# DEEPFAKE DETECTION SYSTEM

Divya K. K.[1,] Muhammad Rafnas[1], K M, Sadeed M T Muhammed [1],Muhammed Minshad[1]

[1]Department of Computer Science and Engineering, P A College of Engineering, Mangalore-574153

*Corresponding Author: Divya K K E-mail: divya_cs@pace.edu.in

## Abstract

This paper explores a deep learning system to detect deepfake videos, a common type of fake media. By using advanced techniques like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), our system accurately distinguishes real videos from manipulated ones. It analyzes both the images and the audio in videos to find signs of deepfake manipulation. We process video frames and audio, extract features with CNNs and RNNs, and combine these features to decide if a video is real or fake. We trained and tested our system on large datasets of both real and fake videos to ensure its reliability. Proposed system helps fight misinformation and protect the authenticity of digital content.

**Keywords:** Convolutional Neural Network, Deepfake, Recurrent Neural Network.

## 1 Introduction

The rapid advancement of artificial intelligence (AI) and machine learning has given rise to deepfake technology, which can create highly realistic fake media by manipulating video and audio content. These AI-generated forgeries can convincingly alter appearances, voices, and actions, making it increasingly difficult to distinguish between authentic and manipulated media. While deepfake technology has promising applications in entertainment, education, and the creative arts, its misuse poses significant threats to privacy, public trust, political stability, and the integrity of information. Deepfakes can be weaponized for malicious purposes, including spreading misinformation, perpetrating fraud, and discrediting individuals or organizations. For instance, deepfakes can be used to create fake news videos that mislead the public or fraudulent videos that harm reputations. This potential for harm underscores the need for robust detection mechanisms to identify and mitigate the effects of deepfake content. Traditional detection methods, which often rely on manual analysis or simple heuristics, have proven inadequate in the face of sophisticated deepfake generation techniques. This work aims to address this critical challenge by developing a deepfake detection system based on deep learning techniques. By leveraging the power of deep learning, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), our system seeks to accurately distinguish between real and manipulated media. The proposed system will analyze both spatial and temporal features of videos to detect subtle inconsistencies and artifacts indicative of deepfake manipulation. Through rigorous training and evaluation on large datasets of real and synthetic videos, the system aims to achieve high reliability and generalization across various deepfake generation techniques. Our ultimate goal is to contribute to the fight against misinformation and help preserve the integrity of digital content in the age of synthetic media.

## 2 Literature Survey

Michael Baker et al. [1] Eagle Eye: Detecting Deepfakes through Gaze Analysis and Facial Landmark Tracking, introduces an innovative method for identifying deepfake media. The study presents a system that uses eye gaze pattern analysis and facial landmark tracking to detect synthetic media where one person's likeness is replaced with another's using deep learning techniques. Dr. Emily Wilson, Prof. Benjamin Lee, and Dr. Rachel Adams [2] Forensic Net: Deepfake Attribution through Source Identification, introduce a groundbreaking method to tackle the growing issue of deepfake proliferation. In the fast-changing world of digital media, deepfakes—synthetic media created by AI algorithms—have become powerful tools for spreading misinformation and manipulating public perception. To address this threat, the authors propose Forensic Net, an innovative system that uses advanced forensic techniques to trace deepfake content back to its source. Megan Chen, David Wilson, and Dr. Sofia Rodriguez,[3] "LipSyncGuard: Deepfake Detection Using Lip-Sync Analysis and Audio-Visual Cues," present a new method for detecting deepfake videos. With the rise of AI- generated fake media, it's crucial to have effective detection techniques. The authors introduce LipSyncGuard, a system that uses lip-sync analysis and audio-visual cues to identify deepfakes. By checking if lip movements match the audio, LipSyncGuard can find signs of manipulated content. Recurrent Neural Network (RNN) for deepfake detection,[4] the authors propose a method that uses Recurrent Neural Networks (RNNs) alongside an ImageNet pre-trained model to process frames sequentially. However, it's worth noting that they employed a small dataset known as the HOHO dataset, comprising only 600 videos. Synthetic Portrait Videos using Biological Signals" [5] by researchers in computer vision, signal processing, and machine learning introduces a new way to detect deepfake videos using biological signals from faces. This paper is a major step forward in deepfake detection. The authors analyze real and fake videos by extracting biological signals from faces, then converting these signals into feature vectors and PPG maps to identify unique patterns. Yuezun Li et al [6] analysis showing AI-synthesized face-swapping videos, commonly known as DeepFakes, is an emerging problem threatening the trustworthiness of online information. The need to develop and evaluate DeepFake detection algorithms calls for large-scale datasets. The current approaches to deepfake detection exhibit a wide range of complexity and effectiveness, spanning from manual inspection techniques to advanced machine learning models. Here are the primary categories of existing deepfake detection systems and their common methodologies:

### 3.1 Manual Inspection and Heuristics:

These traditional techniques depend on human observation to detect inconsistencies in visual and auditory aspects, including unnatural facial movements, irregular blinking, or lip-syncing errors. Examining the metadata of video files can uncover anomalies like atypical file creation dates or signs of specific editing software.

### 3.2 Machine Learning Approaches:

These approaches involve extracting distinct features from videos or images, such as color histograms, edge patterns, and texture characteristics. Traditional classifiers, including Support Vector Machines (SVMs) or Random Forests, are then employed to distinguish between authentic and fake media. Methods like Error Level Analysis (ELA) and frequency domain analysis are used to identify irregularities introduced during the creation of deepfakes.

### 3.3 Hybrid And Ensemble Methods

These hybrid models merge Convolutional Neural Networks (CNNs) for spatial analysis with Recurrent Neural Networks (RNNs) for temporal analysis, offering stronger detection capabilities. They effectively identify both frame-specific artifacts and inconsistencies across sequences of frames.

## 4 Proposed System

The proposed system harnesses advanced deep learning techniques to detect deepfake videos with exceptional accuracy and robustness. By combining spatial and temporal analysis, the system can identify the subtle inconsistencies and artifacts that are characteristic of deepfake media. Specifically, it employs convolutional neural networks (CNNs) for detailed spatial analysis and recurrent neural networks (RNNs) for comprehensive temporal analysis. This integration enables the system to thoroughly examine both individual frames and sequences of frames, significantly enhancing its detection capabilities. By leveraging CNNs for spatial features and RNNs for temporal patterns, the system adeptly captures both frame-level and sequence-level inconsistencies. This dual approach ensures that the model can detect subtle artifacts and anomalies that might be missed by single- method approaches. Furthermore, the system is optimized for real-time processing, making it highly effective for the timely detection of deepfake content in dynamic environments such as live video streams and social media platforms. This optimization ensures that the system can respond quickly and accurately, providing a robust solution for real-time deepfake detection challenges.

### 4.1 Deepfake Videos

Creating deepfake videos involves the use of tools such as GANs and autoencoders. These tools process a source image and a target video by splitting the video into individual frames, detecting faces, and replacing the source face with the target face in each frame. Pre- trained models are then employed to reassemble these frames, improving the video quality and removing any residual traces.

Although deepfakes appear highly realistic, they often leave behind subtle artifacts that are imperceptible to the naked eye. This paper focuses on identifying and classifying these faint traces to differentiate between deepfake and authentic videos.

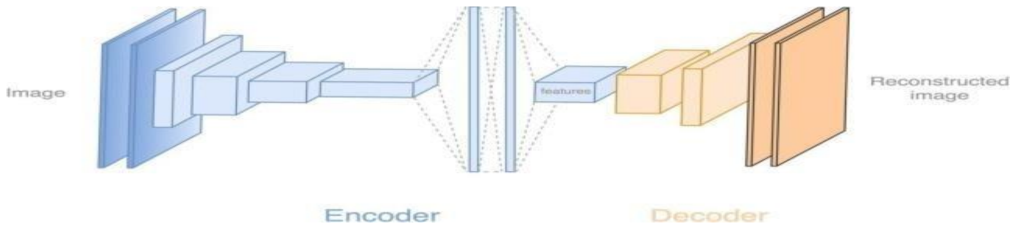


Figure 1: Deepfake generation

### 4.2 Dataset Gathering

To ensure the model's efficiency for real-time prediction, data was collected from various sources, including FaceForensic++ (FF), the Deepfake Detection Challenge (DFDC), and Celeb-DF. These datasets were combined to form a new, comprehensive dataset, enhancing the accuracy and real-time detection capabilities across different video types. To prevent training bias, the dataset was balanced with 50% real and 50% fake videos. Certain audio-altered videos from the

ISBN-978-81-19905-39-3

DFDC dataset were excluded, as they were irrelevant to the paper's scope. A preprocessing step using a Python script was employed to remove these videos. After preprocessing, 1500 real and 1500 fake videos were selected from the DFDC dataset. Additionally, 1000 real and 1000 fake videos were chosen from the FaceForensic++ dataset, and500 real and 500 fake videos were selected from the Celeb-DFdataset. This resulted in a balanced dataset comprising 3000 real and 3000 fake videos, totaling 6000 videos.

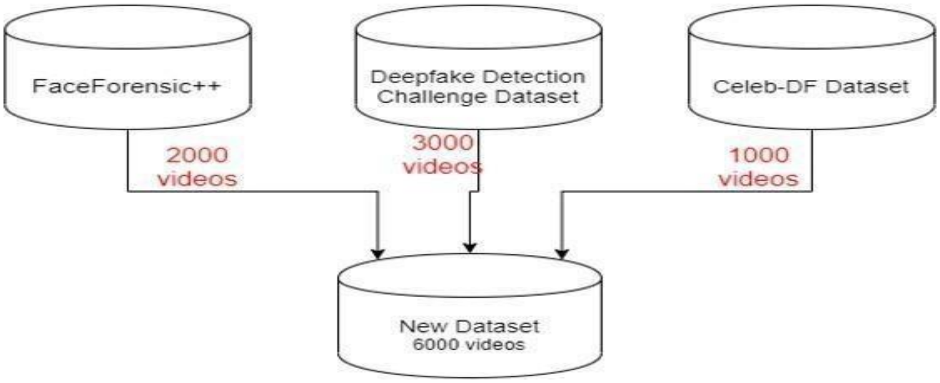


Figure 2: Dataset

## 4.2 Pre-Processing

In this module, video preprocessing involves removing noise and irrelevant content, focusing solely on the necessary facial features. The process starts by splitting each video into individual frames. Face detection is performed on each frame, and the region containing the face is cropped. These cropped frames are then recombined to reconstruct the video, now featuring only the face. To maintain consistency in the number of frames across videos and accommodate computational limitations, a threshold value is determined based on the average total frame count of each video. Given the computational constraints of the experimental environment, a threshold of 150 frames is chosen. This threshold ensures manageable processing while preserving essential information. Thus, only the first 150 frames of each video are retained for the new dataset creation, facilitating sequential processing.

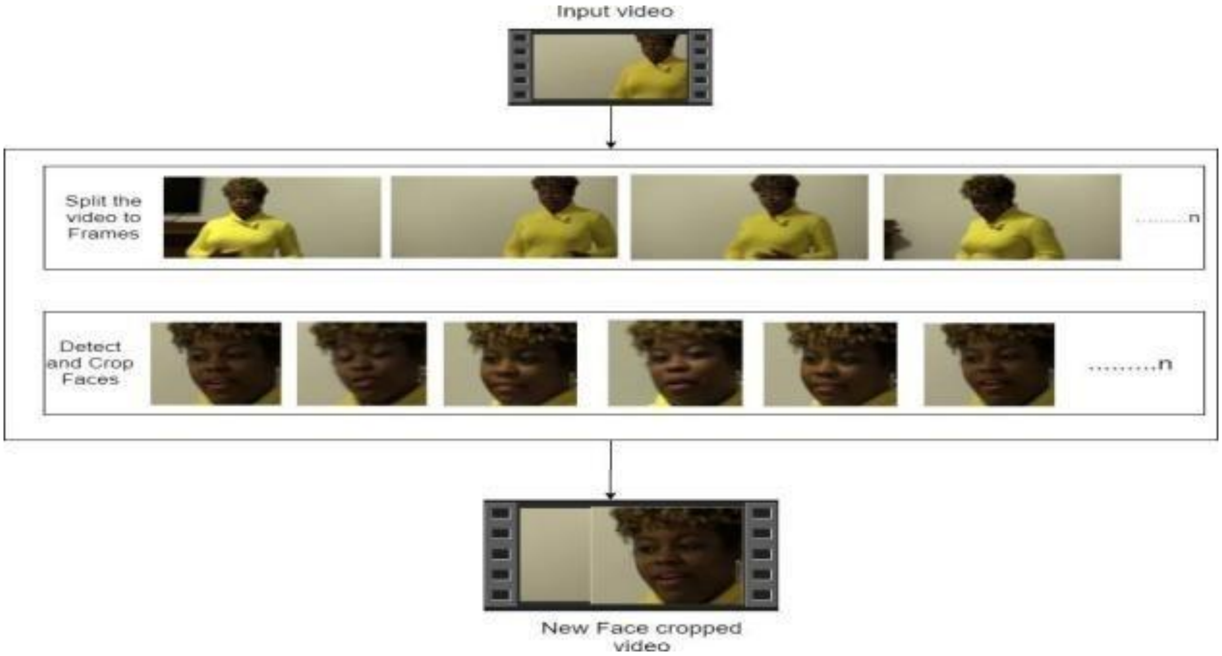


Figure 3: Pre-processing of video

To demonstrate the efficacy of Long Short-Term Memory (LSTM) networks, frames are arranged sequentially rather than randomly. The resulting videos are saved at a frame rate of 30 frames per second (fps) and a resolution of 112 x 112, ensuring compatibility and standardization across the dataset.

### 4.3 Model Architecture

Our model leverages a combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). Specifically, we use a pre-trained ResNext CNN model to extract frame-level features. These features are then fed into a Long Short-Term Memory (LSTM) network, which is trained to classify videos as either deepfake or pristine. To facilitate training, we use a Data Loader to load the video labels from the training dataset. These labels are integrated into the model during the training process, enabling the model to learn from the provided data and optimize its parameters for effectively distinguishing between deepfake and pristine videos.

### 4.4 Resnext

We utilized the pre-trained ResNext model for feature extraction, bypassing the need to write code from scratch. ResNext is a variant of the Residual Convolutional Neural Network (CNN) architecture, optimized for high performance in deeper neural networks. Specifically, for our experiments, we employed the resnext50_32x4d model, which has 50 layers and 32x4 dimensions. After extracting features with the ResNext model, we fine-tuned the network by adding additional layers and selecting an appropriate learning rate to ensure proper convergence of the gradient descent process. The 2048-dimensional feature vectors obtained after the last pooling layer of ResNext serve as the input for the sequential Long Short- Term Memory (LSTM) network. This approach allows us to effectively utilize the extracted features for video classification tasks.

### 4.5 LSTM for Sequence Processing

The model architecture begins with a 2048-dimensional feature vector input to a single LSTM layer, which includes 2048 latent dimensions and 2048 hidden units, along with a dropout probability of 0.4. This configuration enables effective sequential frame processing for temporal video analysis. A Leaky ReLU activation function is applied, followed by a linear layer to learn input-output correlations. An adaptive average pooling layer ensures consistent image dimensions. Sequential processing is managed with a batch size of 4 to enhance training efficiency. Finally, a softmax layer provides confidence scores for the predictions

### 4.6 Hyper Parameter Tuning

The process involves selecting optimal hyperparameters to maximize accuracy. After several iterations, the best hyperparameters for our dataset are determined. We use the Adam optimizer with a learning rate set to 1e-5 (0.00001) to ensure convergence to a better global minimum, along with a weight decay of 1e-3. computational resources. The user interface is developed using the Django framework, chosen for its scalability. The index.html page includes a tab for video upload. Uploaded videos are processed by the model for prediction, which outputs whether the video is real or fake, along with confidence by the model for prediction, which outputs whether the video is real or fake, along with confidence scores.

ISBN-978-81-19905-39-3

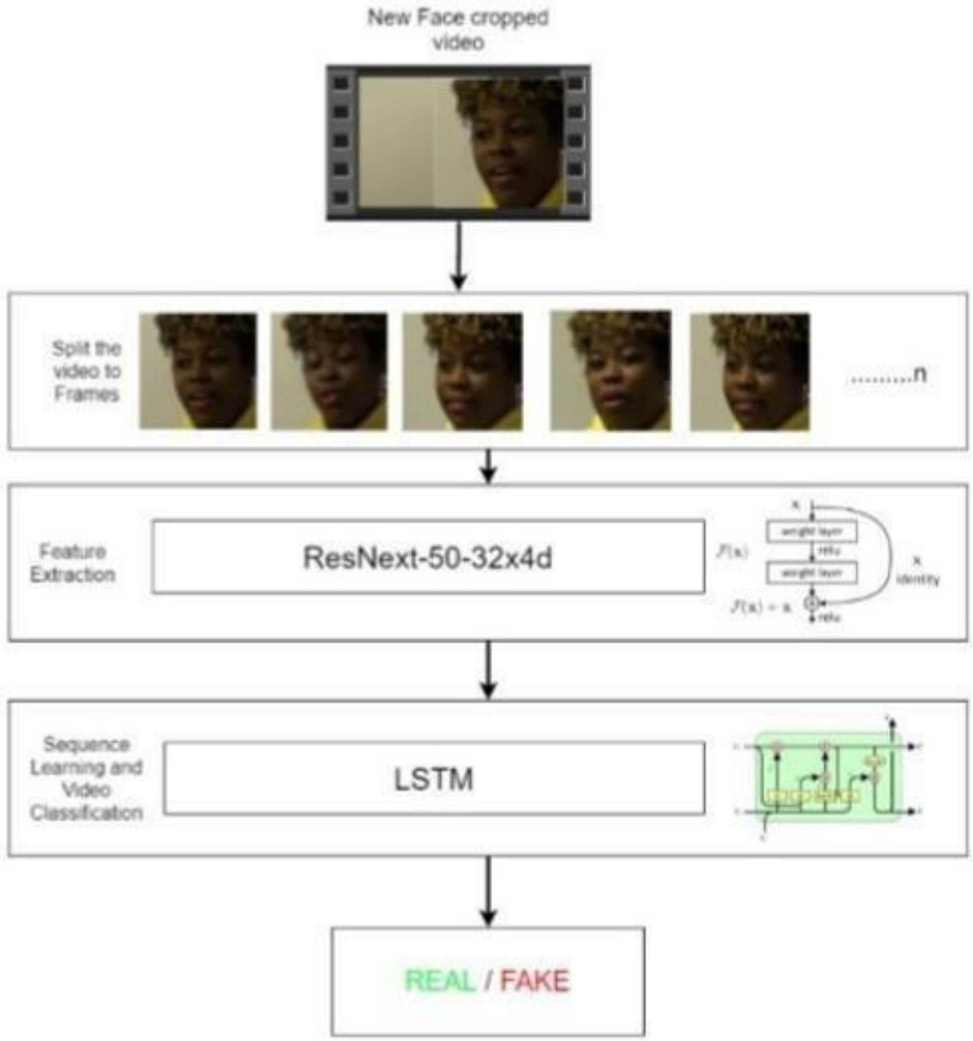


Figure 4: Overview of model

## 5 Testing Strategy

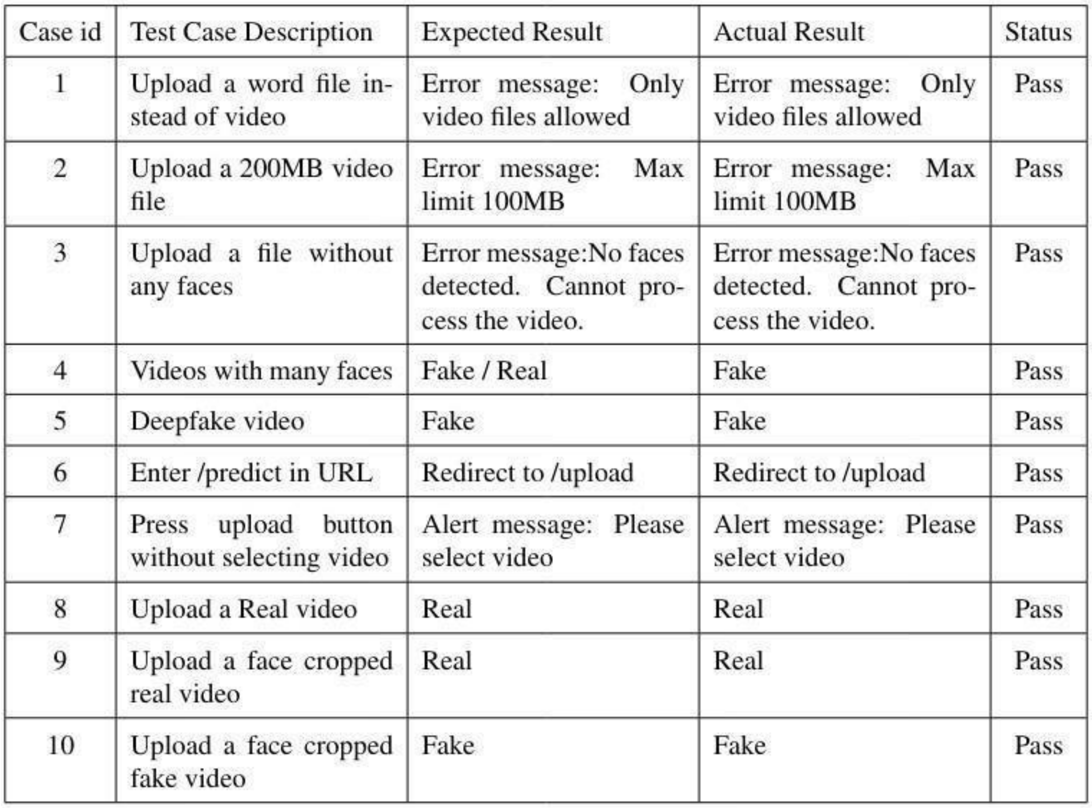
Proposed approach to unit testing is manual, typically conducted through debugging sessions within an Integrated Development Environment (IDE). This involves setting breakpoints and performing step-through debugging. We provide the unit with a variety of inputs, both valid and

ISBN-978-81-19905-39-3

invalid, to observe its responses. The main advantage of this manual approach is its high visibility to the current developer. Since the code is fresh in their minds, any identified bugs can usually be fixed quickly. Each test case outlines several features describing the input-output transformations. The features to be tested focus primarily on the accuracy of the individual unit and the range of inputs for which the unit operates correctly. Items tested include all individual units or functions that collectively make up the entire system. In unit testing, the primary focus is on these individual units. Each test case includes sample inputs, which can be any valid input for the unit, along with the expected output and the actual output. The testing strategy also includes remarks on the performance of the unit in each test case. We emphasize testing a wide variety of scenarios to ensure comprehensive coverage. This includes edge cases, typical use cases, and erroneous inputs to verify how the unit handles unexpected situations. Documentation of each test case is thorough, including descriptions of the test scenario, the inputs used, expected results, actual results, and any discrepancies observed. This documentation helps in maintaining a clear understanding of the unit’s behavior over time.

## 6 Conclusion and Future Scope

The future scope of our deepfake detection project is vast and promising, especially in light of the rapid advancements in deepfake technologies and their growing prevalence. One significant avenue for expansion involves developing browser plugins and integrating them with social media platforms, enabling users to access real-time deepfake detection seamlessly during their everyday digital interactions. Additionally, there’s potential for enhancing the algorithm to detect full-body deepfakes and audio manipulations, thereby creating a more comprehensive detection system capable of identifying a broader range of digital forgeries. In conclusion, the application of deep learning for deepfake detection marks a significant advancement in the realm of digital media forensics. Utilizing sophisticated neural networks, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), has proven highly effective in discerning authentic content from manipulated material with exceptional accuracy.



| Case id | Test Case Description | Expected Result | Actual Result | Status |
|---|---|---|---|---|
| 1 | Upload a word file instead of video | Error message: Only video files allowed | Error message: Only video files allowed | Pass |
| 2 | Upload a 200MB video file | Error message: Max limit 100MB | Error message: Max limit 100MB | Pass |
| 3 | Upload a file without any faces | Error message:No faces detected. Cannot process the video. | Error message:No faces detected. Cannot process the video. | Pass |
| 4 | Videos with many faces | Fake / Real | Fake | Pass |
| 5 | Deepfake video | Fake | Fake | Pass |
| 6 | Enter /predict in URL | Redirect to /upload | Redirect to /upload | Pass |
| 7 | Press upload button without selecting video | Alert message: Please select video | Alert message: Please select video | Pass |
| 8 | Upload a Real video | Real | Real | Pass |
| 9 | Upload a face cropped real video | Real | Real | Pass |
| 10 | Upload a face cropped fake video | Fake | Fake | Pass |

Figure 5: Test case report

Techniques such as transfer learning, data augmentation, and the fusion of temporal and spatial features have further bolstered the resilience of these models. As adversaries develop increasingly sophisticated algorithms to produce highly convincing deepfakes, ongoing updates and enhancements to detection models are imperative. This necessitates a dynamic approach that incorporates new data and adapts to emerging manipulation techniques. Furthermore, the deployment of deepfake detection systems must be underpinned by ethical considerations, addressing concerns related to privacy, consent, and the potential for misuse. Ensuring transparency in the development and deployment of these systems is vital to maintain public trust and uphold ethical standards.

## References:

[1] Rossler, Andreas, et al. "Faceforensics++: Learning to detect manipulated facial images." Proceedings of the IEEE/CVF international conference on computer vision. 2019.

[2]Deepfake detection challenge dataset 2020,

[3]https://fortune.com/2019/06/12/deepfake-mark-zuckerberg/Accessedon26.

[4] Lyu, Yingchun, et al. "Correlations between transition-metal chemistry, local structure, and global structure in Li2Ru0. 5Mn0. 5O3 investigated in a wide voltage window." Chemistry of Materials 29.21 (2017): 9053-9065.

[5] Li, Yuezun, and Siwei Lyu. "Obstructing deepfakes by disrupting face detection and facial landmarks extraction." Deep Learning-Based Face Analytics (2021): 247-267.

[6] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi and Siwei Lyu “Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics” in arXiv: 1909.12962