International Journal of Safety and Security Engineering
Vol. 15, No. 10, October, 2025, pp. 1995-2004
Journal homepage: http://iieta.org/journals/ijsse

A Hybrid Deep Learning Framework for Multimodal DeepFake Detection

Srushti Chikkanna[1*], Chaitra Mrutyunjaya[1], Rajanishree Manjappa[1], Chayadevi Mysore Lakshmanagowda[1], Rohini Thimmapura Venkatesh[2], Sridevi Gereen Murthy[3]

[1] Department of Computer Science Engineering, BNM Institute of Technology, Affiliated to VTU, Bangalore 560070, India
[2] Department of Computer Science Engineering, Dayananda Sagar College of Engineering, Affiliated to VTU, Bangalore 560111, India
[3] Department of Information Science Engineering, Jyothy Institute of Technology, Bangalore 560082, India

Corresponding Author Email: srushtigowda06@gmail.com

Copyright: ©2025 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/ijsse.151002

Received: 13 August 2025
Revised: 9 October 2025
Accepted: 20 October 2025
Available online: 31 October 2025

Keywords:
convolutional neural network, multi-task cascaded convolutional network, long short-term memory, CNN-LSTM architecture

ABSTRACT

DeepFake technology, which uses deep learning to create highly accurate fake images and videos, is posing a growing threat to the integrity of digital media. Available detection methods frequently struggle with real-time flexibility and lack robustness under a variety of manipulations. In order to bridge this gap, this work proposes a hybrid multimodal DeepFake detection system that makes use of a multi-task cascaded convolutional network (MTCNN) for precise facial localization EfficientNet-B0 for efficient spatial feature extraction, and long short-term memory (LSTM) networks for temporal anomalies in videos. The suggested approach outperforms well-known baselines like XceptionNet and CapsuleNet, achieving 97.9% accuracy and computing efficiency. These results confirm that the system is resilient and scalable for real-world applications. All things considered, this work offers a lightweight high-performance DeepFake detection pipeline that maintains confidence in visual content and improves the reliability of digital forensics.

1. INTRODUCTION

Recent developments in deep generative models, particularly generative adversarial networks (GANs), have facilitated the creation of remarkably lifelike synthetic media or DeepFakes. These edited films and images, which closely resemble the appearance and behavior of real people, pose a serious threat to digital authenticity, public trust, and information integrity. In sensitive industries like politics, the media, and cybersecurity, where spreading fake or impersonated content can have serious social, legal, and ethical repercussions, the risks are especially apparent. This problem is exacerbated by the widespread use of DeepFake generating technologies and the large volume of publicly accessible multimedia content, especially that of public figures. In particular, social media plays a significant role in the rapid dissemination of this type of content. Despite the fact that there are numerous DeepFake detection methods, the ones that are currently in use frequently have serious drawbacks, such as poor generalization to invisible manipulation techniques, high computational complexity, and inadequate performance in real-world scenarios like low-resolution inputs, video compression, and occlusion. Additionally, a lot of systems aren't able to function in real-time, which restricts their application in real-world scenarios like surveillance or live media analysis. Even though DeepFake detection has made notable progress, current research still faces significant obstacles with regard to real- time performance, computational economy, and generalization. Many convolutional neural network (CNN)-based methods ignore the temporal discrepancies seen in edited videos in favor of concentrating only on spatial information. Despite their strength, transformer and GAN-based models are frequently computationally demanding and unsuitable for real-time implementation. Furthermore, low-quality situations, including compression, occlusion, and fluctuating illumination, are likely to cause current approaches to deteriorate. These drawbacks highlight the necessity for a powerful yet lightweight hybrid system that can satisfactorily detect picture and video forgeries in real-world settings. In order to tackle these problems offers a strong and effective DeepFake detection system that integrates temporal and spatial analysis. The suggested approach models temporal inconsistencies across video frames using long short-term memory (LSTM) networks, extracts high-resolution spatial features using EfficientNet, and accurately localizes faces using a multi-task cascaded convolutional network (MTCNN). Furthermore, the integration of adversarial training and attention mechanisms enhances generalization and robustness across a variety of DeepFake generating methods. After being trained on a sizable and a variety of datasets, including both real and altered media, the system's overall detection accuracy was about 97.9%. End users can do real- time analysis and visualization on both image and video inputs thanks to the framework's integration into a web-based interface, which facilitates practical deployment. The main goal of this work is to create a hybrid DeepFake detection

system that combines LSTM for temporal video analysis, EfficientNet for compact and effective spatial feature extraction, and MTCNN for precise face localization. This work is motivated by the concept that detection accuracy and generalization can be greatly enhanced while retaining real-time performance by integrating spatial and temporal modeling inside a lightweight architecture. In order to improve resilience against various DeepFake creation strategies, the system also integrates adversarial training and attention processes. This study highlights the significance of responsible development, regulation, and distribution of detection technologies while also addressing the wider ethical implications of synthetic media, in addition to its technological contributions.

They evaluate models including CNNs, RNNs, and transfer learning frameworks, and divide detection techniques into image-based, video-based, and hybrid approaches. Preprocessing techniques that are necessary to improve detection performance, such as face alignment and frame selection, are also included in the review. draws attention to the main issues facing the area, such as the absence of defined benchmarks, real-time performance limits, and generalization across datasets. Their study emphasizes the necessity of advanced, adaptable detection methods to deal with temporal and spatial irregularities in synthetic media [1]. The multi-modal, multi-scale transformer framework (M2TR) detects differences in transformed information by using visual and aural cues. The model enhances generalizability and resilience across various DeepFake manipulation methods by combining audio-visual fusion with multi-scale Vision Transformers, which provide features at various resolutions [2]. By highlighting the need for more diverse and richer datasets and the use of transfer learning it tackles the persistent problem of the rapid advancement of DeepFake generation techniques, which outpace detection advances and accelerate the development of safe generalized models [3]. To enhance the ability to identify temporal and spatial disparities in DeepFake content, the dual attention network (DAN) for facial forgery detection in movies was introduced. The technique makes it easier to identify traits unique to forgeries in a variety of video clips [4]. By increasing feature reuse and CNN structure depth, the model aims to capture fine-grained face manipulation information, especially in high-resolution images. By emphasizing localized facial artifacts and improving feature reuse, it is possible to demonstrate increased accuracy and efficiency. The study highlights the significance of preprocessing methods that assist the model in focusing on the most pertinent facial regions, such as face cropping and alignment [5]. Models can now concentrate on perceptually important facial regions thanks to Cyborg, a technique that integrates human saliency maps into the training loss. The basic idea is that when evaluating a face's legitimacy, people frequently concentrate on particular facial features like the lips, eyes, and forehead. By integrating these saliency-based attention maps into the model's loss function, the detection network is encouraged to learn features from regions that are crucial for human perceptible and forgery recognition [6].

2. RELATED WORK

CNN hybrid models, which combine temporal and spatial data processing, are one type of deep learning architecture that has gained popularity due to recent advancements in DeepFake detection. Numerous studies have suggested innovative methods that use face landmarks frequency domain data and attention mechanisms to differentiate between real and fake faces. The growing complexity of generative models and the widespread dissemination of fake content on digital platforms have made deepfake detection a significant area of study. El-Gayar et al. [7] modeled facial landmarks as graph structures. Geometric flaws and subtle manipulation patterns that pixel-based models occasionally missed could be found using a graph neural network-based framework. Zhang et al. [8] reported that optimization techniques like Particle Swarm Optimization (PSO) have also been used to improve temporal consistency and model generalization under dynamic video settings. Similar to the studies in the audio domain that used the PartialSpoof dataset, focused on segment-level and frame-level speech analysis to identify DeepFakes in artificial audio.

Waqas et al. [9] showed that adding GAN-generated synthetic images to training pipelines increased model resilience to unobserved manipulation strategies, improving robustness and generalization. A thorough analysis of DeepFake detection techniques for photos and videos was carried out by Malik et al. [10], who divided the techniques into feature-based, model-driven, and hybrid frameworks. In a similar vein, Rana et al. [11] offered a comprehensive assessment of the literature emphasizing the necessity of explainable AI, cross-modal fusion, and defined standards to enhance deployment reliability in the real world.

Numerous investigations revealed serious issues with generalization and processing efficiency despite encouraging outcomes. According to Patel et al. [12], many high-accuracy models saw performance loss when subjected to real-time limitations, noise, and compression. Additionally, Alnaim et al. [13] showed that occlusions, such as face masks, significantly decreased the efficacy of traditional CNN-based detectors by concealing important facial features like the mouth and nose.

Cunha et al. [14] suggested a temporal DeepFake detection system that integrated deep neural networks with PSO to enhance temporal consistency and generalization under dynamic video situations in order to combat video-based manipulation. Karaköse et al. [15] investigated DeepFake detection in medical photos in specialized domains and demonstrated that, despite the need for high reliability and resilience, CNN-based techniques were crucial for preserving diagnostic integrity.

Attention has also been drawn to hybrid and adversarial methods. A GAN-based detection method was presented by Preeti et al. [16], and adversarial training enhanced the model's capacity to generalize to previously unseen forgeries on social media platforms. A systematic review by Abbas and Taeihagh [17] demonstrated that whereas temporal and multimodal approaches increased robustness, they frequently resulted in increased processing overhead and latency.

The effectiveness of transfer learning-based spatial models has kept them in widespread usage. In video-based DeepFake detection tasks, Suratkar and Kazi [18] showed that pre-trained CNNs performed competitively. Fine-grained face aberrations in modified photos were successfully caught by InceptionNet-based architectures, as demonstrated by Theerthagiri and Nagaladinne [19]. In the field of audio forensics, Mcuba et al. [20] examined how deep learning models affected the identification of DeepFake audio and identified issues with recording variability and noise sensitivity.

Overall, earlier research showed a definite trade-off between generalization, computing cost, and accuracy. Temporal-based techniques captured motion irregularities at a greater computational cost, spatial-based techniques were effective but less reliable, and hybrid or multimodal systems

increased complexity while improving detection reliability. Inspired by these results, the current work suggests a lightweight hybrid CNN–LSTM framework that combines LSTM for temporal modeling, EfficientNet-B0 for efficient spatial feature extraction, and MTCNN for face localization to achieve high accuracy while retaining real-time viability.

3. PROPOSED SYSTEM

The proposed DeepFake detection system, which can analyze inputs that are both images and videos, is presented in this section. Excellent detection accuracy and generalizability across a variety of manipulated information types are achieved by the suggested method, which employs a hybrid deep learning framework intended to detect false material in both images and videos. The CNN, MTCNN, LSTM, and EfficientNet networks are all combined. This work proposes a hybrid face forensics framework that combines CNN-based facial analysis with general-purpose photo forensics to improve manipulation detection across different DeepFake generating techniques. Accurate and real-time classification of manipulated data is made possible by the suggested DeepFake detection system, which uses a hybrid deep learning pipeline that combines temporal and spatial modeling. The design's scalable and effective modules can handle both image-based and video-based inputs, which makes it appropriate for use in practical situations like media verification and forensic investigation. Face detection, spatial feature extraction, temporal modeling, and classification are the four key components of the system's modular, multi-stage design, which is coordinated to maintain high detection accuracy (97.9%) with minimal computing affordability.

Figure 1 illustrates how the system is set up as a multi-stage pipeline, with data collection and preparation arriving before feature extraction, classification, and real-time inference.

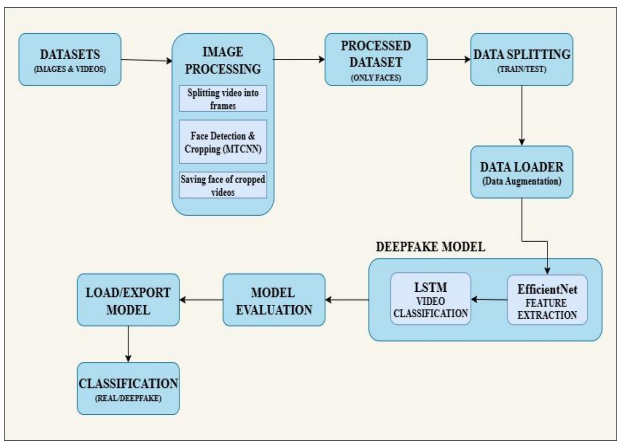

Figure 1. Proposed system architecture

Using the MTCNN, which detects faces reliably in a variety of scenarios, such as changing illumination, poses, and occlusions, the process starts with facial region localization. MTCNN provides precise bounding boxes and facial landmarks that enable accurate cropping of the face regions from pictures or video frames. Frames are extracted at a predetermined rate from visual inputs in order to preserve temporal coherence and offer consistent temporal sampling across sequences. Every recognized face is scaled and normalized to ensure consistency across the dataset and compatibility with the convolutional network downstream. These preprocessing procedures also involve pixel value normalization and standard scaling to get the data ready for feature extraction while preserving manipulation artifacts that can be important for detection. The EfficientNet-B0 architecture, which was selected because it strikes the perfect balance between accuracy and efficiency, is used to process the cropped face regions after preprocessing. Using compound scaling to continuously modify network depth, width, and resolution, EfficientNet-B0 generates a compact yet expressive model that is perfect for resource constrained real-time inference scenarios. Convolutional layers frequently extract hierarchical spatial data from DeepFake material, capturing subtle manipulation indicators such as anomalies in texture, blurring of edges, or strange lighting patterns. To enhance the detection performance of video data, an LSTM network is fed the spatial information that is extracted from sequential frames. The LSTM component helps the system identify abnormalities in facial behavior, such as anomalous blinking, unpredictable changes in expression, or strange head motions, which are signs of changed video information. It does this by modeling temporal dependencies and motion dynamics across frames. Because of this sequential learning, the framework is able to distinguish between real and synthetic content even when individual frames appear visually realistic. Using the binary cross-entropy loss function, the entire network is trained end- to-end, and effective convergence is facilitated by the Adam optimizer. During training, several data augmentation techniques are used, including brightness modification, Gaussian noise addition, random cropping, and horizontal flipping, to enhance model generalization and lower the danger of overfitting. These additions mimic aberrations found in the actual world and improve the model's resistance to various modification methods and outside noise. The detection pipeline is included in a web-based user interface that accepts both image and video inputs to make the user's accessibility and practical application. To preserve modularity and scalability, the frontend interface is separated from the central backend detection mechanism. The entire system is optimized for implementation on general-purpose computing devices or edge platforms with constrained computational resources, and it offers real-time inference with low latency. The hyperparameters of the suggested hybrid model were empirically adjusted using grid search and validation-based optimization. Because they offered the optimum balance between model complexity and temporal representation, 128 hidden units were selected for the LSTM layer. A 0.3 dropout rate was used to avoid overfitting without sacrificing performance. With a batch size of 32, binary cross-entropy loss, and a learning rate, the model was trained using the Adam optimizer. In order to prevent overfitting, early stopping was used based on validation accuracy. Because of its better efficiency-to-accuracy ratio, the EfficientNet-B0 backbone was chosen over its larger counterparts (B1–B7). Compared to heavier alternatives, its compound scaling technique minimizes parameters while maintaining accuracy, making it ideal for resource-constrained situations and real-time deployment.

3.1 Algorithm

To detect fake facial information in images and videos, the proposed DeepFake detection system combines face detection with deep learning-based categorization. For precise face

localization, it uses MTCNN. A refined EfficientNet-B0 model is then used for preprocessing and classification. The workflow allows for automated annotation of predictions on the processed outputs and guarantees consistent detection across a variety of media formats.

Algorithm

Input: file_path, input_size, prediction_threshold. Output: Annotated images/videos.

1. Load Required Libraries
2. Model Initialization & Pre-processing Setup
   -Load the MTCNN for face detection.
   -Load pre-trained Model.
   -Define the input size.
3. File Acquisition
   -If input is an image, read it from file.
   -If input is a video, open video stream and initialize a video writer for annotated output.
4. Frame Extraction & Processing Loop
   -If input is an image → Process it as a single frame.
   -If input is a video → Process each frame sequentially.
5. Face Detection
   -Apply MTCNN to detect the face region in the frame.
   -For each detected face, extract the bounding box & detected confidence.
6. Pre-processing & Feature Extraction
   For each detected face,
   -Crop the facial patch using bounding box.
   -Resize the patch to the input size.
   -Pre-process the image.
7. Classification & Confidence
   -Pass the pre-processed face patch to the trained model.
   -Obtain the prediction probability (P).
   -If P > prediction_threshold, label as 'Real' else 'Fake'.
   -Compute confidence score as either 'P' for Fake else '1-P'.
8. Annotation
   -Draw a bounding box around the detected face.
   -Annotate the box with the prediction label & confidence score.
9. Output Generation
   -If input is an image → save annotated image to output path.
   -If input is a video → write annotated frame to video writer until all frames are processed.
10. Post-processing (For video)
    -Release the video capture.
    -Close all windows.

4. METHODOLOGY

For reliable categorization of synthetic media, the suggested methodology offers an end-to-end DeepFake detection system that integrates temporal analysis, hybrid deep learning models, and image preprocessing. In order to distinguish between authentic and modified content, the detection system uses sophisticated feature extraction and classification techniques to process both pictures and video sequences. The system starts by acquiring datasets that include samples of both DeepFake and real videos. MTCNNs are utilized for facial recognition and cropping after these films have been preprocessed by being divided into frames. By concentrating just on the areas of the face that are most vulnerable to manipulation, this stage guarantees the elimination of extraneous background features. After the face-cropped frames are arranged, a processed dataset is created and split into training and testing sets. In order to ensure uniformity in input dimensions and format, a data loader module efficiently handles the entry of batch samples to the model during training. DeepFake Detection Model is a hybrid detection model comprising two elements, which are Feature Extraction and Video Classification. CNN- based architectures, such as ResNet or EfficientNet, are used to implement feature extraction, which takes advantage of the facial images to extract high-level spatial characteristics. The convolution operation is mathematically represented as:

$$F = f_{cnn}(I) = \mathrm{Re}\,LU(W * I + b) \tag{1}$$

Eq. (1) uses CNN to analyze visual features in facial images or frames. This equation represents the core operation in each convolutional layer**.** This is essential for detecting tampering artifacts like pixel inconsistencies, blending edges, or color mismatches, and helps to learn the hierarchical features.

For video inputs, sequential dependencies across frames are modeled using LSTM networks. The LSTM computes hidden states over time as:

$$h_t = LSTM\left(x_t, h_{t-1}, c_{t-1}\right) \tag{2}$$

Eq. (2) represents the functioning of an LSTM network, which is a type of Recurrent Neural Network (RNN) particularly effective for sequence modeling**.** It plays a critical role, especially when analyzing video data. Preprocessing and face detection using MTCNN,

$$L_{MTCNN} = \lambda_1 . L_{cls} + \lambda_2 . L_{bbox} + \lambda_3 . L_{landmark} \tag{3}$$

Eq. (3) ensures consistent input across all videos and images. Facial regions are extracted using MTCNN**.** The total loss used for face localization is a blend of classification, bounding box regression, and landmark detection. Binary cross entropy (BCE) loss function, which is used in classification tasks like real and DeepFake,

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i . \log\left(\hat{y}_i\right) + \left(1 - y_i\right) . \log\left(1 - \hat{y}_i\right)\right] \tag{4}$$

Eq. (4) Sigmoid Activation is used in the final output layer to squeeze predictions between 0 and 1. Helps the model to discover the correct probability distribution over two classes (Real/Fake). Accuracy metrics and a confusion matrix are used to evaluate the classification performance of the trained model. The model is used for real-time detection after training and evaluation. Users provide preprocessed media samples, which the trained model uses to generate predictions and classify the input as either Real or Deepfake. Our technology ensures a scalable and precise detection pipeline to handle the evolving issues brought on by DeepFake content by integrating

geographical and temporal data. Images and video frames are categorized as Real or Fake using a deep learning-based binary classification technique. In order to eliminate facial regions from every picture or video frame, the MTCNN face detection algorithm is first used. These cropped face areas are then fed into a trained CNN-based classifier such as EfficientNet. LSTM could be used for temporal learning in videos. The possibility that the input face is authentic is indicated by the model's confidence score, which ranges from 0 to 1. The input is classified as Real if the projected confidence score is greater than 0.5 and as Fake otherwise, using a thresholding technique. This procedure is carried out frame-by-frame for video inputs, and the predictions are visualized by superimposing them on the video frames. Each recognized face is surrounded by a bounding box, and labels with the corresponding confidence percentages are shown on the frame. To ensure a guaranteed consistent classification output that can be used for analysis and reporting, this technique enables automatic, interpretable, and real-time labeling of both pictures and videos.

5. EXPERIMENTAL AND RESULTS ANALYSIS

This section evaluates the DeepFake detection system's effectiveness, robustness, and generalizability. The performance is assessed using a variety of datasets, advanced metrics, and comparison with existing models.

5.1 Datasets

The study's dataset, which includes 2,500 images and 400 videos, is in line with dataset scales frequently utilized in DeepFake detection literature. Similar numbers of altered samples per class are used in earlier studies like FaceForensics++ and Celeb-DF, proving that datasets of this size are adequate for training and assessing lightweight hybrid models. In order to provide sufficient variability between subjects, lighting settings, and manipulation methods, the collection contains balanced genuine and fake examples.

A thorough dataset of both authentic and modified material was gathered and processed using a methodical workflow in order to efficiently train and assess the suggested DeepFake detection algorithm. Datasets were divided 80:10:10 for testing, validation, and training, accordingly. The DeepFake video data (in.mp4 format) was divided into distinct image frames using OpenCV. Several DeepFake generation approaches, including identity switching, expression reenactment, and GAN-based synthesis, are used in the modified samples in our dataset to guarantee reproducibility. The model's ability to generalize to both low-level and high-level facial alterations is evaluated thanks to this diversity. To ascertain whether the video labels were authentic or fraudulent, the metadata related to each film was extracted from a file. Every video had a frame rate of five frames per second. Area- based interpolation was used to resize the video frames in order to obtain the best downsampling. The extracted frames were saved as. png files in folders with distinct names for every movie. All video frames were recovered using a uniform temporal sampling technique to avoid temporal bias. A predetermined number of five frames per film was selected by splitting the entire video duration into equal intervals and sampling one frame from each segment. This ensures that the selected frames reflect different phases of facial movement and prevents the model from identifying patterns specific to any one segment of the video. When utilized for order-based cropping and facial identification, MTCNN guarantees consistency and concentrates on particular face regions. To find one or more faces, each frame was first converted from BGR to RGB before being run through MTCNN. All cropped faces were categorized as real or fake using the metadata labels. The dataset was balanced to maintain an equal percentage of real and fake samples in order to prevent model bias. A balanced, well-labeled, and superior dataset for model training and assessment was guaranteed by this configuration.

5.2 Evaluation metrics

Designing any machine learning or deep learning system starts with model training. Adding a large set of labeled images or frames to a neural network is obligatory to train it to recognize discriminative patterns that can distinguish altered data from authentic information in light of DeepFake detection. In this study, both real and DeepFake images and videos were included in the training dataset. The first step in the preparation process is frame extraction, which creates a collection of picture frames from each video using OpenCV. Depending on the frame rate of the video, a frame is extracted every second to maximize data relevance and storage. To ensure uniformity in proportions throughout the dataset, these extracted frames go through dynamic resizing, where each frame's width is changed in accordance with the original resolution. The MTCNN method, which precisely identifies facial regions, is then used to recognize faces. The bounding boxes surrounding each identified face are expanded to guarantee the preservation of every facial feature. To provide a more focused dataset of facial inputs for later model training, the trimmed faces are then saved as separate images in the pertinent face's subfolder. For the DeepFake detection model to be effectively trained, data must be structured during the label assignment and dataset structuring phases. Following the extraction of the labels, ground truth annotations classifying each video as REAL or FAKE are applied. A down-sampling strategy is used to address any potential class imbalance that might result in biased model learning. This procedure ensures a balanced distribution of classes by randomly decreasing the number of fake face samples to match the number of real face samples. The dataset is separated into training, validation, and test subsets after it has been balanced and arranged in an 80-10-10 ratio. This demonstrates that 80% of the data is used for training, 10% is used for validation to adjust the model's performance during training, and the final 10% is used for testing and assessment. During training, real-time data augmentation techniques are used to improve the model's generalization and resilience to various real-world events. By simulating different facial characteristics and environmental conditions, these augmentation techniques strengthen the model's resistance to noise and hidden patterns. The augmentation pipeline uses rotation (± 10°) to simulate different head orientations, width and height shifts to account for off-centered faces, shearing to handle affine distortions, zooming for different distances, horizontal flipping to prepare the model for mirrored inputs, and rescaling pixel values to [0,1] for improved convergence. Global Max Pooling is used as a feature extractor in the proposed model, which comes after EfficientNet-B0, which was trained on ImageNet. Between the thick layers with ReLU activation, there is a Dropout layer

with a rate of 0.5 to avoid overfitting. Binary classification is made possible by an additional Sigmoid-activated dense layer (Real/Fake). The Adam optimizer, which was chosen for its quicker convergence and adaptable learning capabilities, is used to optimize the model. Binary cross-entropy is the loss function, accuracy is the main evaluation metric, and the learning rate is set at 0.0001. This design guarantees resilience, generalization, and effective training. The model is trained using RGB face photos with a batch size of 32 for a maximum of 20 epochs. While ModelCheckpoint preserves the top performing model, EarlyStopping stops training if validation loss doesn't improve after five epochs in order to prevent overfitting. Accuracy and loss are tracked for every epoch for both training and validation sets after the training data is fed into ImageDataGenerator. Two essential performance plots, Training vs. Validation Accuracy and Training vs. Validation Loss, are utilized to depict training outcomes so as to evaluate the learning behavior of the suggested DeepFake detection model. These charts offer insightful information about the learning dynamics of the model over each epoch.

The accuracy curve can be utilized to evaluate how well the model is learning to differentiate between authentic and fraudulent inputs. Good generalization is shown by an accuracy that increases continuously, and the training and validation curves' gaps are small. By highlighting underfitting, overfitting, and convergence patterns, these graphs assist make sure the model learns efficiently and generalizes well.

Figure 2 shows the trends in training and validation accuracy for the suggested DeepFake detection model. Indicating that the model is successfully learning and fitting the training data, the training accuracy, which is displayed in yellow, gradually rises with each epoch until it reaches roughly 97%. Concurrently, the red-colored validation accuracy increases in tandem and reaches a peak of roughly 93%, indicating that the model works effectively when used with unidentified data.

Over 20 epochs, the training and validation loss trends are depicted in Figure 3. The model appears to be maintaining its generalization capacity and not overfitting, as indicated by the moderate and consistent difference between the training and validation losses. All things considered, the model has been trained effectively, striking a balance between learning from the training data and delivering consistent results on unseen validation data. These metrics help evaluate not only accuracy but also the reliability of the model in imbalanced or ambiguous cases.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

Eq. (5) is utilized to decide the accuracy, which indicates correctness in overall classification, to guarantee that the performance review was thorough. With a 97.9% accuracy rate, the suggested model outperformed baseline models.

The execution of the proposed DeepFake detection model on the test dataset is summed up in the confusion matrix shown in Figure 4. The program correctly identified four real occurrences as real and four phony instances as fake, according to the data. Minor misclassifications did occur, though: one fake occurrence was mistakenly forecasted as real (false negative), and one actual instance was mistakenly predicted as fake (false positive). This balanced distribution demonstrates the model's excellent ability to discriminate between authentic and fraudulent inputs, attaining high recall and precision. The matrix is an essential tool for assessing the efficacy of classification and offers information about areas where minor adjustments could increase accuracy even more.

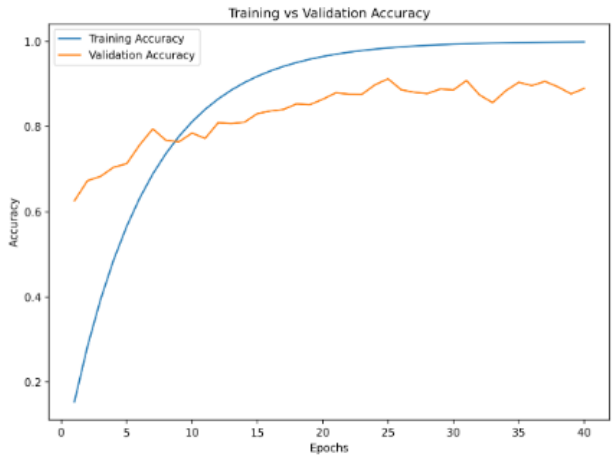

Figure 2. Training and validation accuracy plot

Figure 3. Training and validation loss plot

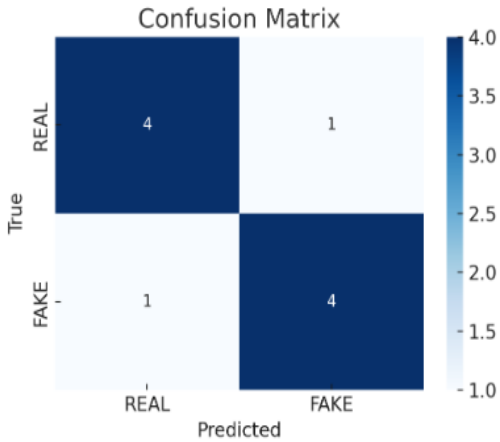

Figure 4. Confusion matrix visualization

5.3 Comparative analysis

A comparative analysis is a methodical process that compares and contrasts various approaches, models, or systems according to predetermined standards or performance indicators. A thorough comparison with current state-of-the-art models was carried out to determine the efficacy and

inventiveness of the suggested DeepFake detection method. The study can successfully highlight the advantages, disadvantages, and improvements of the proposed strategy by examining a number of performance metrics, including accuracy, precision, recall, F1-score, AUC, and model complexity. Comparing several methods makes it simpler to illustrate the benefits or enhancements that the system offers, such as improved generalization, reduced model complexity, or increased accuracy. This comparison supports the selected technique and shows the resilience and efficacy of the developed system. The comparison may uncover areas where existing models underperform, suggesting future directions for improvement or research. In comparison, this research is integrating a hybrid model that combines CNN, EfficientNet, LSTM, and MTCNN. On both image and video datasets, it achieves a superior accuracy of 97.9%. The analysis emphasizes our system's improved performance, generalizability, and decreased complexity, highlighting its role in the development of reliable and scalable DeepFake detection solutions. Table 1 summarizes the key aspects of each study.

The suggested hybrid MTCNN–EfficientNet-B0–LSTM framework performs better than XceptionNet, ResNet50, and CapsuleNet in both image and video classification accuracy when compared directly to cutting-edge DeepFake detection techniques Table 2. The model's advantage for real-time or resource-constrained contexts is highlighted by the fact that it delivers higher accuracy while using much fewer parameters and faster inference.

Table 1. Various DeepFake method comparison

Sl No.	Paper	Model Used	Modality	Accuracy (%)
1.	Alnaim et al. [13]	Multiple CNNs on DFFMD	Image	95.40
2.	Cunha et al. [14]	PSO-improved Deep Neural Network	Video	94.60
3.	Karaköse et al. [15]	Medical DeepFake Detection DL Model	Image	91.30
4.	Preeti et al. [16]	GAN-based Detection	Image + Video	89.80
5.	Theerthagiri and Nagaladinne [19]	Deep InceptionNet	Video	93.00
6.	Suratkar and Kazi [18]	Autoencoder + CNN + LSTM(RNN)	Video	85.84
7.	Mcuba et al. [20]	Chromagram, Spectrogram + VGG-16	Audio	86.90
8.	**Proposed System**	**CNN + EfficientNet + LSTM + MTCNN**	**Image + Video**	**97.9**

Table 2. Model comparison

Model / Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Reason
ResNet50	91.3	90.8	91.0	90.9	Popular CNN, good accuracy, but heavier model
XceptionNet	93.5	93.0	93.3	93.1	High accuracy, effective feature extraction
CapsuleNet	89.7	88.9	89.4	89.1	Captures spatial relationships, moderate performance

5.4 Results

Based on the model's output confidence score, the proposed method uses a threshold-based decision strategy to improve the interpretability and dependability of the classification results. After the deep learning model has processed an input, which could be a face image or a sequence of video frames, a probabilistic score typically ranging from 0 to 1 indicates whether the input is authentic or fraudulent. Inputs are categorized as genuine or fraudulent using a binary threshold-based decision logic. The deep learning model generates a confidence score for each input that indicates the possibility that the input is authentic. If the expected threshold value is higher than 0.5, the input is considered real. On the other hand, if the value is less than or equal to 0.5, the input is deemed fraudulent. This thresholding process ensures a clear and intelligible decision border and enables robust and dependable classification of both images and video frames. After processing an uploaded image or video frame, a trained deep learning pipeline is deployed on a backend server. For temporal analysis and feature extraction, the system uses a hybrid model that combines EfficientNet and LSTM after first detecting and cropping faces using MTCNN.

Figure 5 shows the image-based prediction workflow from input to output in real-time, demonstrating the essential features of the DeepFake Detection web interface. integrating real-time prediction with image analysis and visualizing predictions with confidence scores. With a focus on useful usability and insightful decision feedback, it demonstrates the last phase of the DeepFake detection process.

The proposed system's video-based detection module is shown in Figure 6. Users can upload video files (such as MP4 files) using a file input field on the left panel. Once Submit is clicked, the video is sent to the backend and processed through a multi-phase DeepFake detection pipeline. The Input panel in the center displays a preview of the uploaded video. The system internally extracts frames on a regular basis, utilizes a face detection algorithm like MTCNN after that (every nth frame). The Output panel on the right displays the processed video frame with the detection results superimposed. The green masks on both faces most likely indicate that the model has identified them as Real or authentic based on the classification result in this instance. The system may be using OpenCV or similar libraries to rebuild the annotated video and generate masks or bounding boxes. The user is then presented with the combined annotated frames. From upload and face tracking to classification and output rendering, it shows that the system can manage whole video pipelines.

Figure 7 shows how the proposed detection model processes a single frame taken from a DeepFake video. For precise face detection, the system initially uses MTCNN, making sure that only pertinent facial regions are sent on for additional examination. A CNN-LSTM-based detection pipeline is then fed the identified face. The bounding box draws attention to the recognized face, and the confidence score and the categorization label "Fake" or "Real" are superimposed. This graphic depicts how the system would be used in real-time or almost real-time situations. In dynamic, media-rich settings

where modified content can spread quickly, such as social media platforms, advertising networks, or digital journalism, this kind of functionality is essential.

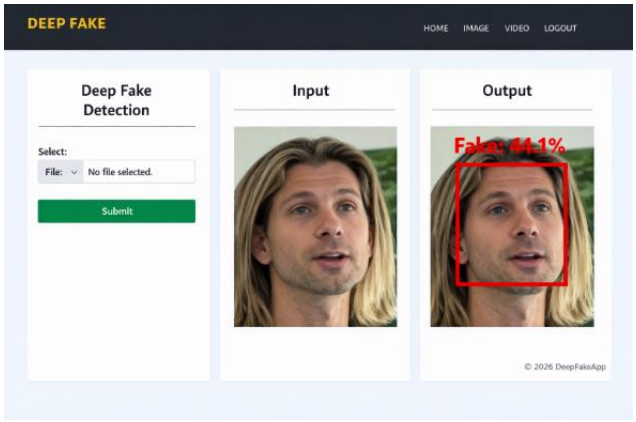

Figure 5. Image-based DeepFake detection interface

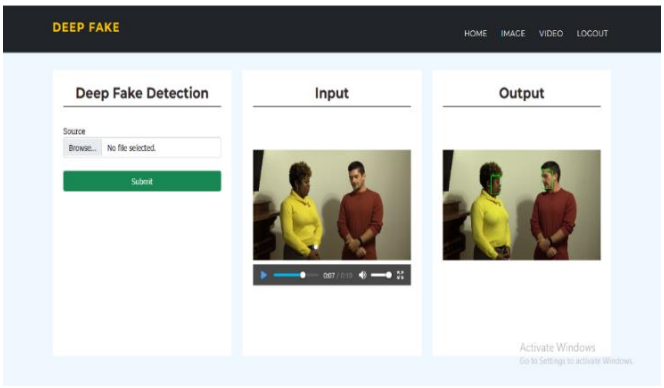

Figure 6. Video classification results

Figure 7. DeepFake detection output from video input

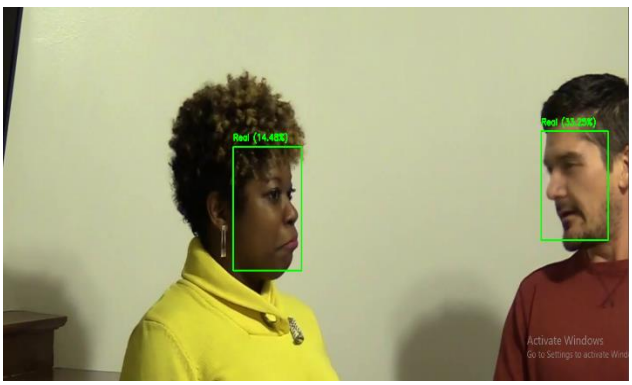

Figure 8. Frame-level face classification result

A video-based detection system's real-time output is shown in Figure 8, which also demonstrates the model's capacity to precisely identify and categorize several faces in a single video frame. Two people are shown in a frame that the system has processed; each is surrounded by a bounding box, indicating successful face detection. As evidence of the model's confidence in recognizing the faces as genuine, the categorization labels "Real" for the individual on the left and "Real" for the one on the right are displayed. Face localization starts the detection pipeline, which then moves on to cropping and preprocessing. After a softmax layer determines the final classification, OpenCV renders the results frame by frame before sending them back to the frontend. The system's ability to handle many facial regions at once and produce high-confidence predictions is seen in this figure, which also showcases the system's end-to-end detection process.

6. DISCUSSION

The suggested DeepFake detection framework, which includes MTCNN for facial localization, EfficientNet-B0 for spatial feature extraction, and LSTM for temporal modeling, has demonstrated remarkable efficacy, achieving an accuracy on benchmark datasets. Due to the combination of temporal and spatial cues, the system is able to detect a wide range of modification artifacts in both images and videos. However, despite its effectiveness, there are still a number of technological limitations and practical application challenges. To further validate the performance reliability of the proposed model, statistical significance tests were performed over five different runs. With an average accuracy of 97.9% and a low standard deviation, the model showed consistent learning. A paired t-test comparing the model with baselines such as XceptionNet and CapsuleNet revealed $p < 0.05$, suggesting that the observed improvements are statistically significant rather than the result of random variability. Difficult visual conditions, such as low light levels, strange head postures or large facial occlusions from objects or accessories, may have a detrimental effect on the model's performance. An incomplete feature extraction could result from MTCNN's inability to provide precise bounding boxes and facial landmarks in such circumstances. Second, CNN-based feature extractors may struggle to identify manipulation artifacts because social media sites frequently employ downsampling noise or excessive video compression. The bias of the dataset is the third limitation. The model's capacity to generalize to unfamiliar domains may be hampered if a training dataset is biased toward particular demographics, lighting conditions, or modification techniques, which could result in integrity issues. Lastly, the LSTM module's contribution to the final classification decision is limited because although it is good at identifying motion irregularities in longer video sequences, it performs poorly in brief clips with few temporal cues. The small dataset size increases the risk of overfitting, even with the model's high accuracy. Training accuracy stabilizes later than validation accuracy in certain training patterns. Although this was mitigated by extensive data augmentation, regularization, and dropout, a more comprehensive cross-dataset evaluation is required to fully validate generalization. Future studies will employ larger-scale datasets and cross-domain validation techniques to further reduce the possibility of overfitting. To boost robustness, the framework might incorporate Swin Transformers or Vision Transformers (ViTs)

for the extraction of spatial characteristics. This would enhance the capacity to describe global connections and long-range interdependence within an image. LSTMs have sequential bottlenecks. 3D CNNs or temporal attention mechanisms could be used instead to better capture spatio-temporal dynamics for temporal modeling. Adaptive thresholding based on prediction confidence may improve classification under uncertain conditions. To reduce dataset bias and improve generalization, domain adaptation and adversarial training strategies could be employed, ensuring resistance to covert manipulation methods and a variety of demographic distributions. Multimodal analysis, which combines visual cues and audio data, can also be used to detect lip-speech synchronization discrepancies, improving detection for videos where both modalities may be altered.

7. CONCLUSIONS

This study offers a reliable and effective DeepFake detection system that reliably makes a distinction between real and DeepFake facial images and videos by utilizing CNNs. While real-time performance and an intuitive web interface facilitate practical implementation. The suggested system offers a theoretically solid, scalable, and interpretable DeepFake image detection method, making a significant contribution to the fields of digital content authentication and multimedia forensics. The proposed method combines the advantages of multiple deep learning architectures, CNN for spatial feature extraction, MTCNN for precise facial region localization, EfficientNet**,** and Long Short-Term networks for temporal sequence Memory modeling in videos. With this hybrid architecture, subtle manipulation artifacts from a variety of DeepFake generating approaches may be robustly detected. With a high detection accuracy of roughly 97.9%, the model keeps computational complexity low while performing noticeably better than conventional single architecture methods. Because the system is modular, it is easy to deploy and integrate into real-time applications, which makes it ideal for processes including digital forensics, social media monitoring, and media verification. This study significantly advances secure and trustworthy multimedia settings by offering a dependable and useful DeepFake detection method.

The DeepFake detection system has certain drawbacks that restrict its use, even though it is reliable and efficient in controlled settings. The system's heavy reliance on the caliber and variety of the dataset is one of the primary problems. The pipelines' preprocessing stages, such as frame extraction, face detection with MTCNN, and deep learning model training with LSTM and EfficientNet, demand a significant amount of processing power. This restricts its use on low-resource devices and may have an impact on the performance of real-time detection. The model's reliance on face-swapping and facial duplication, DeepFakes for training, is another drawback. When exposed to other forms of counterfeit, like full-body DeepFakes audio manipulation or synthetic voice synthesis, it might not function properly. Inaccurate classifications may result from the LSTMs' temporal modeling failing to detect minute variations in short or still video. Lastly, there are issues with generalization across various video formats, compression settings, and new processing tools. The system might not be able to withstand the most recent sneaky DeepFake generation techniques employed by contemporary social media platforms in the absence of regular updates. Moreover, the model's reliance on facial-region data limits its capacity to identify manipulations, including non-facial cues or full-body movements. More lightweight architectures are required since the computational cost of frame extraction, MTCNN-based face localization, and LSTM-based temporal modeling restricts deployment on low-power or mobile devices.

Future studies will expand the system to address further types of manipulations, such as reenactment-based forgeries, full- body DeepFakes, and audio-visual irregularities like lip-sync incompatibilities. Generalization will be strengthened by testing the system on multimodal datasets and investigating transformer-based spatiotemporal topologies. Additionally, the pipeline's practical deployment across security, forensic, and media governance applications will be improved by optimizing it for low-resource devices and real-time streaming contexts.

REFERENCES

[1] Mary, A., Edison, A. (2023). Deep fake detection using deep learning techniques: A literature review. In 2023 International Conference on Control, Communication and Computing (ICCC), Thiruvananthapuram, India, pp. 1-6. https://doi.org/10.1109/ICCC57789.2023.10164881

[2] Wang, J., Wu, Z., Ouyang, W., Han, X., Chen, J., Jiang, Y.G., Li, S.N. (2022). M2TR: Multi-modal multi-scale transformers for deepfake detection. In Proceedings of the 2022 International Conference on Multimedia Retrieval, Newark, NJ, USA, pp. 615-623. https://doi.org/10.1145/3512527.3531415

[3] Heidari, A., Jafari Navimipour, N., Dag, H., Unal, M. (2024). Deepfake detection using deep learning methods: A systematic and comprehensive review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 14(2): e1520. https://doi.org/10.1002/widm.1520

[4] Luo, Y.X., Chen, J.L. (2022). Dual attention network approaches to face forgery video detection. IEEE Access, 10: 110754-110760. https://doi.org/10.1109/ACCESS.2022.3215963

[5] Patel, Y., Tanwar, S., Bhattacharya, P., Gupta, R., Alsuwian, T., Davidson, I.E., Mazibuko, T.F. (2023). An improved dense CNN architecture for deepfake image detection. IEEE Access, 11: 22081-22095. https://doi.org/10.1109/ACCESS.2023.3251417

[6] Boyd, A., Tinsley, P., Bowyer, K.W., Czajka, A. (2023). Cyborg: Blending human saliency into the loss improves deep learning-based synthetic face detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 6108-6117. https://openaccess.thecvf.com/content/WACV2023/papers/Boyd_CYBORG_Blending_Human_Saliency_Into_the_Loss_Improves_Deep_Learning-Based_WACV_2023_paper.pdf.

[7] El-Gayar, M.M., Abouhawwash, M., Askar, S.S., Sweidan, S. (2024). A novel approach for detecting deep fake videos using graph neural network. Journal of Big Data, 11(1): 22. https://doi.org/10.1186/s40537-024-00884-y

[8] Zhang, L., Wang, X., Cooper, E., Evans, N., Yamagishi, J. (2022). The partialspoof database and countermeasures for the detection of short fake speech segments

embedded in an utterance. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 31: 813-825. https://doi.org/10.1109/TASLP.2022.3233236

[9] Waqas, N., Safie, S.I., Kadir, K.A., Khan, S., Khel, M.H.K. (2022). DeepFake image synthesis for data augmentation. IEEE Access, 10: 80847-80857. https://doi.org/10.1109/ACCESS.2022.3193668

[10] Malik, A., Kuribayashi, M., Abdullahi, S.M., Khan, N. (2022). DeepFake detection for human face images and videos: A survey. IEEE Access, 10: 18757-18775. https://doi.org/10.1109/ACCESS.2022.3151186

[11] Rana, M.S., Nobi, M.N., Murali, B., Sung, A.H. (2022). Deepfake detection: A systematic literature review. IEEE Access, 10: 25494-25513. https://doi.org/10.1109/ACCESS.2022.3154404

[12] Patel, Y., Tanwar, S., Gupta, R., Bhattacharya, P., et al. (2023). Deepfake generation and detection: Case study and challenges. IEEE Access, 11: 143296-143323. https://doi.org/10.1109/ACCESS.2023.3342107

[13] Alnaim, N.M., Almutairi, Z.M., Alsuwat, M.S., Alalawi, H.H., Alshobaili, A., Alenezi, F.S. (2023). DFFMD: A deepfake face mask dataset for infectious disease era with deepfake detection algorithms. IEEE Access, 11: 16711-16722. https://doi.org/10.1109/ACCESS.2023.3246661

[14] Cunha, L., Zhang, L., Sowan, B., Lim, C.P., Kong, Y. (2024). Video deepfake detection using Particle Swarm Optimization improved deep neural networks. Neural Computing and Applications, 36(15): 8417-8453. https://doi.org/10.1007/s00521-024-09536-x

[15] Karaköse, M., Yetış, H., Çeçen, M. (2024). A new approach for effective medical deepfake detection in medical images. IEEE Access, 12: 52205-52214. https://doi.org/10.1109/ACCESS.2024.3386644

[16] Preeti, Kumar, M., Sharma, H.K. (2023). A GAN-based model of deepfake detection in social media. Procedia Computer Science, 218: 2153-2162. https://doi.org/10.1016/j.procs.2023.01.191

[17] Abbas, F., Taeihagh, A. (2024). Unmasking deepfakes: A systematic review of deepfake detection and generation techniques using artificial intelligence. Expert Systems with Applications, 252: 124260. https://doi.org/10.1016/j.eswa.2024.124260

[18] Suratkar, S., Kazi, F. (2023). Deep fake video detection using transfer learning approach. Arabian Journal for Science and Engineering, 48(8): 9727-9737. https://doi.org/10.1007/s13369-022-07321-3

[19] Theerthagiri, P., Nagaladinne, G.B. (2023). Deepfake face detection using deep InceptionNet learning algorithm. In 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, pp. 1-6. https://doi.org/10.1109/SCEECS57921.2023.10063128

[20] Mcuba, M., Singh, A., Ikuesan, R.A., Venter, H. (2023). The effect of deep learning methods on deepfake audio detection for digital investigation. Procedia Computer Science, 219: 211-219. https://doi.org/10.1016/j.procs.2023.01.283