learning rate lr
Output: Trained model M

1. Initialize ResNeXt-50 weights (pretrained at ImageNet)
2. Initialize LSTM weights randomly
3. For each training epoch:
   a. For each batch of video samples:
      i. Extract frames and preprocess
      ii. Generate feature vectors via ResNeXt
      iii. Pass sequence through LSTM
      iv. Compute predicted labels

v. Compute loss using cross-entropy
vi. Backpropagate errors
vii. Update weights using Adam optimizer
b.Compute validation accuracy and loss
4. Apply Early Stopping if validation loss increases
5. Return trained model M

Algorithm 3: Inference Procedure
Input: Unseen video Vtest
Output: Predicted label

1. Preprocess Vtest (frame extraction, face detection, resizing)
2. Generate ResNeXt embeddings
3. Feed embeddings into trained LSTM
4. Compute output probabilities
5. Return class with highest score

IV. Result

The experiment was done with the help of NVIDIA RTX 3060 GPU, Intel Core i7 processor, and 16 GB RAM with PyTorch [14] and CUDA [15]. The dataset of 6000 videos after being split into 70% (4200) for training and 30% (1800) for testing were further divided into epochs or sets of 4200 / 20 = 210 for training and 1800 / 20 = 90 for testing videos in each epoch.
Initially, for the 2nd epoch the validation accuracy was about 54-56%. The performance improved remarkably after epoch 12 and the training accuracy reached till 95.83%. At Epoch 12, the model showed maximum validation accuracy i.e. 88.78%. The results of accuracy of after every epoch is shown in the graph below.

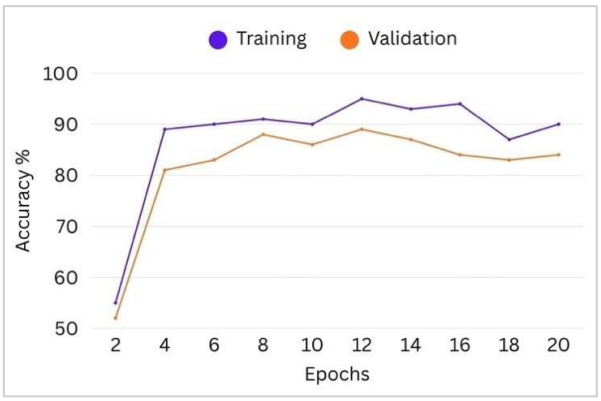

Figure 2 Training and Validation graph

After 20 epochs, the performance of the model was recorded in the table below:

Table 2 Classification Performance Metrics

Metric	Value
Accuracy	88.78%
Precision	85.76%
Recall	86.91%
F1-score	85.33%

TP = 782, TN = 770, FN = 118, FP = 130

Where TP means true positives, TN means true negatives, FP means false positives and FN means false negatives. The calculation of results, ROC curve and the confusion matrix is shown below:

$$Accuracy = \frac{782 + 770}{782 + 770 + 130 + 118} = \frac{1552}{1800}$$
$$= 88.78\%$$
$$Precision = \frac{782}{782 + 130} = \frac{782}{912} = 85.76\%$$
$$Recall = \frac{782}{782 + 118} = \frac{782}{900} = 86.91\%$$
$$F1 - score = \frac{2 \times 85.76 \times 86.91}{85.76 + 86.91} = \frac{7453.40}{172.67}$$
$$= 85.33\%$$

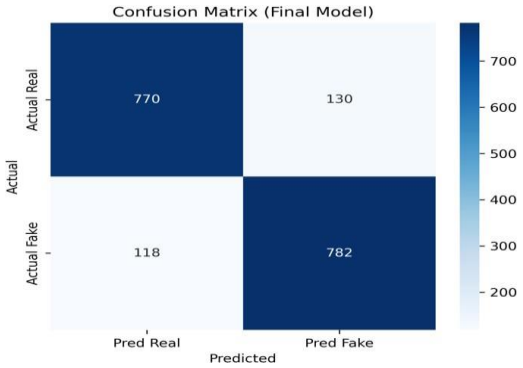

Figure 3 Confusion Matrix

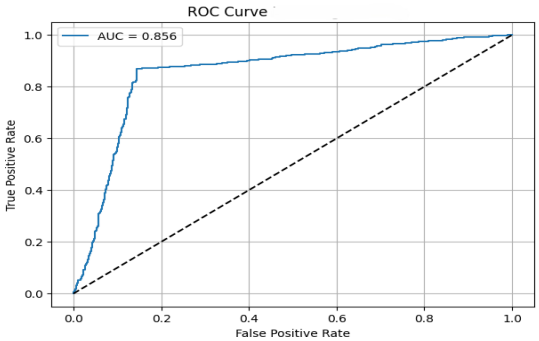

Figure 4 ROC Curve

The performance of other models against our dataset were recorded and the results obtained are shown in the table below.

Table 3 Performance Comparison with Existing Models

Model	Architecture	Spatial Analysis	Temporal Analysis	Accuracy	Real-time Suitability
Xception-Net	Deep CNN	Yes	No	82.0 %	No
ResNet + LSTM	CNN + RNN	Yes	Yes	83.2 %	Moderate
Efficient-Net-B4	Lightweight CNN	Yes	No	78.5 %	Yes
CNN + GRU	CNN + GRU	Yes	Yes	80.3 %	Moderate
Vision-Transformers (ViT-Base)	Transformers	Yes	Limited	84.0 %	No
ResNext + LSTM	ResNeXt-50 (32×4d) + LSTM	Yes	Yes	88.78 %	Yes

V. Conclusion and Limitations

This research demonstrates the effectiveness of a hybrid ResNeXt-50 + LSTM architecture for spatiotemporal deepfake detection. By integrating strong spatial feature extraction with temporal sequence learning, the model achieves 88.78% accuracy, outperforming CNN-only, RNN-only and transformer-based baselines.

Compared to the standalone CNN and RNN, the hybrid model showed a better performance, improvement in robustness, generalization and overall detection performance and thus the model successfully fulfils the research gap and contributes a promising methodology in the domain of deepfake detection.

The model generalizes well across datasets and is lightweight enough for near real-time deployment. Results indicate strong potential for security applications, social media screening and forensic analysis. There are also some limitations in this model such as its inability to work on low quality videos and blurred videos, but with trying other variants and networks these problems can be overcome in the future. This model cannot detect 3D-rendered human avatars and full-body deepfakes yet.

Incorporating transformer based architectures like ViT can offer better detection by capturing long-range dependencies and complex temporal relationships more precisely. Deepfakes are being used for a lot of purposes like blackmailing and frauds. This model will help reduce such problems and make security tighter.

VI. Future Scope

1. Face detection is in trend right now, but there are also full body deepfakes. There is a need to build models that can detect full-body deepfakes, animal deepfakes, landscape deepfakes and in general deepfakes of any types basically.

2. Right now the model is in software form, but it can be integrated as a browser and social media extension or plugin so that a wider range of audience get know about it from word of mouth and because it would become mainstream to use.

3. This model could be integrated into surveillance footages and systems where biometric authentication is required, where a tighter security is required.

4. The model could be trained on data collected from all countries and ethnicities so that it becomes fined-tuned to work on all demographics groups.

5. There is still a huge room for improvement to make it much lighter than the current model so that the model is compatible with most of the low-end devices.

6. If audio detection feature could be integrated in the current model, the results would be more

accurate by matching lip-synchronization and speech patterns.

VII. Acknowledgement

I would like to acknowledge my gratitude towards my guide Prof. Swati Khanve for helping me understand the process of research and for her constant support during this experiment. I would also like to thank my HOD Prof. Nitya Khare for regularly giving me the motivation to keep going and all the faculties of the CSE department that helped during my M.tech endeavour.

References

1) H. Singh, K. Choudhry, A. Kumar, and S. Sharma, "Detecting Digital Deception: A CNN-RNN Hybrid Approach for Deepfake Detection," in Proceedings of the 2025 International Conference on Pervasive Computational Technologies (ICPCT), 2025, pp. 667–672, doi: 10.1109/ICPCT64145.2025.10940830.
2) V. Bankar and R. Pawar, "Hybrid ResNeXt and LSTM Model for Enhanced Deepfake Detection on the FaceForensics Dataset," *International Journal of Engineering Research & Technology (IJERT)*, vol. 12, no. X, pp. 45-52, 2023. Available: https://www.ijert.org/hybrid-resnext-and-lstm-model-for-enhanced-deepfake-detection-on-the-faceforensics-dataset
3) Al-Zahrani, A., & Malik, "*Existing Deepfake Detection Methods with Limitations*". ResearchGate. Retrieved from https://www.researchgate.net/figure/Existing-Deepfake-Detection-Methods-with-Limitations_tbl1_369423547
4) V. L. L. Thing, "Deepfake Detection with Deep Learning: Convolutional Neural Networks versus Transformers," *arXiv preprint* arXiv:2304.03698, 2023. Available: https://arxiv.org/pdf/2304.03698.pdf.
5) M. Abbasi, P. Váz, J. Silva, and P. Martins, "Comprehensive Evaluation of Deepfake Detection Models: Accuracy, Generalization, and Resilience to Adversarial Attacks," *Applied Sciences*, vol. 15, no. 3, p. 1225, Feb. 2025, doi: 10.3390/app15031225.
6) I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," *arXiv preprint* arXiv:1406.2661, 2014.
7) B. A. Kumar, N. K. Misra, N. Pathak, S.-S. Ahmadpour, M. Krishnamoorthy, D. K. Shukla, M. Patidar, and M. Hakimi,"*Hybrid CMNV2: DeepFake faces classification and recognition using deep learning methods*," *Neurocomputing*, vol. 512, no. 4, pp. 220–232, 2024, doi: 10.1016/j.neucom.2024.01.015.
8) D. Xie, P. Chatterjee, Z. Liu, K. Roy, and E. Kossi, "DeepFake Detection on Publicly Available Datasets using Modified AlexNet," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, Canberra, ACT, Australia, 2020, pp. 1866–1871, doi: 10.1109/SSCI47803.2020.9308428.
9) A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics: Learning to Detect Manipulated Facial Images," *ResearchGate*, 2019. Available: **https://www.researchgate.net/publication/330672957_FaceForensics_Learning_to_Detect_Manipulated_Facial_Images**
10) B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The DeepFake Detection Challenge (DFDC) Dataset," *arXiv preprint* arXiv:2006.07397, 2020.
11) Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 3207–3216, doi: 10.1109/CVPR42600.2020.00325.
12) C.-Y. Lin, J.-C. Lee, S.-J. Wang, C.-S. Chiang, and C.-L. Chou, "Video Detection Method Based on Temporal and Spatial Foundations for Accurate Verification of Authenticity," *Electronics*, vol. 13, no. 11, p. 2132, May 2024, doi: 10.3390/electronics13112132.
13) *S. Lyu, "DeepFake Detection: Current Challenges and Next Steps," arXiv preprint* arXiv:2003.09234, 2020. Available: https://arxiv.org/abs/2003.09234**
14) A. Paszke, S. Gross, F. Massa, A. Lerer, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.
15) R. S. Dehal, C. Munjal, A. A. Ansari, and A. S. Kushwaha, "GPU Computing Revolution: CUDA," in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, Greater Noida, India, 2018, pp. 197–201, doi: 10.1109/ICACCCN.2018.8748495.