iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** VI **Month of publication:** June 2024

DOI: https://doi.org/10.22214/ijraset.2024.62936

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

Multimodal Deepfake Detection

Prof. Sneha G[1], Prof. Divya S[2], Lavanya R[3], Leesha V Kumar[4], Navyatha A[5], Nishu Agarwal[6]
Department of Computer Science and Engineering, Dr. Ambedkar Institute of Technology

Abstract: ***Deepfake technology has become a significant threat to the integrity of multimedia content, posing challenges to areas such as cybersecurity, media forensics, and information authenticity. To address this, our research introduces a Multimodal Deepfake Detection system capable of identifying manipulated content by combining visual and auditory cues. The model employs convolutional neural networks (CNNs) to analyse video frames and process audio spectrograms, providing a comprehensive approach to detecting deepfake content. Experimental results demonstrate the system's effectiveness in accurately identifying both visual and auditory indicators of deepfake manipulation. This solution shows promise in combating the spread of deepfake content on digital platforms, thereby preserving the integrity and trustworthiness of multimedia content in the digital era.***

Keywords: Deepfake Detection, Multimodal Analysis, Convolutional Neural Networks, Deep Neural Networks, Multimedia Forensics

I. INTRODUCTION

In recent years, the proliferation of deepfake technology has presented significant challenges to various fields, including media, politics, and cybersecurity. Deepfakes, which are highly realistic synthetic media created using advanced artificial intelligence techniques primarily deep learning algorithms have the potential to manipulate perceptions, spread misinformation, and deceive individuals on a large scale. As deepfake technology becomes increasingly sophisticated, the urgency for robust detection methods intensifies.

Traditional deepfake detection approaches typically focus on analysing either visual or audio cues in isolation. However, these unimodal techniques often struggle to effectively detect advanced deepfake content that incorporates multiple modalities. To overcome this limitation, researchers have shifted to multimodal analysis, leveraging information from both visual and audio sources to enhance detection accuracy.

This research paper presents a comprehensive study on developing a multimodal deepfake detection system. The proposed system integrates convolutional neural networks (CNNs) for both visual and audio analysis, enabling a holistic assessment of multimedia content. By combining information from different modalities, the system aims to achieve superior performance in detecting deepfake videos and audio clips.

II. RELATED WORK

In this section, we provide an overview of existing approaches in the field of deepfake detection. We discuss various methods proposed by researchers to detect and mitigate the spread of synthetic media. These approaches encompass both unimodal and multimodal techniques, each with its advantages and limitations.

A. Unimodal Approaches

Unimodal approaches focus on analysing either visual or audio cues to identify signs of manipulation in media content. Visual-based methods typically involve the use of convolutional neural networks (CNNs) to analyse facial features, inconsistencies in lip movements, and artifacts introduced during the generation process. Similarly, audio-based techniques utilize deep learning models, such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, to detect anomalies in speech patterns, spectral characteristics, and other audio attributes. While unimodal approaches have achieved moderate success in detecting basic deepfake content, they often struggle to discern highly realistic manipulations that mimic genuine human behaviour. Moreover, these methods are susceptible to adversarial attacks and may fail when faced with subtle alterations in media content.

B. Multimodal Approaches

Multimodal approaches integrate information from multiple modalities, such as visual and audio signals, to enhance detection accuracy and robustness. By combining complementary features from different sources, multimodal systems can better distinguish between genuine and synthetic media.

These approaches typically employ deep learning architectures that jointly process visual and audio inputs, allowing for a more comprehensive analysis of multimedia content.Recent research has shown promising results with multimodal deepfake detection systems. These systems leverage the synergistic benefits of combining visual and audio cues, enabling them to detect sophisticated deepfake content with higher accuracy. However, challenges remain in optimizing the fusion of multimodal features and addressing scalability issues in real-world applications.

C. Limitations and Challenges

Despite advancements in deepfake detection technology, several challenges persist. One major limitation is the rapid evolution of deepfake generation techniques, which constantly challenge the effectiveness of existing detection methods. Additionally, the scarcity of labelled training data for multimodal analysis poses a significant obstacle to training robust detection models.

Furthermore, the computational complexity of multimodal approaches and the need for large-scale computing resources hinder their widespread adoption. Moreover, ethical considerations surrounding the development and deployment of deepfake detection systems raise concerns regarding privacy, consent, and potential misuse.

III. LITERATURE SURVEY

David, et.al has proposed a model [1] that incorporates two-stage analysis which includes CNN and LSTM to automatically detect deepfake videos. This system uses CNN to frame-level extraction of feature and LSTM for temporal sequence analysis. The Dataset of 300 videos is used for experimentation of the proposed model known as HOHA dataset.

Kurniawan, et.al proposes study[2] about many algorithms built to detect deepfake content in images and videos. They include several approaches like visual feature based approach, local feature based approach, deep feature based approach and temporal feature based approach.

Aya, et.al presents [3] you only look once– convolutional neural network–extreme gradient boosting (YOLO CNN XGBoost). The YOLO face detector extracts the face area from video frames, while the InceptionResNetV2 CNN extract features from these faces. These features are fed into the XGBoost that works as a recognizer on the top level of the CNN network. The proposed method achieves 90.73% accuracy on the CelebDF-FaceForencics++ (c23) merged dataset.

Duha, et.al., proposes model [4] to detect deepfakes is based on a hybrid approach for feature extraction by using 128-identity features obtained from facenet_ CNN combined with most powerful 10 PCA features. All these features are extracted from cropped faces of 10 frames for each video. FaceForensics++ (FF++) dataset is used to train and test the model, which gives a maximum test accuracy of 0.83, precision of 0.824 and recall value of 0.849.

Neha, et.al., presents study [5] that focuses on presenting generative and detection techniques for visual deep fake media using various technologies, including deep learning (CNN, RNN, LSTM), machine learning (SVM, KNN, Random Forest, Decision Tree), and statistical learning (3D Morphable Model). The research conducts a comprehensive analysis of existing literature on deepfake technology, exploring open-source tools for generating manipulated media. The study extensively reviews face manipulation methodologies, specifically addressing Identity swap, Image Synthesis, Face Re-enactment, and Attribute Manipulation. A novel taxonomy is proposed based on spatial, temporal, and frequency-based features for the detection of visual deepfakes, surpassing existing surveys in terms of domain coverage, learning methods, features, and manipulation techniques. The study also highlights challenges and research gaps, offering an analysis of each to prioritize the development of deep fake detection tools. Overall, the research aims to contribute to the understanding of deepfake technology, its detection methods, and the ongoing challenges in this rapidly evolving domain.

IV. METHODOLOGY

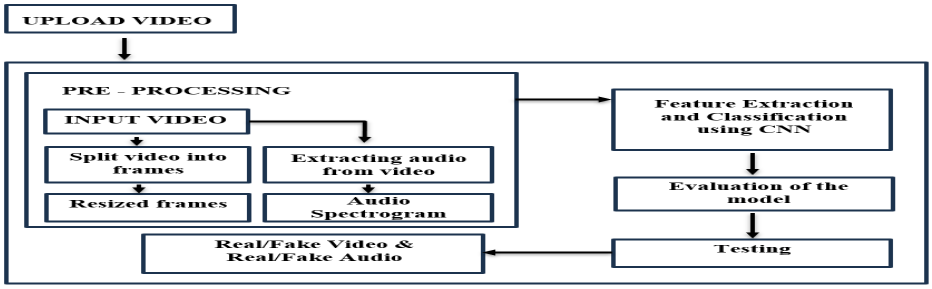


FIG I. Methodology

A. Data Collection

The FakeAVCeleb dataset is a novel Audio-Video Multimodal Deepfake Detection dataset that contains both video and audio deepfakes with accurate lip-sync and fine-grained labels. It was generated using real YouTubevideos of 6,112 celebrities from the VoxCeleb2 dataset, with an emphasis on gender, ethnicity, and age diversity. For the creation of manipulated video content, state-of-the-art techniques were employed including Faceswap and FSGAN. To synchronize the generated lip movements with the target speech, Wav2Lip was utilized and Real-time voice cloning (SV2TTS) was implemented to clone the audio. Total 434 videos has been used for training and testing.

TABLE I
DATA USED FOR TRAINING

Sl. No.	Class	Number of Videos
1	Fake Video Real Audio	192
2	Fake Video Fake Audio	222
3	Real Video Real Audio	10
4	Real Video Fake Audio	10

FIG II. Table describing the data collection

B. Pre-processing

1) *Video Extraction from Videos*: The code extracts audio from video files using MoviePy library. It iterates through each folder containing video files, extracts audio segments, and saves them as WAV files. Audio Evaluation of the model Testing extraction is performed using the extract_audio function, and folders are processed using the process_folder function.
2) *Spectrogram Generation from Audio:* This segment processes audio files by calculating the Short Time Fourier Transform (STFT) and generating spectrograms. It breaks the audio signal into short overlapping segments using a window function and applies logarithmic scaling to the spectrogram to emphasize weaker frequency components. The spectrogram is visualized using Matplotlib, with adjustments for figure size and resolution to ensure a detailed view, and includes features like colorbars and axis labels. The code automates the processing of multiple audio files, iterating through a directory to compute and save spectrograms for each file in a specified output folder. This setup allows for efficient handling and visualization of large batches of audio files, providing clear insights into their frequency content over time. Fig III is the output of spectogram after Short-time Fourier Transformation. Fig IV is the resultant Transformation.

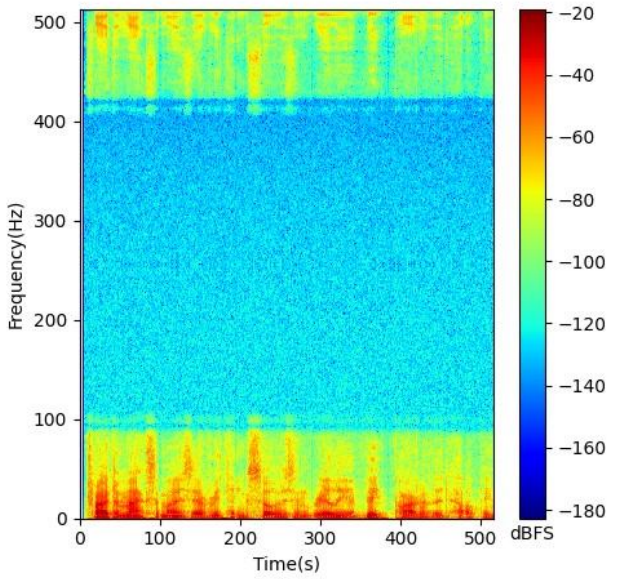


FIG III. Spectogram after Short-Time Fourier Transformation

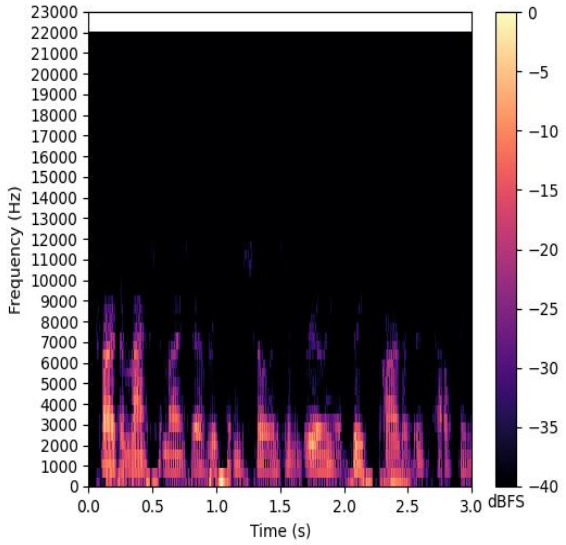


FIG IV. Spectogram after Logarithmic Transformation

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

C. Training

The training process involves training two convolutional neural network (CNN) models, one for image data (trained_model_CNN.h5) and another for spectrogram data (trained_model_DNN.h5). Here's an overview of the training process for each model:

1) Video: CNN Model

The model architecture consists of convolutional layers, max-pooling layers, dense layers, activation functions (ReLU), and dropout layers. The model is compiled with categorical cross-entropy loss and RMSprop optimizer. The training data consists of images loaded from the ./img/ directory. The training process is supervised, with class labels specified by the directory structure. During training, the model is validated on a 30% split of the training data. The training is conducted for 100 epochs with a batch size of 10. Training progress is visualized using accuracy and loss graphs.

2) Audio: DNN Model

The model architecture is similar to the CNN model but adapted for the spectrogram data. Spectrogram data is loaded from the ./audio_output/ directory. The training process follows a similar procedure as the CNN model, including compilation, validation, and evaluation. The same evaluation metrics are calculated and printed for this model as well.

D. Testing

The code in the testing process demonstrates a system that utilizes pre-trained model which includes CNN to analyze both the visual and audio aspects of a video to classify it as real or fake. It is a combination of audio processing, deepfake detection, and a Tkinter-based GUI for interacting with these functionalities.

1) Video Classification

Capture Frame: When the "Capture Frame" button is clicked, the user is prompted to select a video file from their system. The code then captures the 20th frame from the selected video and saves it as a temporary image file (temp.jpg).
Load and Display Captured Frame: The captured frame is loaded and displayed in the GUI window. This frame represents a snapshot of the video content chosen by the user.
Load Pre-Trained CNN Model: The code loads a pre-trained convolutional neural network (CNN) model (trained_model_CNN.h5) that has been trained to classify videos as either real or fake.
Preprocess the Captured Frame: The captured frame image is pre-processed to prepare it for input to the CNN model. This preprocessing involves resizing the image to the required dimensions and normalizing its pixel values.
Make Predictions Using CNN Model: The pre-processed frame image is passed through the CNN model, which predicts whether the video represented by the frame is real or fake.
Display Predicted Video Classification: The predicted classification result (real or fake) is displayed in the GUI window.

2) Audio Classification

Extract Audio from Video: The code extracts the audio from the selected video file and saves it as a separate audio file (audio.wav).
Plot Spectrogram of Audio: The code generates a spectrogram of the audio file using the Short-Time Fourier Transform (STFT) technique. The spectrogram represents the frequency content of the audio signal over time.
Load Pre-Trained CNN Model: The code loads a pre-trained CNN model (trained_model_DNN.h5) that has been trained to classify audio spectrograms as either real or fake.
Preprocess the Spectrogram Image: The generated spectrogram image is pre-processed to prepare it for input to the model. This preprocessing involves resizing the image to the required dimensions and normalizing its pixel values.
Make Predictions Using CNN Model: The pre-processed spectrogram image is passed through the trained CNN model, which predicts whether the audio represented by the spectrogram is real or fake.
Display Predicted Audio Classification: The predicted classification result (real or fake) is displayed along with the video classification result in the GUI window.
The model has been trained with a dataset split of 50:50 (i.e., using 50% of the data for training and 50% for testing).

V. RESULT

The training has been carried out using the selected videos from each class of dataset. The accuracy obtained through proposed model is 85%. Testing results in detection of the reality of video and audio, ensuring both are fake or real.

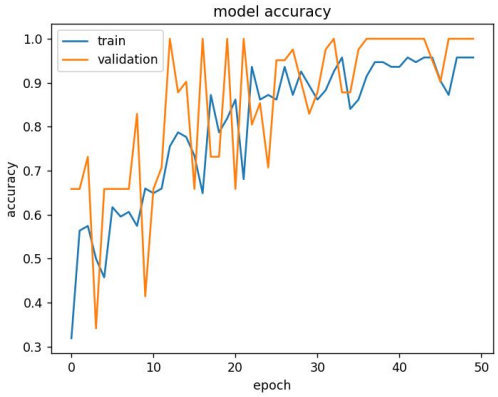


FIG V. Model accuracy

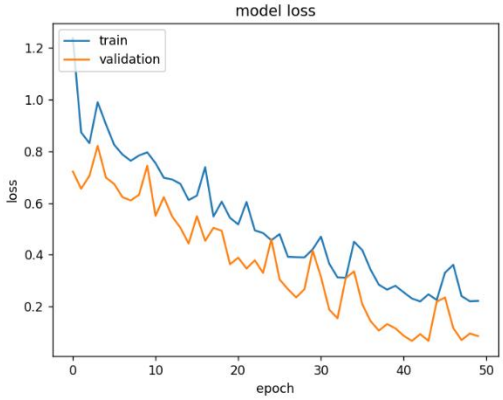


FIG VI. Model loss

The graph (fig V) shows how well a model is learning (training accuracy) and how well it generalizes to new data (validation accuracy) over training iterations (epochs). Ideally, both should increase, but validation accuracy should not shoot up too fast, which might indicate overfitting.

The graph shows (fig VI) how much error (loss) the model has during training (train) and on unseen data (validation) as it learns (epochs). Ideally, both errors go down, but the unseen data should not improve too quickly, as that might signal overfitting.

VI. CONCLUSIONS

This research demonstrates a significant advancement in the field of deepfake detection through the development of a Multimodal Deepfake Detection System. By integrating Convolutional Neural Networks (CNNs) for visual analysis and audio analysis, it effectively leverages both visual and auditory cues to identify manipulated content. By analysing both video frames and audio spectrograms, the system can identify inconsistencies and artifacts that are indicative of deepfake content.

In conclusion, the Multimodal Deepfake Detection System represents a promising step forward in combating the proliferation of deepfake content. By combining visual and auditory analysis, the system offers a more comprehensive and effective solution to preserving the authenticity and trustworthiness of multimedia content in the digital age. Future research should continue to build on these findings, addressing the outlined challenges to further enhance detection capabilities and ethical implementation.

VII. FUTURE ENHANCEMENT

1) Incorporation of Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs):

Objective: Improve temporal analysis in video data.

Description: While the current system leverages CNNs for frame-based video analysis, incorporating RNNs or LSTMs could enhance the system's ability to understand temporal dependencies and motion patterns in video sequences, leading to more accurate deepfake detection.

2) Integration of Advanced Preprocessing Techniques:

Objective: Enhance data quality for better model performance.

Description: Implement more sophisticated preprocessing techniques such as advanced noise reduction for audio data and state-of-the-art image enhancement methods for video frames. This would help in extracting more robust features from the data.

3) Multi-Feature Fusion:

Objective: Utilize a more comprehensive set of features for detection.

Description: Develop methods to fuse features from different modalities (e.g., facial landmarks, voice tonality, and lip synchronization) to create a more holistic and robust deepfake detection system. This can be achieved using attention mechanisms or graph neural networks.

4) Real-Time Detection Capability:

Objective: Detect deepfakes in real-time.

Description: Optimize the model and system architecture to allow for real-time detection of deepfakes. This involves reducing the computational complexity and ensuring the system can process and analyze data swiftly without significant latency.

5) Enhanced Dataset and Training:

Objective: Improve model generalization and robustness. Description: Collect and utilize a more diverse and larger dataset, encompassing various deepfake generation techniques. Additionally, employing data augmentation strategies and semi-supervised learning could enhance the model's ability to generalize to unseen data.

6) User-Friendly Interface:

Objective: Make the system accessible to non-experts.

Description: Develop a more intuitive and user friendly graphical interface, including better visualization of detection results and providing users with detailed explanations of the detection process. This could help users understand the reasons behind the classification.

7) Cross-Platform Deployment:

Objective: Increase accessibility and usability.

Description: Adapt the system for deployment across various platforms, including mobile devices, web applications, and desktop applications. This would involve optimizing the system for different hardware configurations and user environments.

8) Explainable AI (XAI):

Objective: Increase transparency and trust in the detection system.

Description: Incorporate explainable AI techniques to provide insights into how the model makes decisions. This can help users understand why a particular video or audio segment was classified as a deepfake, improving transparency and trust in the system.

9) Collaboration with Social Media Platforms:

Objective: Broaden the impact of the detection system.

Description: Partner with social media platforms to integrate the deepfake detection system, helping to identify and mitigate the spread of deepfake content on these platforms. This could involve creating APIs or plugins that social media platforms can use.

REFERENCES

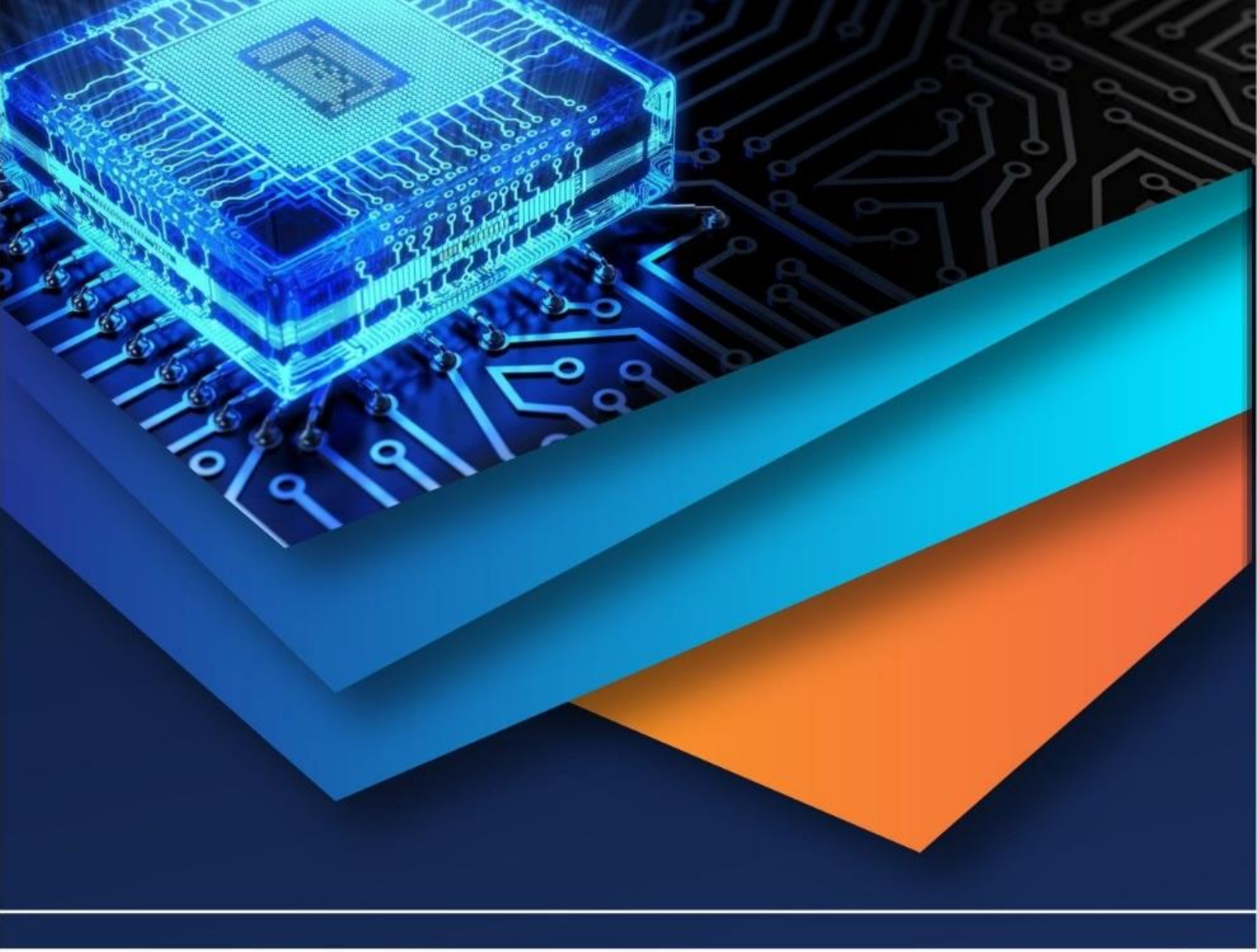
[1] D. Güera, E.J. Delp, Deepfake video detection using recurrent neural networks, in: 2018 15th IEEE international conference on advanced video and signal based surveil- lance (AVSS), IEEE, 2018, November, pp. 1–6 .

[2] Kurniawan Nur Ramadhani, Rinaldi Munir, A Comparative Study of Deepfake Video Detection Method, in: 2020 3rd IEEE International Conference on Information and Communications Technology (ICOIACT), IEEE, 2020, November

[3] Ismail A, Elpeltagy M, Zaki M, ElDahshan KA. 2021. Deepfake video detection: YOLO-Face convolution recurrent approach. PeerJ Comput. Sci. 7:e730 DOI 10.7717/peerj-cs.730

[4] Duha Amir Sultan1,Laheeb Mohammad Ibrahim, "Deepfake Detection Model Based on Combined Features Extracted from Facenet and PCA Techniques", Vol. 17, No. 2, 2023 (19-27)

[5] Neha Sandotra, Bhavna Arora,"A comprehensive evaluation of feature-based AI techniques for deepfake detection", 6 November 2023



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY