Md Ashikuzzaman Esti

A Hybrid Approach to Machine Learning-Based Deepfake Video Detection

School of Technology and
Innovations
Masters Thesis
Computing Sciences

Vaasa 2025

UNIVERSITY OF VAASA
School of Technology and Innovations
Author: Md Ashikuzzaman Esti
Title of the thesis: A Hybrid Approach to Machine Learning-Based Deepfake Video Detection
Degree: Master of Science in Computing Sciences
Major: Sustainable and Autonomous Systems
Supervisor: Mohammed Elmusrati
Thesis Evaluator: Petri Välisuo
Year: 2025 **Pages:** 72

ABSTRACT:

In modern times, people regularly share a large number of videos on the internet by posting in different social media or online video sharing platforms. These videos are floating around the internet and are easily accessible to the public. The emergence of generative AI technology has made the process of manipulating these videos very easy. One common method is to replace an individual's face or copy facial features and movements in a video. The entire process is done so smoothly that the final video looks almost real, even though it is fake. This type of video manipulation is known as a deepfake. Furthermore, it creates serious concerns about security, misinformation, and personal privacy, as these videos often portray individuals doing things they never did. However, these circumstances can be tackled by differentiating fake videos from real ones, creating an effective detection system. In this research, a publicly available UADFV deepfake dataset was selected. A hybrid approach incorporating machine learning and its subset technology was proposed to detect fake videos by analyzing both spatial and temporal features present in frame sequences extracted from the videos. The model was created effectively by combining multiple convolutional neural networks for spatial analysis with a bi-directional recurrent neural network understanding the temporal dependencies across video frames. This hybrid structure detects both frame-level visual inconsistencies and unnatural frame transitions that, in most cases, flag the content as deepfake. In addition to the hybrid architecture, different effective preprocessing techniques were applied to clarify the video frames and highlight subtle inconsistencies before model training. It is also worth noting that interpretability in an AI detection system is very important. Thus, the model also incorporated explainable AI methods to illustrate which area in the facial region impacts the model's final binary prediction the most. The interpretability of this model further validates its final decision. Overall, this study aims to identify deepfake videos with high accuracy using an effective hybrid modelling approach, and also involves interpretability in the model's final outcome.

KEYWORDS: (Generative AI, Deepfake, Spatial, Temporal, Explainable AI, Hybrid)

Contents

Figures

Abbreviations

AI	=	Artificial Intelligence
API	=	Application Programming Interface
AUC	=	Area Under the Curve
BCE	=	Binary Cross-Entropy
Bi-LSTM	=	Bidirectional Long Short-Term Memory
CNN	=	Convolutional Neural Network
CV	=	Computer Vision
DFDC	=	Deepfake Detection Challenge
DNN	=	Deep Neural Network
EVM	=	Euler Video Magnification
FFT	=	Fast Fourier Transform
GAP	=	Global Average Pooling
GAN	=	Generative Adversarial Network
GRU	=	Gated Recurrent Unit
JPEG	=	Joint Photographic Experts Group
KNN	=	K-Nearest Neighbors
LIME	=	Local Interpretable Model-Agnostic Explanations
LRCN	=	Long-term Recurrent Convolutional Network
LSTM	=	Long Short-Term Memory
ML	=	Machine Learning
PPG	=	Photoplethysmogram
ReLU	=	Rectified Linear Unit
RNN	=	Recurrent Neural Network
ROC	=	Receiver Operating Characteristic
SHAP	=	SHapley Additive exPlanations
SSIM	=	Structural Similarity Index Measure
SVM	=	Support Vector Machine
UADFV	=	University at Albany Deepfake Video
ViT	=	Vision Transformer
XAI	=	Explainable Artificial Intelligence

1 Introduction

In modern times, vast amounts of resources are available publicly on the internet, including images and videos of random individuals. This massive number of resources has added scope for researchers in the sector of machine learning applications, such as face recognition and other detection systems. However, this is also the reason that has compelled the growth of deepfake technology in recent years (Caldelli et al., 2021). Deepfake technology is an application of deep learning that can generate entirely new, yet fake, media content, such as images, videos, and sounds, or manipulate and replicate an individual's facial or vocal traits (Abdelkhalki et al., 2022). In different areas of creativity, learning, and entertainment purposes, this is still being used and contributes positively to the general public (K & M, 2024a). Despite the positive impacts, deepfakes pose substantial threats, including the spread of misinformation and invasion of privacy, but primarily deepfake phishing to fabricate audio, video, or image content to appear authentic. Using this technology, cyber attackers tend to manipulate the target into taking actions beneficial to the attackers (Siddiqui et al., 2025).

Furthermore, Countries such as China, the UK, and the members of the EU have developed legal frameworks in response to the malicious use of deepfakes by enforcing watermarking, metadata reveals, and consent-based content creation. In contrast, Australia has thus far been risk-centric with proposals to restrict or ban deepfakes, but attention paid to their value in the content industries has been far less (Broinowski & Martin, 2024).

Because of the rapid process of developing deepfake technology, there has been an increase in research interest in developing effective detection systems (Maheshwari & Paulchamy, 2024). The main tool used by variational autoencoders in the initial development of deepfake technology was the encoder-decoder data transmission process (Heo et al., 2023). According to Siddiqui et al. (2025), this model variant becomes relatively easy to recognize. The first systems were designed to detect inconsistencies

from frame-by-frame based on unnatural facial features and abnormal eye movement or illumination mismatch. Initial achievements were obtained in these detection approaches, but they failed to keep pace with deepfakes having the capability of both spatial and temporal fidelity. As the needs are critical, current requirements demand strong detection approaches to handle the analysis of spatial and temporal video features. Different architecture types and models used in each detection system perform the identification of authentic and fake human characteristics. In response to the current challenges, hybrid deepfake detection systems have been produced by many researchers and technological companies. This involves multiple levels of neural network architecture as well as machine learning algorithms to find patterns such as eye blinking frequencies, skin texture inconsistencies, breathing irregularity, lip synchronization (mismatch between the subject-speech audio and mouth movement), pupil dilation-constriction, facial muscle dynamics, estimation of heart rate via skin color changes (PPG signals) and the temporal flickering or face tracking jitter (shaky or unstable movement of the face when transitioning from one frame to another in a video) (Siddiqui et al., 2025). Since the hybrid system captures spatial features of the subject, including the temporal patterns, it is ideal and justifiable to incorporate the system with both CNN and RNN neural network models in order to fully balance the detection system when it is trained properly and thus able to cover large domain in the fields of deepfake detection systems (Singh et al., 2025).

The main object of this research is to produce a hybrid deepfake detection system through hybrid architectures, modern feature extraction techniques, and frame sequence-based learning methods, along with the integration of Explainable AI to accomplish a maximum percentage of accuracy, making the final result vigorous, generalizable, and easily interpretable by the users. A hybrid deep learning pipeline based on multiple CNN backbones with a sequence learning module is followed by the core detection architecture. Therefore, three strong spatial feature extractors, such as EfficientNetV2B3, Xception, and VGG16, are all deployed in parallel, where each of these feature extractors are encoded inside the TimeDistributed layers, processing each frame

independently as a sequence. The output of the extracted features is concatenated into a single spatiotemporal vector and traversed through a Bidirectional Long Short-Term Memory (Bi-LSTM) network. This final merged output extracts forward and backward dependencies in a frame sequence to learn subtle inconsistencies of facial dynamics. The dataset UADFV, which was trained during the development, comprises over 34,000 real and fake cropped facial frames. Preprocessing of the UADFV dataset involves the use of aggregation of frequency and structure-based techniques to uncover high-frequency anomalies of video frames. Structural Similarity Index Measurement (SSIM) measures perceptual and structural inconsistency between video frames for a practical presentation of the problem to detect manipulated videos in different situations. Additionally, the preprocessed frames are resized and normalized to maintain constant resolution throughout the whole process, which is the fundamental idea before training the model. In order to minimize the overfitting as much as possible, some training parameters like cosine decaying learning rate scheduling, early stopping, and model checkpointing are used to keep the best validation performance. Moreover, accuracy, precision, recall, F1-score, ROC-AUC, confusion matrix analysis, and two Explainable AI (XAI) tools, namely Grad-CAM and LIME, are used to assess the model well and highlight the most impactful specific parts of the video frames.

1.1 Research Questions

The contribution of this thesis is to address the following primary research questions for developing a robust hybrid deepfake video detection system:

1. How to develop a deepfake video detection model towards a hybrid approach combining more than one convolutional neural network with a single recurrent neural network that can successfully identify and differentiate deepfake video contents from real ones?

2. What preprocessing techniques are compatible with the dataset UADFV and should be applied in order to train the deepfake video detection model?

3. What impact does the adoption of Explainable AI (X-AI) tools have on the interpretability, trustworthiness, and evaluation of deepfake detection models?

2 Literature Review

This section will delve into the concept of deep-fake technology, including its background and the steps required in order to generate manipulated contents such as deepfakes, as well as the methods for detecting them. Additionally, the impact of explainability or explainable AI in deepfake evaluation and interpretation will also be discussed.

2.1 Overview of Deepfake Technology

Deepfake technology derives from the implementation of artificial intelligence that manages to imitate images, videos, or vocal sounds of a human subject that replicates realistic human facial features and movement (Broinowski & Martin, 2024). This term, Deepfake, merges the advanced models of deep learning, which is a subset of machine learning, with fake, pointing out its ability to generate visually manipulated content that seems very realistic. Originally, this term was named in an internet blog by a researcher, and various kinds of representation exist regarding its definition, generation processes, and the technologies involved in generating them (Twomey et al., 2023).

Twomey et al. (2023) also described in their work that the extraction of features from the face of a human subject and processing the input data to imprint a new identity or manipulate facial characteristics are crucial in deepfake generation. These collected features, such as expressions, skin texture, and consistent lighting, are the primary variable that quantifies the amount of realism within the output of the result. Modern methods, including autoencoders and generative adversarial networks (GANs), use the most out of these extracted features to produce realistic deepfakes. These models carefully map and blend facial details, which allows the smooth alteration of someone's identity while maintaining a natural appearance and motion within the frame, thus complicating the detection of manipulated contents (Twomey et al., 2023).

Chauhan (2024) describes reciprocating face or face swapping as the most common and essential form of deepfake generation that utilizes autoencoders and face recognition

algorithms to manipulate targeted areas and facial regions. This process involves training two autoencoders in a simplified manner. One focuses on the target face of the person who is going to be imitated, and the other one is taught through various sources of facial structure and texture of frames through the internet. Thus, these data are encoded and fed into the trained decoder to imprint the input data on target face (Chauhan, 2024).

Maniyal and Kumar (2024, p. 5) illustrate in Figure 1 that deepfake technology has the capability to be employed in various ways, depending on both optimal and suboptimal contexts. The fundamental objective of this technology is to replicate an individual's characteristics and transfer those traits to a designated target. This clearly indicates that it can serve either positive or harmful purposes.

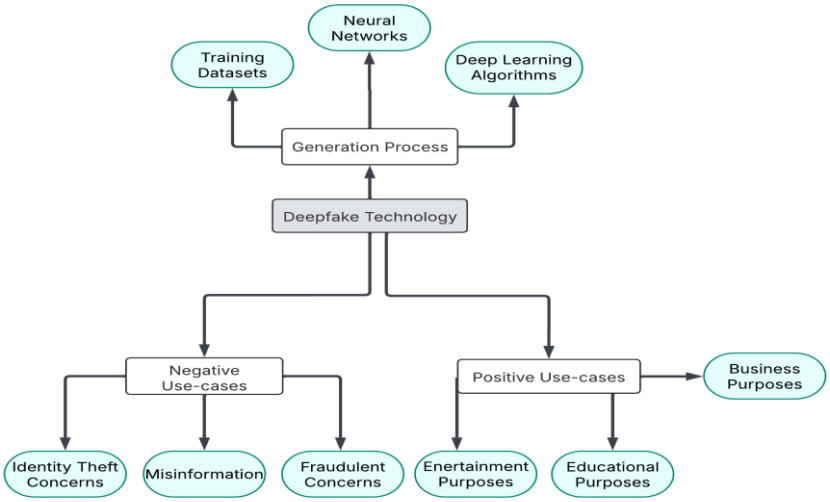


Figure 1. Deepfake generation and use cases (Maniyal & Kumar, 2024, p. 5).

There are numerous beneficial applications, one of which is in the education sector, where it could function as a learning tool to create auto-generated lectures and interactive guidelines for students, making the experience more engaging than traditional lectures or textbooks. Additionally, it holds promising opportunities in the digital marketing realm for content generation and promotion. Furthermore, this technology is accessible to the public, making it absolutely important considering the

responsibility for its positive applications rather than allowing any potential misuse (Maniyal & Kumar, 2024, p. 5).

2.2 Methods for Deepfake Generation

According to Mirsky and Lee (2020), the latest developments in deep learning have considerably improved the techniques used to create deepfake content. One of the most widely used techniques is Generative Adversarial Networks (GANs). In this strategy, two neural networks operate in a continuous active framework. The generator's task is to create artificial media, while the discriminator assesses whether the content is real or imitated. This continuous flow allows both networks to enhance their capabilities. As the generator continuously perfects its output to hoax the discriminator, the discriminator again refines its ability to detect fakes, making the entire system increasingly effective at producing highly realistic results. Another notable technique is Variational Autoencoders (VAEs), which consist of two interconnected autoencoders. The collected features get simplified into a lower-dimensional vector representation by the compression method of the autoencoders. This encoded information is decoded to generate convincing imitated images (Mirsky & Lee, 2022).

In addition to that, there are also many GAN-based face-swapping models. One of the most popular and widely used at a large scale is FaceSwap-GAN, which completely extracts the source facial features and imports them to the target source face (Yang et al., 2019).

In short, cutting-edge deepfakes are produced using advanced deep learning models, particularly neural networks tailored for generative tasks. The two primary architectures that drive deepfake generation are Autoencoders and Generative Adversarial Networks (GANs) (Iglesias et al., 2023). The forthcoming sections will present a thorough analysis and detailed illustration of their architectures, methodologies, and effectiveness.

2.2.1 Autoencoder in Deepfake Creation

In the initial development of deepfake technology, Autoencoders, a specific type of neural network, were employed for direct face swapping through a more straightforward and foundational methodology (Pei et al., 2024). Currently, Autoencoder models such as DeepFaceLab and Face-swap have gained significant popularity in this domain. These models excel in creating highly convincing content with accurate facial alignment and consistent expressions (Nirkin et al., 2022).

Katarya and Lal (2020, p. 3) explain in Figure 2 below that the Autoencoder is composed of two essential functions. The first one is the encoder, and the second one is the decoder. The encoder processes an individual's face as input, transforming it into a set of simplified code segments known as a latent representation, which captures the most important aspects of the facial features. The decoder, on the other hand, recreates the original image from the code segments (Katarya & Lal, 2020).

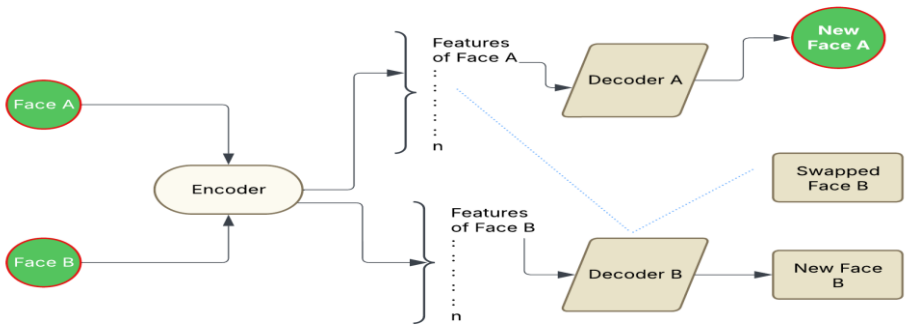


Figure 2. Deepfake generation using autoencoders (Katarya & Lal, 2020, p. 3).

Katarya and Lal (2020) outline the components of autoencoders and the detailed rationale for producing deepfakes, consisting of a system with two decoders rather than only one. Assuming Face A and Face B are two people whose unique facial traits are reflected in their particular data, and this information is trained by a shared encoder. Figure 2 offers a clearer understanding of this method. This encoder learns repeatedly to capture shared facial characteristics between both people, including facial structure,

expressions, and texture. The technique uses two distinct decoders to recreate person A and B's faces. Using their facial expressions, postures, and important facial features, the shared encoder handles both the images of Person A and person B (Katarya & Lal, 2020).

Katarya and Lal (2020) further elaborate in Figure 3 and specifically focus on explaining the stages of faces being swapped. Typically, the encoded information would first be sent to Person A's decoder to reconstruct their original face. However, in this case, the encoded data is sent to Person B's decoder. This allows the decoder to recreate Person B's face while using the expressions and facial features of Person A. As a result, this swaps the identities of Person A with Person B.

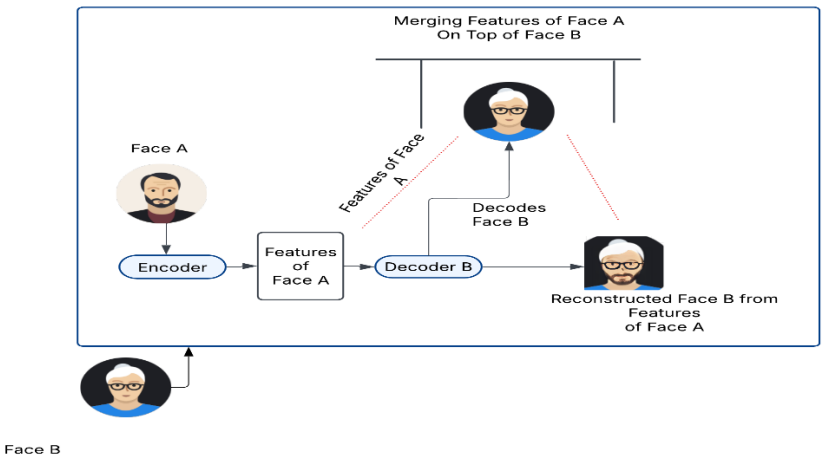


Figure 3. Fundamental face-swapping concept using autoencoders (Katarya & Lal, 2020).

This approach is simple but impactful because the shared encoder is trained to focus on facial features that are common to both individuals, such as the shape of the eyes, mouth movements, and head positioning, which are the major data points when considering generating deepfakes (Katarya & Lal, 2020).

2.2.2 Generative Adversarial Networks (GANs) in Deepfake Creation

Generative Adversarial Networks have progressed over time and are widely used on a broader scale nowadays, specifically for generating deepfakes. This technology plays a vital role for the purposes of creating deepfakes (Preeti et al., 2023). GAN comprises two neural networks that engage in adversarial training, which are the Generator and Discriminator. The first network generator is built to create artificial media content that closely resembles real media, while the discriminator's purpose is to discriminate genuine and artificial content. The constant communication between the two networks drives the imitated content to improve quality. Compelling deepfake content is obtained at the final stages of adversarial training as the generator refines its output, bypassing detection from the discriminator (Seow et al., 2022).

In a study by, Remya Revi et al. (2021), they asses the significance of GANs for generating content that is almost real or imitative. In Figure 4, the fundamental architecture of a GAN is illustrated, in which there are two essential components: the generator (G) and the discriminator (D). These are adversarial networks that compete against each other in a process that improves their ability. The random noise as input is first ingested by any given generator and sampled from predetermined mechanisms. The source of this noise is further used to generate artificial data. The unstructured noise is routed through a sequence of convolutional and upsampling layers within the generator. The final generated result is refined in this process, creating synthetic images that aim to mimic the properties of real-world images. That is why the discriminator acts as a classifier, distinguishing between genuine images from the training dataset and fake images created by the generator. Furthermore, during the analysis of both authentic and synthesized images a probability score is labeled, indicating the possibility of the images being genuine or fake. The score is labeled according to the binary value 0 or 1, 0 indicating real and 1 for fake. The crucial continuous communication is the fundamental idea behind this process (Remya Revi et al., 2021).

The continuous exchange of information is summarized by Remya Revi et al. (2021) in the following objective function:

$$\min_{G}\max_{D} V\,(D,G) = E_{x\sim p_{data}(x)}[\log D\,(x)] + E_{n\sim p_n(n)}\left[\log\left(1 - D\big(G(z)\big)\right)\right], \qquad (1)$$

In this equation, $G(z)$ represents the generator opting to convert random noise into realistic data. In contrast, $D(x)$ is the discriminator's prediction of the authenticity of an image. The term $p_{data}(x)$ denotes the actual data distribution from the training dataset, and $p_n(n)$ is the noise distribution used as an input for the generator.

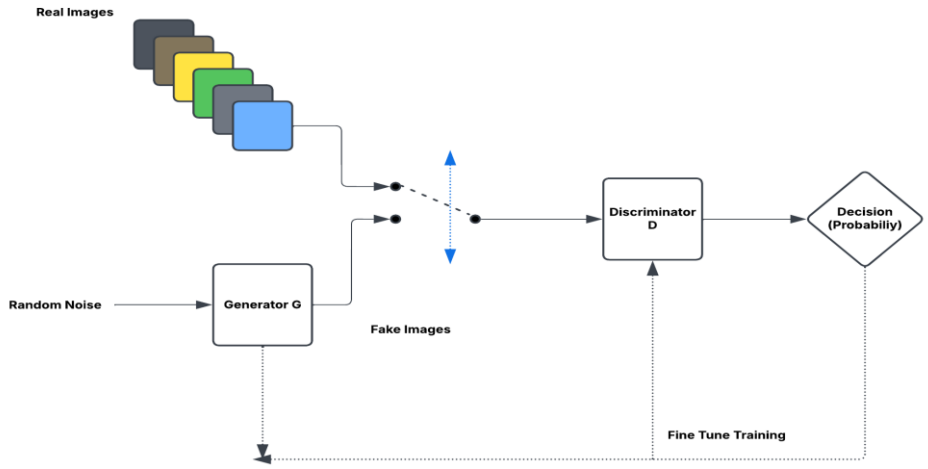


Figure 4. Fundamental face-swapping concept using Generative Adversarial Network (Katarya & Lal, 2020, p. 3).

The above Figure 4 explains that the discriminator's absolute tendency is to maximize the probability of accurately detecting real and fake images. At the same time, the generator thrives on minimizing the discriminator's capacity to recognize its fake generated content. This competitive nature forces the generator to improve its outputs until the final generated fake content become indistinguishable from accurate data. Finally, after a certain point, the networks reach an equilibrium state, making the

generator's outputs so reasonable that the discriminator can not reliably distinguish the generated images from actual data (Remya Revi et al., 2021).

Furthermore, almost every day, some part of GAN technology is being used in three major sectors: Face swapping, target facial region enhancement, and basic facial synthesis. Models such as StarGan and InterFaceGan leverage multi-domain transfer learning to achieve these results (Pang et al., 2023).

In the majority of cases, GAN-based deepfake generation leads to decent, realistic content of human countenance while conserving the stability of facial feature extraction (Guarnera et al., 2022).

Jiang et al. (2021) explained in their work that GAN-based technology excels in acquiring meticulous and concentrated feature exploitation by blending latent spaces between reference and original faces. Even though the outputs generated from the GAN-based technology are worthy of attention, it has a noteworthy downside, causing perceptible degradation of generated outputs, exceptionally in low-light or low-resolution environments, either exposing the manipulated region of the face or completely changing the expected results (Jiang et al., 2021).

2.3 Deepfake Detection Techniques

According to Yan et al. (2025), the increasing number of deepfake generation is constantly being used, both for positive and negative purposes, still requires detection if the necessary situation arises. In most cases, deepfake-generated content is difficult to identify with the simple eye, thus it is necessary to utilize modern tools and technology to identify this content. There are various approaches, ranging from traditional, simple machine learning algorithms to advanced, deep learning, neural network-based approaches.

The first approach is simple machine learning-based algorithms, which refers to detection techniques that use manually modified features (e.g., eye blink frequency, head pose, color histograms) extracted from videos or frames, merged with simple classifiers (like Support Vector Machines, Decision Trees, or K-Nearest Neighbors) to determine the authenticity of the input frames. These techniques are often employed for systems that can operate the deepfake detection program with limited computational resources (Yan et al., 2025).

Secondarily, due to the advancement of GAN technology, single-architecture-based deepfake detection techniques have been introduced, where only one single-architecture-based neural network is involved and trained against each frame of a video to classify inspected frames or videos as real or fake. This can either be CNN or RNN, but the majority of the cases are CNN-based single-architectures that perform well in image classification tasks. The architecture includes networks like Convolutional Neural Networks (CNNs), Vision Transformers (ViTs), and Capsule Networks (Adnan & Abdulbaqi, 2022)

Lastly, after considering the drawbacks of the previously used architectures in many cases in terms of achieving a high quality of accuracy in deepfake video detection, ongoing research is increasing at a rapid rate. Primarily, these architectures involve utilizing different structures of networks that are capable of handling or detecting deepfakes faster and accurately. These combinations could be CNNs with RNN, incorporating temporal learning modules, for example, LSTMs or GRUs (Singh et al., 2024).

2.3.1 Simple Machine Learning Approach

Early approaches to detecting deepfakes relied on manually trained and simple machine-learning algorithms (Pei et al., 2024). According to the study by Li et al. (2018), GAN-based technology is capable of manipulating videos using multiple facial features, textures, and tones trained from large datasets, but in many cases, it fails to capture the

genuine human anatomical eye blinking patterns, which is a significant factor for detecting deepfake videos. To begin with, multiple authentic videos were collected from face-to-face human interviews, specific eye-blinking patterns were more clearly captured through the device, and fake videos were collected using GAN-based technology, which primarily involved face-swapping and manipulation. After that, only the eye regions were extracted, and a landmark identification technique consisting of a block of code was applied, which is capable of pinpointing the key points surrounding both eyes, which goes through a manually created metric known as the eye aspect ratio (EAR). This process further calculates the frequency of eye openness by computing the absolute distances between specific landmarks around the eye region. The value of EAR changes according to the appropriate calculation of key points. While the eye is open, EAR sustains comparatively incremental and stable values, but during blinking, EAR drastically declines due to eyelid closure, providing a clear temporal signature of blinks. After per-frame calculations, a temporal EAR signal with a time-series representation of eye-opening variations across frames in each video segment is created. These data immediately showed that the blinking rhythms in deepfake motion videos were artificial or incorrect. The rhythm was quite different from the typical rhythmical structures observed in authentic videos. Authentic videos, for instance, exhibited clear periodic dips in the EAR signal that were exactly equal to natural blink intervals of time that occur roughly every couple of seconds. Deepfake videos, however, always showed asymmetrical oscillations, which was clearly a telltale sign that the videos were artificial (Frank et al., 2020; Y. Li & Lyu, 2018; Rafique et al., 2023; Wolter et al., 2022; Yang et al., 2019).

Frank et al. (2020) clarified in their paper that GAN-generated images contain frequency-domain artifacts with subtle visibility, a periodic pattern due to the structure elements in GAN-based architectures, such as repeated upsampling layers. Real images have unique frequency distributions. Subsequently, each image was transformed from the spatial domain to the frequency domain. Using the discrete Fourier transform, breaking down each image into frequency components gives rise to a two-dimensional frequency

spectrum. Simple machine learning classifiers were utilized to distinguish real and generated images on the basis of frequency-based features. Logistic regression and linear Support Vector Machines (SVMs) were particularly applied because they are interpretable, efficient, and powerful in performance. These classifiers operated directly on the averaged frequency vectors and effectively distinguished authentic images from GAN-generated ones because the periodic artifacts visible in GAN architectures became linearly separable in the reduced feature space (Frank et al., 2020).

Another study by R. Das et al. (2021) utilised micro-movements accompanied by blood circulation and respiration, called Euler Video Magnification (EVM), a powerful video-processing tool originally developed to highlight precise motions and changes in the color, in most cases, not visible to the open eyes. Initially, their method began by capturing real video data, which naturally occurs as micro-variations caused by human physiological movements, such as skin pulsation due to the heartbeat and subtle movements incorporated with the respiration cycle. Deepfake videos, in contrast, were generated with the aid of GANs but lacked naturally occurring, subtle bodily cues, mainly because GANs focus relied on visual realism in static frames or facial expressions and overlooked temporal bodily signs. The EVM operates in the first place by decomposing the frames in a video into spatial frequency bands and subjecting these to temporal filtering to decompose and amplify precise bodily movements, such as fluctuations in skin color due to heartbeat or flux of motion artifacts due to respiration. The real videos, when processed with the aid of EVM, revealed strong, periodic color and motion changes in accordance with real bodily signs. Deepfake videos, in contrast, when magnified with the aid of EVM, revealed a strong lack of inconsistency in such signs.

The researchers used simple and easy-to-interpret statistical descriptors based on the magnified video signal rather than the complex, high-dimensional deep features. For example, applying manually implemented features that directly reflect the temporal coherence and realism of magnified videos, such as signal variance, periodicity, and frequency distribution of the pixel intensity changes in a transparent manner. Using

these manually created features is an effective option since the process becomes easy to compute, easy to interpret, and has an intuitive connection with genuineness. Simple but strong, classical machine learning classifiers were then used in conjunction with these manually created features. It has consistently shown a strong ability to detect inconsistencies in deepfakes (R. Das et al., 2021).

In addition, according to the research by Wolter et al. (2022), their work depicts the excellence of deepfake detection and classification via a comprehensive but interpretable wavelet-packet transform-based machine learning approach. Subsequently, the collected fake and real images from the dataset are split into several frequency bands via wavelet packet transform to simultaneously preserve spatial and frequency domain information. In particular, the wavelet packet decomposition algorithm recursively divides the original image data into a given sub-band to obtain the coefficients in the different resolution levels, allowing very precise analysis of image structures. The resultant sub-bands strongly emphasize faint artifacts introduced by generative adversarial networks, such as abnormal textures, unusual spatial frequency patterns, and faint distortions not easily visible in the original spatial domain. In this sense, classical machine learning algorithms, properly coupled with specifically designed and interpretable wavelet-based statistical features, perform in an efficient and effective way in detecting deep fake images under different scenarios (Wolter et al., 2022).

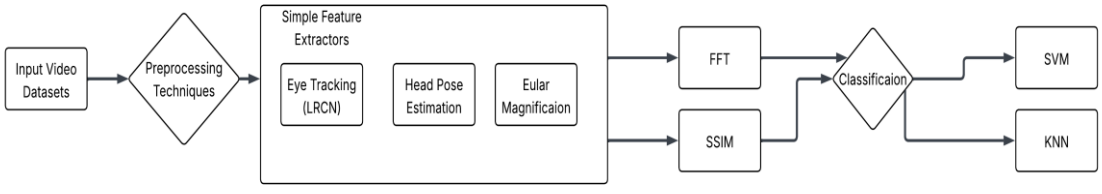


Figure 5. Commonly Used Traditional Deepfake Detection Approaches. Adapted from Li et al. (2018), Frank et al. (2020), Das et al. (2021), Yan et al. (2025), Pandey et al. (2024), and Rafique et al. (2023).

Summarizing the above-illustrated Figure 5, it can be described that the above-displayed techniques are the most conventional form of simple traditional deepfake techniques.

In addition, the procedures involved in using these simple techniques to create any basic deepfake detection model remain the same. Firstly, a single or large dataset consisting of videos is taken as input. Secondly, a combination of three main feature extraction approaches, Eye Tracking using LRCN, Head Pose Estimation, and Euler Video Magnification, is employed. Feature extraction methods collect features that are further analyzed based on Fast Fourier Transform (FFT) and Structural Similarity Index Measure (SSIM) to detect subtle anomalies and inconsistencies. Lastly, the input media contents are classified as real or fake using traditional machine learning classifiers, including Support Vector Machines (SVM) or K-nearest neighbors (KNN) (R. Das et al., 2021; Frank et al., 2020; Y. Li et al., 2018; Rafique et al., 2023; Yang et al., 2019).

2.3.2 Single Architecture Model

Zhao et al. (2021) claim that their research offers a robust CNN-based single architecture that is capable of efficiently identifying deepfake videos by capturing subtle yet discriminative visual artifacts present in the video. GAN-generated videos comprise a variety of subtle inconsistencies and synthetic artifacts that distinguish them from real videos. They start their approach with a dataset generated from diverse publicly available deepfake sources, FaceForensics++, and DeepfakeTIMIT. The datasets presented are comprehensive in terms of their range of synthetic manipulations generated by different GAN models, thus allowing the model to effectively generalize beyond particular training datasets. The authors highlight that CNNs are able to learn robust, generalizable features that are indicative of deepfake manipulations with training and validation datasets that are diverse. Robust facial landmark detection techniques are employed to detect, crop, and align the faces across frames. The CNN model utilizes convolutional layers to represent subtle GAN-produced artifacts, both in a hierarchical and higher-to-lower abstract manner, for example, unnatural texture, color faults, and, consequently, facial landmark discrepancies. It explicitly points out the critical challenge of data augmentation via dedicated strategies that improve robustness with respect to data perturbation. In particular, they introduced realistic variability in illumination, pose, and resolution upon their augmentation approach, forcing the CNN to learn invariant

features that are robust under such variation. This augmentation greatly enhances model generalization beyond the particular test cases and, hence, provides robust detection even in scenarios that involve untrained or unseen manipulation methods (Zhao et al., 2021).

Another work presents a novel Fast Parallel CNN architecture that has parallel convolutional layers to achieve higher computational efficiency and higher detection speed. Unlike the traditional CNN models that perform convolution layers sequentially, the FPC tends to divide the input data into multiple parallel branches to extract supportive yet discriminative features in parallel. All the branches here are convolutional layers with different kernel sizes and stride patterns, tailored to capture multi-scale and diverse visual prompts in parallel. In particular, the parallel design allows the CNN model to be much faster and more robust while processing multiple feature dimensions in parallel in an efficient manner. The CNN architecture also includes a dedicated feature selection module, which further improves the performance. Classical feature selection methods are used in the module through such algorithms as Recursive Feature Elimination and Principal Component Analysis. The objective of this feature selection operation is to gather the most discriminating and relevant features obtained from the parallel convolutional layers and remove redundant or irrelevant information. These classical feature selection methods are successfully employed to reduce computational overhead and improve the efficacy of the CNN model to attain faster real-time processing speed and interpretability. Finally, fully connected layers are used for final classification with the FPC architecture. Based on the compact, selected feature representations by the parallel convolutional branches and the feature selection module, the layers are built up to efficiently classify the real and fake contents (A. Das & Sebastian, 2023).

According to Caldwell et al. (2021), in a study on the detection of deepfake video, a new approach has been proposed to extract temporal features from extracted video frames without the use of recurrent neural networks. The proposed method actually makes use of a CNN-based architecture coupled with the computer vision technique of optical flow.

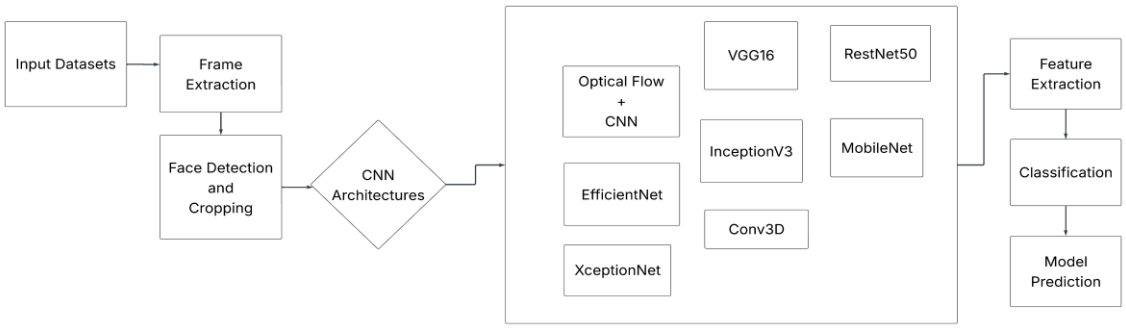


Figure 6. Commonly Used Single-Architecture-Based Deepfake Detection Approaches. Adapted from Li et al. (2018), Frank et al. (2020), Das et al. (2021), Yan et al. (2025), Pandey et al. (2024), and Rafique et al. (2023).

The illustrated Figure 6 represents the most commonly used pre-trained models that are used to capture spatial features from video datasets in deepfake video detection research. Each of these models has its own expertise in detecting deepfakes. VGG16 performs well in capturing texture inconsistencies, RestNet50 is well known for detecting framewise subtle distortions, EfficientNet is skillful and performs above average even in a limited dataset size environment, InceptionV3 is effective at extracting small and large scale inconsistencies parallelly at the same time, MobileNet is a lightweight model integrated to achieve decent accuracy and XceptionNet performs adeptly in capturing complex features. Finally, the Optical flow computer vision technique can be implemented with any one of the models to capture motion inconsistencies (Ajoy et al., 2021; Caldelli et al., 2021; A. Das & Sebastian, 2023; Fahad et al., 2025; Jolly et al., 2022; Kohli & Gupta, 2021; Liu et al., 2021; Solaiman & Rana, 2024; Suratkar et al., 2020; Tran et al., 2021).

2.3.3 Hybrid Architecture Model

According to many researchers, it is understandable that a hybrid model strictly involves combining more than one neural network model and receiving a single output capable of extracting different forms of features from human faces, which was primarily lacking in the single architecture-based model in terms of detecting deepfake videos (Kaddar et al., 2024; S. A. Khan & Dang-Nguyen, 2022).

S. Khan (2024), introduced a unique hybrid architecture able to boost deepfake detection accuracy by combining the spatial characterization factors of convolutional neural networks and the global attention capabilities of transformer models. Furthermore, their work states strong steps of data preprocessing. To keep the quality of datasets high, different datasets are used from publicly available internet sources (FaceForensics++, Celeb-DF, DFDC). Facial areas are detected using face detection models and then cropped, aligned, and resized so that the input data is consistent and maintains quality. Consequently, the quality of the data improves greatly when training the hybrid model. The CNN component uses a ResNet-based architecture to effectively capture local target features present in facial textures, edges, and fine-quality GAN artifacts. This CNN layer forms compact hierarchical feature representations that are tuned for local visual features. CNN spatial features are overlaid for processing a vision transformer to learn the temporal relation. The vision transformer works on patches extracted from CNN feature maps and applies multi-headed self-attention modules to capture global long-range context and spatially dispersed anomalies that are often difficult to discern solely with CNNs. By introducing transformers purposefully, this approach considerably bolsters the model's capacity to comprehend GAN-spawned visual aberrations that go beyond localized aspects, precisely seizing intricate dependencies of artifacts across the complete facial spectrum. Finally, implementing a learning rate schedule, early stopping, and several regularization methods (i.e., dropout layers) during the training greatly enhances the training stability as well as generalization performance. Their hybrid model consistently delivers remarkably high accuracy above precision, recall, and F1 Scores, explicitly outperforming standard CNN-only and transformer-only architectures (S. Khan, 2024).

Some GAN-based models can generate videos that are not solely detectable by capturing temporal or spatial features. To fortify the robustness, generalization, and interpretability of detection, researchers apply multi-modal features (visual, audio, and textual metadata) and multi-scale transformer mechanisms in a strategic manner. This

process follows a unique methodological process with a systematic collection and pre-processing of a large multimodal dataset extracted from popular benchmarks. The methodological improvement is the development of a Multi-modal, Multi-scale Transformer (M2TR) architecture. In this approach, they combine features of modality (CNNs for images, RNNs for audio features, transformer encoders for text data), generating multimodal embeddings, which are then fed into a sophisticated multi-scale transformer architecture. Multi-scale transformer leverages parallel attention heads at different spatial-temporal scales to capture complex modal consistency across different levels of abstraction. Specifically, the advantage of the explicit methodology of M2TR stems from its hybrid structure. Additionally, multi-scale transformer blocks, model and improve representation capability over global dependency, long-range time serial relationships, and cross-modal relations, which single-modality models often neglect (S. Khan, 2024; Salvi et al., 2023; Yu et al., 2024).

Sudarshana and Vamsidhar (2025) introduced a hybrid detection architecture in their insightful and innovative research, combining CNNs, ViTs, and advanced attention mechanisms. This architecture incorporates a unified attention mechanism network, which takes advantage of the spatial pattern recognition abilities of CNNs and the global attention modelling strengths of ViTs (Sudarshana & Vamsidhar, 2025).

Another advanced study by K and M (2024b) developed the HODFF-DD framework, a hybrid deepfake detection framework that improves accuracy and generalization across video modification approaches. The model uses spatial and temporal feature analysis. It extracts facial characteristics from video frames using two convolutional neural networks, InceptionResNetV1 and InceptionResNetV2. The generated outputs are concatenated to form a more complete frame feature representation. Long Short-Term Memory networks carry concatenated features. This assists the model in understanding the time-dependent dynamics of facial expressions and movements while detecting tampering. The spatial richness of CNNs and the temporal sensitivity of LSTMs create a dual-layered forgery protection system. Another notable characteristic of HODFF-DD is

its use of a bio-inspired spotted hyena optimizer to better adjust model parameters. This technique navigates complex optimization landscapes to aid the model in avoiding local minima and finding superior solutions. This results in lower cross-entropy loss and enhances detection accuracy (K & M, 2024b).

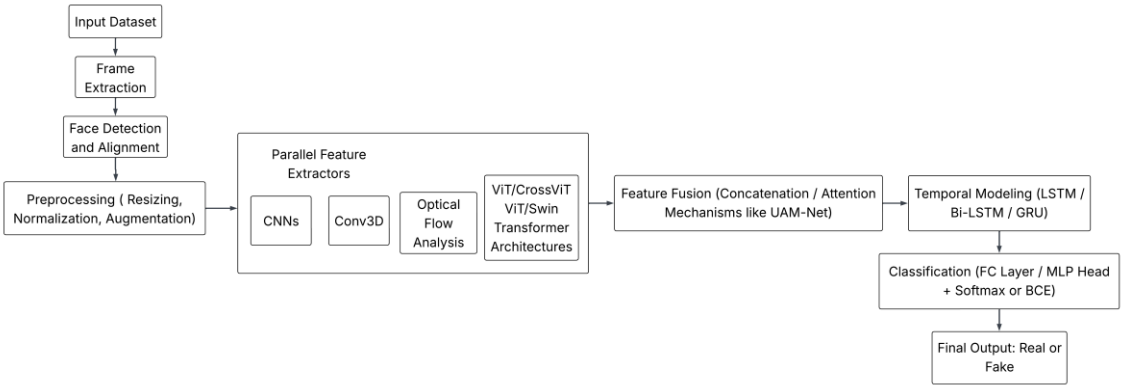


Figure 7. Overview of hybrid architecture-based deepfake video detection. Adapted from Gong et al. (2023), Heo et al. (2022), Helode et al. (2025), Kaddar et al. (2023), Khan and Dang-Nguyen (2022), Pandey and Kushwaha (2024), Siddiqui et al. (2025), and Singh et al. (2024).

The illustrated Figure 7 above provides a comprehensive overview of the commonly applied hybrid deepfake video detection pipeline that integrates both spatial and temporal feature modelling techniques. As usual, the process starts from dataset collection to preprocessing and selects or combines multiple architectures from the parallel feature extractor block, creating a strong hybrid model that captures complex spatial inconsistencies. The collected features are integrated using either concatenation or attention modules. The merged features are passed through a type of Recurrent Neural network, such as LSTM/BI-LSTM/GRU, to make the detection model completely hybrid, which captures both spatial and temporal features effectively. Finally, Softmax or Binary Cross-Entropy (BCE) is applied to achieve the final hybrid deepfake video detection predictions, which are either real or fake (Gong et al., 2024; Helode et al., 2024; Heo et al., 2023; Kaddar et al., 2021; S. A. Khan & Dang-Nguyen, 2022; Pandey & Kushwaha, 2024; Siddiqui et al., 2025; Singh et al., 2024; Sudarshana & Vamsidhar, 2025).

2.4 Explainability in AI-driven Deepfake Detection Models

With the rapid growth of users in the field of deepfake technologies, there is now a dire need for transparency and trust related to all kinds of detection systems, particularly AI-based ones. Deepfakes can be integrated into almost any type of media. Therefore, they pose a significant security concern for the general public. Although impressive accuracy percentages exist in DL models like ResNet-50, Inception V3, and XceptionNet in the field of detecting deepfakes, a major challenge still remains: the lack of interpretability in these models. Researchers still seek an explanation of how these models operate in a logical manner, rather than merely being classified as fake or real. Therefore, there should remain an intelligible rationale behind these model predictions when providing status, such as real or fake, thereby creating a major focus on explainability or X-AI (Gowrisankar & Thing, 2024; Mansoor & Iliev, 2025).

Gowrisankar and Thing (2024) explained in their research that in current times, XAI tools are used to point out the particular regions in the frame that influence a model's decision from most to least. Some tools are used for evaluation, such as saliency maps, gradient-based shadings like Grad-CAM, or model-agnostic tools like LIME and SHAP. Specifically, they found in their study that the application of XAI tools such as SOBOL, Grad-CAM, and LIME on the same real image provided differing regions of focus on the left eye, nose, and hair, respectively, underscoring the ambiguity and subjectivity in explanation outputs. Moreover, it also described traditional XAI evaluation techniques, for example, pixel flipping or salient region removal, which are decently effective in object identification tasks but fail in the area of deepfake systems. These methods often assume that removing important regions will reduce model confidence in proportion to the importance of the removed regions. Furthermore, to overcome this issue, the researchers have focused on domain-specific evaluation strategies in their current research. They proposed a unique adversarial evaluation framework that utilizes real and fake image pairs, perturbing the salient visual concepts of authentic images in corresponding fake images. This targeted adversarial attack measures the drop in classifier confidence, thereby validating the correctness of the XAI tool used to generate

the saliency map. Their approach demonstrates that Grad-CAM and XRAI outperform others in identifying visually and semantically meaningful regions for deepfake detection, particularly in comparison to generic methods that do not account for facial spatial dependencies (Gowrisankar & Thing, 2024).

There is a significant importance factor linked with the ability to explain the model's prediction in trustworthy detection systems, which can be achieved by integrating the network dissection algorithm to interpret the hierarchical feature activations in CNNs. The interpretability enhances stakeholder trust and supports the forensic and legal acceptability of AI-based evidence, especially in scenarios where human judgment and accountability are crucial (Mansoor & Iliev, 2025).

Moreover, Maheshwari & Paulchamy (2024) state in their work that by combining X-AI with ART in a hybrid deepfake detection framework, their model achieved not only higher accuracy, which is close to 97.5%, but also received transparent insight into the decision-making part of the processes in models predictions, thereby increasing user confidence in system outputs. They also stated that this kind of explanation and transparency is very important in essential industries, such as news broadcasting platforms, law enforcement, and digital media regulation, where explainability is not optional but necessary for the ethical deployment of AI (Maheshwari & Paulchamy, 2024).

2.4.1 Explainable AI Techniques

Implementing explainability in deepfake detection draws on methods from the extensive area of XAI, specifically computer vision and machine learning. Quite a few approaches have been explored in order to understand the decisions of deepfake detectors (Venkateswarulu & Srinagesh, 2024).

Saliency Maps and Heatmaps: One of the most direct ways to explain an image classification (such as distinguishing between deepfakes and real images) is to pinpoint

which parts of the image most affected the model's decision. Methods such as Grad-CAM and other saliency map techniques can be applied to the layers of CNN to generate a heatmap over the input frame (Mansoor & Iliev, 2025). The heatmap shades regions that most influenced the fake prediction. For example, an explainable deepfake detector, in most cases, outputs a face image with specific areas, such as the eyes, mouth, or edges, colored to indicate high model attention. This factor reveals a clear understanding that the detector focused on the eyes and lips of a face as evidence of manipulation. In practice, such visual interpretability has been demonstrated: researchers have shown heatmaps on fake images where the model zeroes-in on the unnatural eye and lip regions rather than, the background. These visual explanations have been tested against human intuition and the results align in majority of the cases. Many deepfake detection systems incorporate this kind of visual explanation interface (Gong et al., 2024).

Feature Attribution Methods: Model-agnostic explanation tools, such as LIME or SHAP, can also be used as an evaluator for the deepfake detector. These methods consider the model as a non-transparent entity and study it by processing the input to see how the prediction changes. For instance, LIME can take a video frame and break it into superpixels, then systematically zero-out or alter parts of the image to see if the prediction flips from "real" to "fake." By repeating this process multiple times, LIME constructs a straightforward, interpretable model, such as a linear model, that approximates the complex model's behavior locally, indicating specific parts of the image that contribute positively or negatively to the fake classification. SHAP utilizes coalitional game theory to compute Shapley values, assigning significance to features such as pixels or regions. The advantage of these methods is that they do not require alterations to the detection model, as they can be applied to any classifier. In short, Grad-CAM is fast and provides coarse localization, while SHAP is considered to be computationally heavier (Gong et al., 2024; Tsigos et al., 2024).

Prototype and Network-Dissimilation Approaches: This method is used to employ network dissection or prototype visualization. Network dissection entails mapping

neurons or filters in a CNN to human-interpretable concepts, such as textures, objects, and facial features. In the realm of deepfake detection, certain filters may exhibit a pronounced response to compression noise or the existence of double edges surrounding a face. Furthermore, it becomes more evident that the model's decision-making is contingent upon principles such as the identification of abnormal eye blinking activity and discrepancies in skin texture. For example, it is activated by the engagement of neuron X in relation to eye blinks and neuron Y in response to the smoothness of skin texture (Bouter et al., 2023; Trinh et al., 2020).

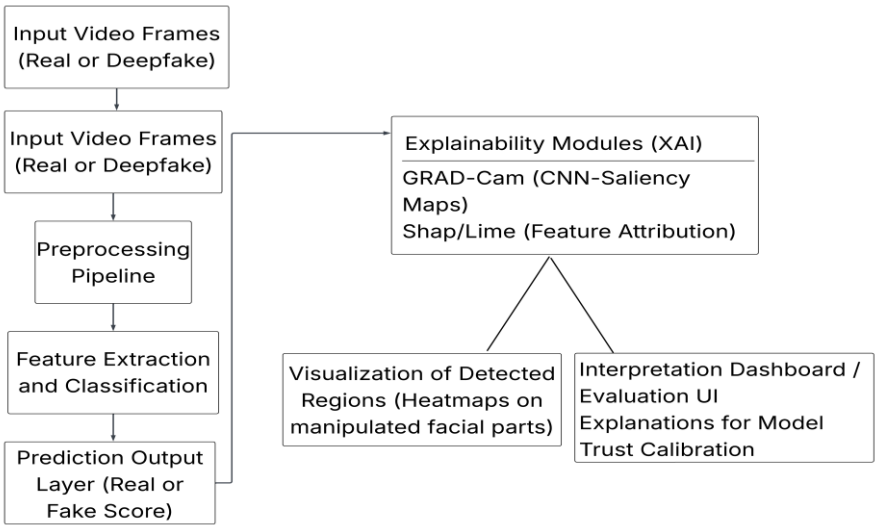


Figure 8. General Outline of Explainable AI Techniques. Compiled from Venkateswarulu and Srinagesh (2024), Mansoor and Iliev (2025), Gong et al. (2024), Bouter et al. (2023), and Trinh et al. (2020).

Figure 8 accommodates an overall insight into the Explainable AI techniques used for interpreting the deepfake detection models. The working mechanism of Explinable AI gets initiated after the prediction (real or fake) of deepfake detection models. The input frames of datasets are passed through Explainable AI modules such as Grad-Cam and Shap/Lime, producing saliency maps over CNN-based feature activations. These methods serve the purpose of an evaluation tool or the interpretation of model results by visualizing or highlighting the most important areas of the model's prediction (Bouter

et al., 2023; Gong et al., 2024; Maheshwari & Paulchamy, 2024; Mansoor & Iliev, 2025; Trinh et al., 2020; Venkateswarulu & Srinagesh, 2024).

3 Research Methodology

Deepfakes generally target the facial region. Therefore, to completely detect different categories of Deepfakes, the proposed methodology combines both CNN and RNN, a hybrid architecture focusing on targeting spatial and temporal patterns of facial regions.

3.1 Dataset Selection

The UADFV (University at Albany Deepfake Video) dataset, which is publicly available for research purposes, was primarily selected for training and analyzing the introduced hybrid deepfake detection model (Li & Lyu, 2018).

Figure 9. The UADFV dataset **(Y. Li & Lyu, 2018).**

The dataset comprises 49 fake and 49 real videos of human individuals, which add up to a total of 98 videos for the dataset. Each video consists of the human subject's movement. Fake videos were generated through GAN-based technology, and the real ones were collected from highly trusted sources. The dataset is well structured, lightweight, and consists of controlled video conditions and clear manipulation labels,

which makes it a strong foundation during the integration of the deepfake detection model (Y. Li & Lyu, 2018).

3.2 Proposed Steps for Detecting Deepfakes

One of the leading research areas in the field of computer vision, as well as machine learning, has been the methodologies of deepfake detection. Most of these methodologies consist of a number of stages of processing and computation. Frame extraction, facial feature detection, temporal feature analysis, and integration of hybrid models are the fundamental stages that will be discussed in the forthcoming sections.

3.2.1 Facial Feature Extraction

The first step of the proposed method extracts frames and crops facial regions by using OpenCV and RetinaFace. The major process involves isolating and enhancing facial features from each video frame to expose subtle artifacts that are indicative of deepfakes. In this case, all frames are cropped to the face region and standardized to a consistent resolution (e.g., 128 × 128 pixels) with pixel value normalization to provide a uniform input to the model. Several image transformation techniques are also used for each face-frame to exaggerate the suggestive inconsistencies due to manipulation. Additionally, the images undergo a Fast Fourier Transform (FFT) to reveal anomalies in the frequency domain, where deepfake forgeries often introduce unnatural frequency patterns. This process further involves parallel computation of a Structural Similarity Index (SSIM) map that captures structural distortions of the original frame compared to a mildly degraded version. The SSIM localizes regions where the structural integrity (lighting, contrast, and texture) differs. Furthermore, a JPEG compression simulation is applied to the frame to imitate real-world compression artifacts, and any extreme response to this compression can indicate texture inconsistency indicative of fake content. The obtained representations are merged with the original face image to create a composite that enhances possible deepfake artifacts. In particular, the original RGB frame, the FFT output (frequency anomalies), the difference map calculated from the SSIM (structural

inconsistencies), and the JPEG simulated image are blended using weighted blending. The outcome of this facial feature extraction step consists of a face image per-frame, enriched with the original content embedded with additional frequency and structure cues. The main goal of incorporating this preprocess technique was to make subtle forgery artifacts more salient for the subsequent feature extraction by the deep learning model. By the end of this stage, each raw face frame has been turned into an enhanced standardized image, which demonstrates a good starting point for detecting deepfake traces in the subsequent stages.

3.2.2 Temporal Feature Collection

The proposed methodology goes further beyond analyzing individual frame anomalies as the approach also utilizes temporal information by collecting sequences of frames. The facial images are systematically grouped all together according to their respective video identity, such that frames from the same video source are kept together. A consistent naming convention or identifier of the video (video ID) is extracted from the frame filename to group frames for the same video. Therefore, a sequence is recognized by frames, from frame0 to frameN, of a single video.

Fixed-length sequences of five consecutive frames are constructed from each video's representative frames in order to capture motion cues and temporal inconsistencies, which may not be noticeable in isolated images. A balance of five frames is chosen as the window to capture short-term temporal dynamics without having a very large input length for the model. Overlapping sequences are generated by applying the sliding window technique. To give an example, frames 1–5 are the first sequence, frames 2–6 are the second sequence, and so on, until all frames have been included in a sequence. This overlap allows for an effective increase in the number of training samples and guarantees that the important transitions in a video are kept at the boundaries of nonoverlapping segments, which improves model generalization.

The model arranges the five-frame sequence in chronological order so that it can detect the progression of facial movements and anomalies over time. The method is designed to prepare data in this way so that temporal artifacts, including inconsistent facial expressions, flickering due to poor face blending, or unnatural motion dynamics that are often present in deepfakes but that may be difficult to detect in isolated frames, can be detected. Furthermore, each frame sequence is labeled with a class label according to its source video, either 'real' or 'fake', allowing for a sequence-to-label learning method. One hot encoding method is used to encode the class labels ([1,0] for real and [0,1] for fake) so that a binary vector representation is clear for the classification task. The dataset is prepared as a robust dataset of sequential data by aligning each short frame sequence with a corresponding ground truth label, and this is adopted to train the hybrid deep learning model.

3.2.3 Integration of Hybrid Model

The hybrid architecture model is integrated to retrieve both spatial and temporal characteristics in order to derive the authenticity of an input sequence. This approach mixes (to a certain degree) the strong features extraction capability of Convolutional Neural Networks (CNNs) with the features extraction capability of a Single Recurrent Neural Network (RNN), which is particularly good in cases where extracting features from sequences of frames is demanded. The aggregated feature outputs from multiple CNN features for each frame are generated in this framework, including feature vectors generated from EfficientNetV2B3, XceptionNet, and VGG16. This concatenates the vectors to build a cohesive feature vector of each frame. Therefore, the concatenation of this process results in a 4096-dimensional feature vector for each frame consisting of 1536 dimensions from EfficientNet, 2048 dimensions from Xception, and 512 dimensions from VGG16. This fusion step combines diverse feature descriptors from the parallel CNNs, acquiring an extensive range of facial characteristics in a single representation. The concatenation is performed frame-wise, preserving the temporal ordering. After the process of fusion, each 5-frame sequence gets represented as a sequence of 5 fused feature vectors (each of length 4096). The parallel CNNs generate

diverse feature descriptors, providing a single representation that encompasses diverse facial characteristics. The temporal modeling component consists of bi-directional Long Short-Term Memory (Bi-LSTM) network that takes these fused features as input. The Bi-LSTM feeds the sequence of feature vectors, one by one, and learns the change in facial features from one frame to the next. BI-LSTM (a type of recurrent neural network with gating mechanisms) can keep an internal memory of previous frames and thus detect temporal inconsistencies or smoothness in the face's appearance over time. The bi-directional configuration allows the sequence to be analyzed both in the forward and backward direction, utilizing the entire sequence context information for evaluation. The sequence of frames is distilled by the end of this LSTM stage into a single learned representation (the LSTM's final hidden state) that captures the salient temporal dynamics in the input frames. Finally, this representation is fed into a set of dense (fully connected) layers to complete the classification. Furthermore, the LSTM's output is fed into a dense layer with a rectified linear unit activation, which reduces dimensionality and extracts important nonlinear combinations of features. Additionally, a dropout regularization is applied to prevent overfitting. The last layer is a softmax classification layer with two output neurons that provide a probability distribution over the two classes: real or fake. For instance, if the probability for the "fake" class is high, the model will find patterns characteristic of a deepfake in the input sequence. This integrated approach makes the decision of the system jointly using both spatial features (from CNNs) and temporal features (from BI-LSTM). The CNN feature extractors and the LSTM-based sequence classifier are trained end to end as a unified model, where they minimize classification loss during training. In summary, the model can combine the components of the CNNs to flag subtle texture irregularities in a single frame and use the LSTM to check whether these irregularities persist or evolve strangely across forthcoming frames.

3.3 Proposed Model Architecture

Proposed hybrid architecture discussed in section 3.2 divides the entire task into distinct parts, and a clear role is defined for each components: the CNN branches are feature extractors of individual video frames, whereas the Bi LSTM acts as the temporal encoder,

capturing the dynamics of the sequence. However, this structured training makes sure that the respective network components focus on the problem of extracting spatial features as well as temporal modeling and combine the outputs from both of them into making the final decision.

In this section, the architecture of each implemented models, the reason for including it, and how it interacts with the other components will be thoroughly discussed.

3.3.1 EfficientNetV2B3

One of the CNN backbones that functions as a spatial attribute collector in the hybrid model is EfficientNetV2B3. EfficientNetV2B3 is of the EfficientNet family of models, which are well known to achieve an optimized accuracy-to-parameter ratio by scaling through the compound process. This implies that all scales of the network (depth, width, input resolution) are proportioned in a balanced way to effectively improve performance. The model consists of a series of convolutional blocks stacked together with modern design elements such as mobile inverted bottlenecks and squeeze-excitation attention. EfficientNetV2B3 uses convolutional kernels of varying sizes and applies with different strides in the network, allowing it to capture multi-scale features from the input image. For instance, early layers with small kernels are used for fine details such as edges or micro textures on the face, while deeper layers with larger receptive fields learn broader facial structures and context. Detecting deepfake anomalies can happen from a pixel-level color inconsistency to a slightly off facial structure, which makes this hierarchical feature learning well suited for detecting such deepfake anomalies.

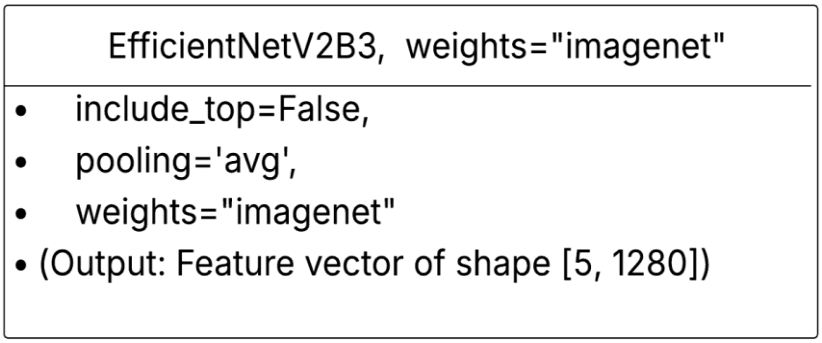


Figure 10. EfficientNetV2B3 architecture.

The above-mentioned Figure 10 portrays the architecture of EfficientNetV2B3. Using a pre-trained network on ImageNet (to make use of its learned visual patterns) and removing the top classification layer from the network (include_top=False) so that the global average pooling output remains as the feature representation. For example, when a 100×100 (RGB) face image is fed to EfficientNetV2B3, the network processes it through its layers and outputs a 1536-dimensional feature vector from the last pooling layer. It is a compact summary of the frame's visual content in terms of the facial features that the network's convolutional filters produce. During the training phase of this hybrid model, the EfficientNetV2B3 parameters are frozen (nontrainable) at first so that the network does not fine-tune the deepfake data but supplies stable high-level features. This way of proceeding prevents overfitting with the small size of used dataset and greatly reduces computational costs without compromising rich feature descriptors.

The outputs of other CNN branches are forwarded to the next stage (feature fusion and LSTM) together with the output of EfficientNetV2B3 for each frame. EfficientNetV2B3 leverages a state-of-the-art feature extractor that can capture a large variety of facial attributes efficiently. It enhances the spatial feature representation robustness with the other CNNs.

3.3.2 XceptionNet

The second CNN architecture integrated into the model as a parallel feature extractor is XceptionNet. Xception (Extreme-Inception) is a deep convolutional network that extends the Inception family of architectures by using depthwise separable convolutions. Spatial filtering and channel-wise combination are done as separate operations in a depthwise separable convolution, which considerably decreases the number of parameters by a large extent compared to a standard convolution but can still learn complex features. In addition, this model can capture fine-grained details and patterns of images. This feature extractor also includes a series of convolutional blocks, which use depthwise convolution, pointwise convolution, and residual connections for the gradient flow.

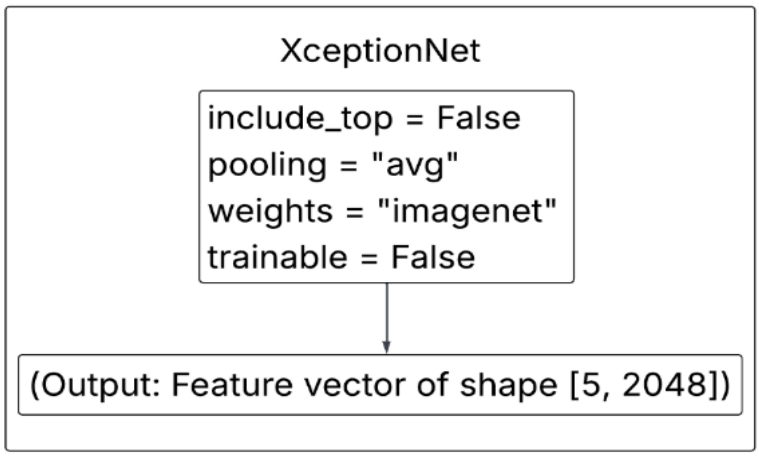


Figure 11. XceptionNet architecture.

As shown in Figure 11, the Xception model pre-trained on ImageNet is reduced after its final global average pooling layer to obtain feature vectors from each frame. For each 100×100 face image input, the model produces a 2048-dimensional feature. These are the features that Xception pays attention to through its depthwise convolutional filters. In the early phase of training, the pre-trained weights of the Xception branch are kept frozen. This enables the model to generate robust features without the reliance on a large quantity of new training data. The 2048-dimensional Xception feature vector is

concatenated with the features from other CNNs and passed to the Bi-LSTM during the fusion step.

3.3.3 VGG 16

The third convolutional neural network integrated into the hybrid model as a parallel feature extractor is VGG16. It is a simple, uniform layer design, consisting of multiple stacked 3×3 convolution layers separated by 2×2 max-pooling layers and followed by fully connected layers. VGG16 comes with a relatively large number of parameters and depth (16 layers, 13 of which are convolutional). However, simplicity and heavy use of small convolutional kernels enable it to pick up features incrementally, from simple edges to complex object parts, with very uniform receptive field progression. VGG16 has been broadly implemented in face recognition and verification tasks within the scope of face analysis, as it is effective at capturing facial characteristics.

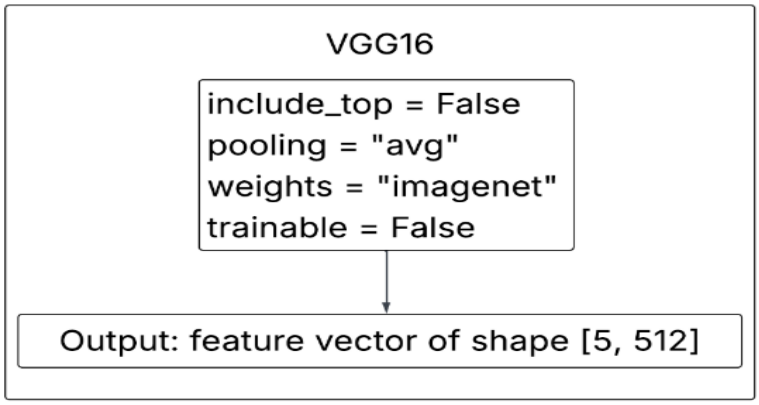


Figure 12. VGG16 Architecture.

In Figure 12, it is seen that the deepfake detection model adopts VGG16 trained on ImageNet before removing its classification layers to extract a feature vector from the final convolutional block after global average pooling. The visual content of every input frame generates a 512-dimensional feature vector from VGG16. The VGG16 architecture compresses information, resulting in a smaller 512-dimensional output compared to EfficientNet and Xception. VGG16 parameters in the hybrid model maintain their pre-training configuration during training, thus preserving general and reliable features

across limited data. The implementation of VGG16 serves to broaden the collection of features extracted by the system. VGG16 brings complementary information to the model because of its learned feature space and architectural design differ from EfficientNetV2B3 and XceptionNet. The filters within VGG16 detect facial cues and artifacts differently from the other two networks in the analysis. Each video frame receives a comprehensive 4096-D representation when the 512-D VGG16 feature is merged with the 1536-D EfficientNet and the 2048-D Xception features. Every frame receives a combined input, including multiple learned features that span VGG16 classical representations to Xception detailed textures and EfficientNet optimized features during the fusion process.

3.3.4 Bi-Directional LSTM (Recurrent Neural Network)

The temporal component of the proposed hybrid model is based on a Bidirectional Long Short-Term Memory (Bi-LSTM) network, a particular variant of recurrent neural network (RNN) that is explicitly intended to process sequences of data. The CNN modules (EfficientNetV2B3, XceptionNet, and VGG16) examine spatial properties of individual frame instances separately, whereas the Bi-LSTM component is able to learn temporal dependencies and dynamics among a sequence of such spatial features.

The reason behind adopting a Bi-LSTM is that it mitigates the vanishing gradient issue, which is generally presented in standard RNNs. The diminishing gradient signals during backpropagation through time can make traditional RNNs unable to learn from long-range sequential dependencies. On the contrary, the LSTM architecture makes use of gating mechanisms, input, forget, and output gates, that precisely regulate the transmission of information through memory cells and hence maintain stable gradients over longer sequences. The forget gate decides which parts of the last cell states should be kept or removed, the input gate selects which new information should enter the cell state, and the output gate decides how much of the cell state should be exposed as output.

This approach is based on the bidirectional configuration of two independent LSTM networks that process each input sequence at the same time but in opposite temporal directions. The forward LSTM treats the sequence chronologically (1 to 5), while the backward LSTM treats it in reverse (5 to 1). This dual-direction analysis enables the model to contextualize each frame using information from both past and future time steps as a sequence. As a result, temporal evaluation of this nature leads to the detection of minor inconsistencies that would not be perceptible from single-direction analysis.

The formal input to the Bi-LSTM module at each timestep t was a fused feature vector generated from concatenating the output of EfficientNetV2B3 (1536-dimensional), XceptionNet (2048-dimensional), and VGG16 (512-dimensional), resulting in the unified feature representation of size 4096. Therefore, a complete input sequence of 5 frames can be formulated as:

$$X = [x_1, x_2, x_3, x_4, x_5] \quad where\ each \quad x_t \in \mathbb{R}^{4096}, \tag{2}$$

Equation (2) allows the Bi-Directional LSTM to be used as an input in the architecture. It is considered a sequence of five feature vectors where each vector relates to the five-frame sequence. The vector, denoted as Xt (where t ranges from 1 to 5), consists of 4096 dimensions, which are obtained by concatenating the features of EfficientNetV2B3, XceptionNet, and VGG16. This structure of input helps the Bi LSTM to learn temporal dependencies between successive frames, which makes deepfake detection more effective.

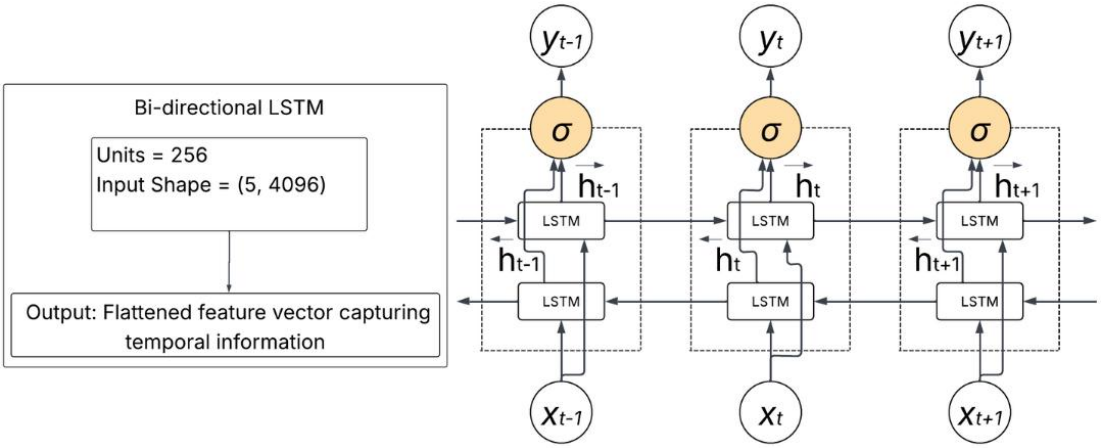


Figure 13. Overall detailed architecture of BI-LSTM.

The Bi-directional Long Short-Term Memory (Bi-LSTM) architecture consists of two distinct hidden layers: one facilitating a forward pass ($\overrightarrow{h_t}$) and the other a backward pass ($\overleftarrow{h_t}$). Each directional layer utilizes its own set of gates, namely, the input, forget, and output gates, and maintains independent memory states. The forward hidden state at time step t, denoted as $\overrightarrow{h_t}$, is calculated based on the input at the time step t and the forward hidden state from the previous time step, $t-1$. In contrast, the computation of the backward hidden state, $\overleftarrow{h_t}$, is derived from the input at time step t and the backward hidden state from the subsequent time step, t+1. Mathematically, the updates to the LSTM memory and the corresponding outputs at each time step can be represented using the following equations, adapted from (Y.-H. Li et al., 2020).

Forward direction ($\vec{h}_t$) :

$$\begin{gathered}\overrightarrow{i_t} = \sigma\left(W_i x_t + U_i \vec{h}_{t-1} + b_i\right) \\ \overrightarrow{f_t} = \sigma\left(W_f x_t + U_f \vec{h}_{t-1} + b_f\right) \\ \overrightarrow{o_t} = \sigma\left(W_o x_t + U_o \vec{h}_{t-1} + b_o\right) \\ \vec{c}_t = \vec{f}_t \odot \vec{c}_{t-1} + \vec{i}_t \odot tanh(W_c x_t + U_c \vec{h}_{t-1} + b_c) \\ \vec{h}_t = \vec{O}_t \odot tanh(\vec{c}_t)\end{gathered} \quad (3)$$

Backward direction ($\overleftarrow{h}_t$) :

$$\begin{aligned}
\overleftarrow{i}_t &= \sigma\left(W_i x_t + U_i \overleftarrow{h}_{t+1} + b_i\right) \\
\overleftarrow{f}_t &= \sigma\left(W_f x_t + U_f \overleftarrow{h}_{t+1} + b_f\right) \\
\overleftarrow{o}_t &= \sigma\left(W_o x_t + U_o \overleftarrow{h}_{t+1} + b_o\right) \qquad (4) \\
\overleftarrow{c}_t &= \overleftarrow{f}_t \odot \overleftarrow{c}_{t+1} + \overleftarrow{i}_t \odot tanh(W_c x_t + U_c \overleftarrow{h}_{t+1} + b_c) \\
\overleftarrow{h}_t &= \overleftarrow{o}_t \odot tanh(\overleftarrow{c}_t)
\end{aligned}$$

In these equations, the symbols W_i, W_f, W_o, and W_c, as well as U_i, U_f, U_o, and U_c, refers the learnable weight matrices, while b_i , b_f , b_o , and b_c refer to the bias vectors associated with the input, forget, output gates, and candidate memory states, respectively. The functions denoted by σ and tanh represent the sigmoid and hyperbolic tangent activation functions, while $\odot$ signifies element-wise multiplication.

Upon completing the computations for the forward and backward hidden states across all time steps, the outputs from the final time step of both directions ($\vec{h}_5$ and $\overleftarrow{h}_1$) are concatenated, yielding a comprehensive representation of the entire sequence :

$$h_{sequence} = \left[\vec{h}_5; \overleftarrow{h}_1\right] where \;\; h_{sequence} \in \mathbb{R}^{512} \qquad (5)$$

The temporal dynamics of inputs in both forward and backward time are summarized in a single hidden state referred to as $h_{sequence}$. It gives an informative embedding about the temporal consistency or anomalies in the sequence. The embedding is run through a fully connected dense layer with 128 neurons and a Rectified Linear Unit (ReLU) activation function to capture complex nonlinear relationships. After this dense layer, a dropout layer of rate 0.3 is appended to avoid overfitting by randomly making some neuron activations to zero. A final classification is made through a two-unit softmax layer, producing class probabilities reflecting if the input data sequence is genuine or manipulated.

3.3.5 Compete Hybrid Architecture

The hybrid deepfake video detection pipeline is presented in the form of a systematic evolution starting from raw input preprocessing up through the final prediction and output of explainable AI.

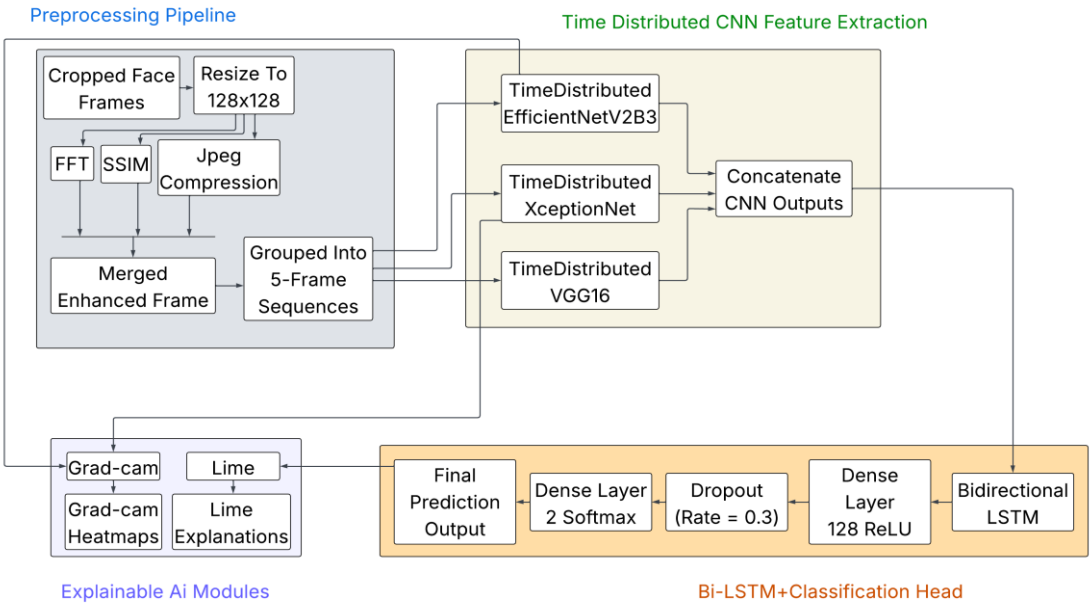


Figure 14. The complete architecture of the hybrid model.

In Figure 14, first, in the Preprocessing Pipeline (blue), cropped facial frames are resized to an image specification of 128 x 128 pixels. These frames are passed through three different enhancement paths, which aim to emphasize the manipulation regions from a multitude of view angles, the first of which is the FFT, the second is SSIM, and the last is JPEG compression simulation. The results of these operations are combined into an enhanced frame that is arranged into sequences of five frames for purposes of temporal processing.

Furthermore, in the Time-Distributed CNN Feature Extraction module, the grouped sequences are processed in parallel through a set of three backbones: TimeDistributed EfficientNetV2B3, XceptionNet, and VGG16. Each of these networks (pre-trained on

ImageNet and configured with include_top=False and pooling='avg') extracts different high-dimensional features from each frame. The feature vectors of dimensions [5, 1280], [5, 2048], and [5, 512] are the outputs of these networks. To create a unified spatial feature representation across the frame sequence, the temporal vectors are concatenated to direct this consolidated feature vector toward the BiLSTM and Classification. The Bidirectional LSTM layer is used to collect time-dependent patterns in forward and backward directions in the concatenated feature space to learn sequential change and context in dual directions. The temporal output is later refined by a dense layer with 128 units, ReLU activation, and a dropout layer with a rate of 0.3 to avoid overfitting. Lastly, this is traversed into a Dense layer, with two output neurons using softmax activation, to receive the classification of whether the video is authentic or fabricated.

In addition to that, the architecture includes Explainable AI Modules for the interpretation of the model's anticipated outcome. The mechanism of Grad-CAM generates heatmaps based on intermediate CNN activations, highlighting spatial regions of importance, and LIME uses local perturbation-based analysis of the model's decision with a rationale in terms of the interpretation for prediction.

4 Results and Analysis

This section details the performance and evaluation of the hybrid CNN + Bi-LSTM deepfake detection model by utilizing a comprehensive set of metrics and visualizations. Standard classification metrics are first reported, including overall accuracy, loss, precision, recall, F1-score, and Receiver Operating Characteristic Area Under the Curve (ROC-AUC). In addition, the confusion matrix and the distribution of confidence scores of the model for both classes are reviewed. Further, explainable-AI tools, including the Gradient-weighted Class Activation Mapping (GradCAM) and the Local Interpretable Model-Agnostic Explanations (LIME), are also implemented to offer some insight as to how the model makes decisions. These tools show which regions of an input frame the model considers when classifying the input as real or fake. At last, the hybrid model is used to annotate predictions on real and fake video frames with the model's confidence scores. The results show that the hybrid architecture can learn spatial and temporal features and achieve high detection accuracy and interpretable decision-making.

4.1 Test Accuracy and Loss

On the designated test set, the hybrid model achieved an impressive classification accuracy.

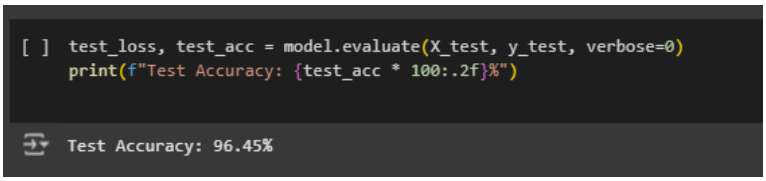


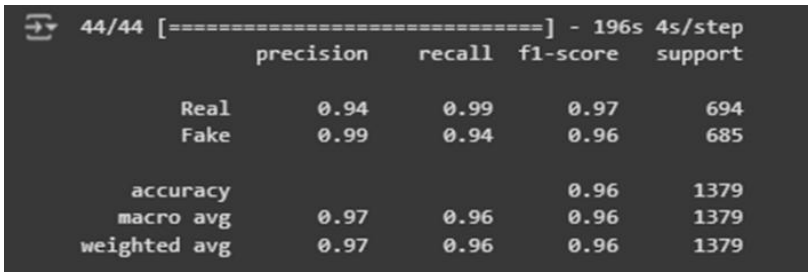
Figure 15. Test Accuracy.

The outcome from Figure 15 clearly states that the test accuracy was around 96.5%, which means that the model was capable of identifying whether a video was genuine or fake in almost all cases. The test loss was minimal, which provides further assurance of the model's predictions being very confident. With such a low binary cross entropy loss

value and a high accuracy, this indicates that the model can maintain performance on unseen datasets very well without overfitting. This outcome was probably due to the hybrid design of the model, which exploited both spatial and temporal cues.

4.2 Classification Report

A detailed classification report was produced to explain the performance of the model in each class (Real vs Fake).



```
44/44 [==============================] - 196s 4s/step
              precision    recall  f1-score   support

        Real       0.94      0.99      0.97       694
        Fake       0.99      0.94      0.96       685

    accuracy                           0.96      1379
   macro avg       0.97      0.96      0.96      1379
weighted avg       0.97      0.96      0.96      1379
```

Figure 16. Classification report.

The precision, recall, and F1 scores for both classes and overall accuracy are summarized in Figure 16. In the case of the fake class, the model obtained a precision of 0.99 (=99%) and 0.94 (=94%) precision for the real class, with a recall of 0.94 (fake) and 0.99 (real). These are excellent and balanced performances. High precision of the fake class means that if the model predicts fake, it is true in 99% of cases. The high recall for real (99%) implies that the model properly identifies nearly all the true videos, except for very few that the model mislabels as an actual video as fake. Recall for the fake class was high, at 94%, meaning a small fraction of deepfakes were missed (false negatives). Conversely, the precision for the real class is 94%, indicating that about 6% of videos predicted as 'real' were actually deepfakes that were falsely classified. Additionally, the F1 scores for both classes are almost equivalent, with very high values (around 0.96–0.97), which means that the model represents a good balance between precision and recall. Both weighted-average and macro-average metrics come out to be roughly 0.96 as well. These

results indicate that performance is similarly strong on the real and the fake classes regardless of any class imbalance in the test set.

4.3 Training vs Validation Accuracy and Loss

The model's accuracy grew rapidly during training and converged to a high value, while the validation accuracy closely followed the training accuracy.

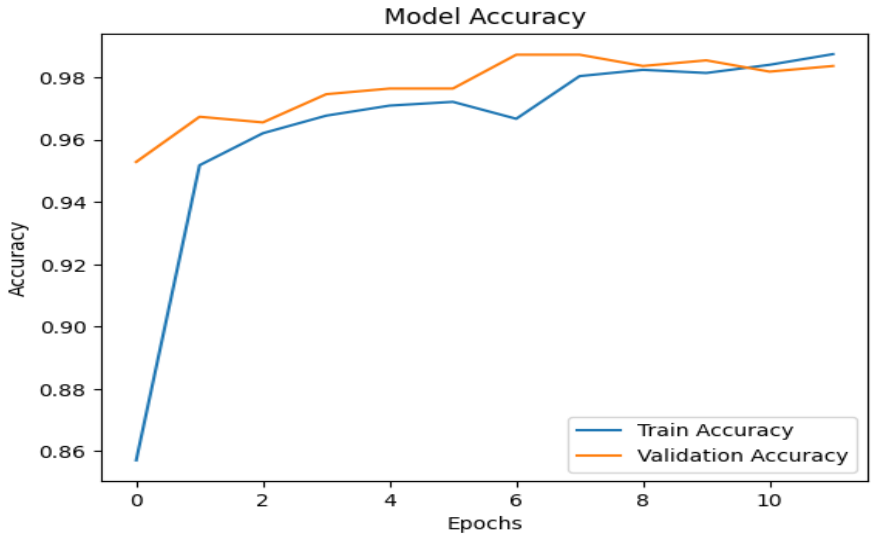


Figure 17. Training and Validation Accuracy.

The accuracy of both training (blue line) and validation (orange line) is shown in Figure 17 with each epoch. After the first epoch, training accuracy went from near zero to about 86%, and validation accuracy increased higher (about 90–95%). This indicates that the transfer learning (the CNN was probably initialized with pre-trained weights, being an EfficientNet-based CNN backbone) helped the model, and it was already extracting useful features. Training and validation accuracy continue to go up over the next few epochs. However, at epoch 3, training accuracy is around 95%, with validation accuracy also stuck in the mid-90s. Around 5 epochs, the curves essentially start plateauing around 97–99%. The validation accuracy is around 98% (reached at epoch 6–7), and the training accuracy is just above 98%, with validation accuracy also near 98% in the last epoch. More importantly, there is no noticeable gap opening between the training and

validation curve throughout the process. In particular, the validation accuracy slightly exceeds the training accuracy at some points (epochs 1, 2, and again around epoch 7). This indicates excellent generalization and implies that regularization strategies (dropouts in the CNN and maybe early stopping) were good. The curves converge steadily toward the same high number, indicating that the model has learned the deepfakes underlying pattern quite well and can apply it to unencountered data.

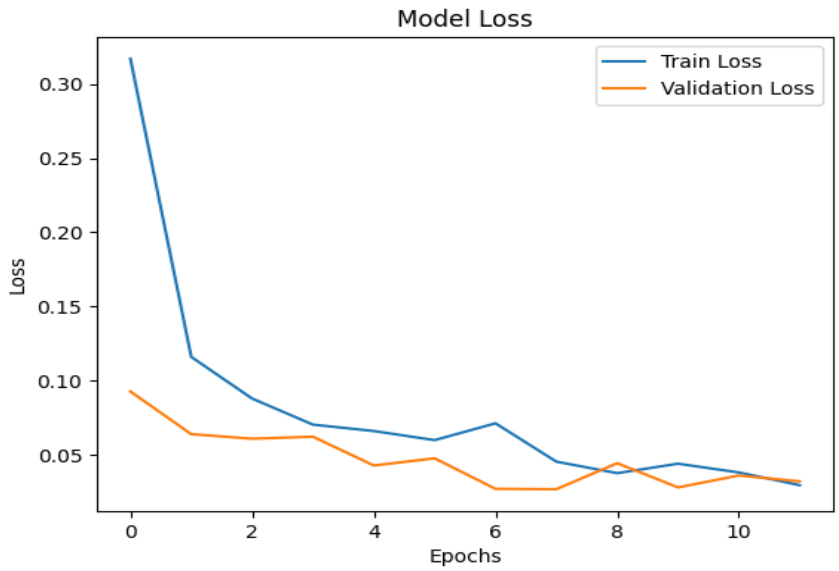


Figure 18. Training and validation loss.

Figure 18 shows the loss curves through the training process. The validation loss starts around 0.10 and also lowers down, while the training loss drops from around 0.33 in the first epoch to below 0.15. At epoch 2, the validation loss is 0.07–0.08, and the training loss is 0.10, which is a huge jump. The loss continues to fall for both as the training continues, but with decreasing returns as it nears a low asymptote. Moreover, the validation loss is very low (in the range of 0.05 to 0.03) after epoch 3. It even occasionally goes slightly below the training loss (for instance, the validation loss is slightly lower than the training loss at around epochs 7–8). Moreover, this is a good sign of no overfitting. With the last epochs (10–11), the training loss is roughly 0.02, and the validation loss as well is around 0.02–0.03, converged and essentially the same. This is a very low loss value, indicating that the model's predicted probability distributions are nearly identical

to the actual labels. Practically, the model gives 0.99 on the fake frame and 0.01 on the real frame, so it makes almost no error. Both curves smoothly decrease and converge to a low value, showing the effectiveness of the training. Overall, the training vs. validation accuracy and loss shows that the model taught the task of deepfake detection very well and settled for high accuracy with minimal overfit.

4.4 Roc Curve and Auc

The ROC (Receiver Operating Characteristic) curve has been plotted to further assess the model's discriminative ability.

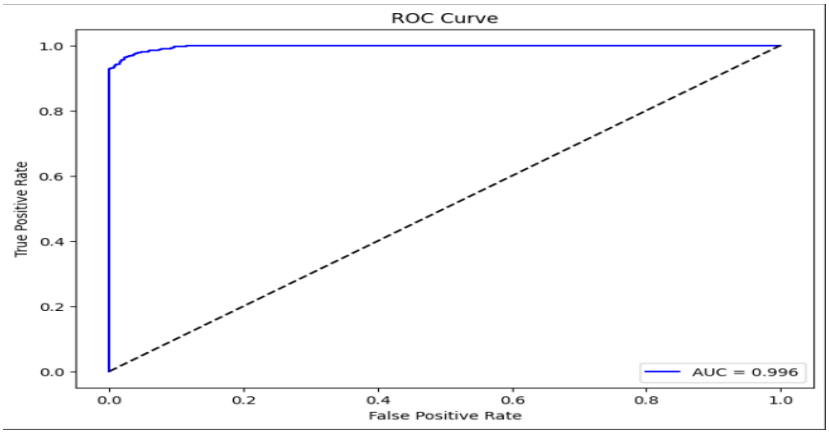


Figure 19. ROC curve for the model on the test set.

The blue curve in Figure 19 is the true positive rate (TPR) versus the false positive rate (FPR) relationship. The dashed line is the baseline for no discrimination. The hybrid model has an AUC of 0.996, which is very close to the perfect score of 1.0. The ROC curve rises steeply towards the upper left corner of the plot. The model quickly reaches a high true positive rate while maintaining very low false positive rates. The graph shows that the model is able to accurately identify a large number of deepfakes while generating very few false positives. With the high AUC, there is probably a threshold that simultaneously achieves TPR > 0.98 and FPR < 0.02, indicating the model's quality. This is due to the fact that the model produces highly differentiated probability outputs for the two classes, resulting in near-ideal performance of the ROC. AUC of 0.996 is very

close to 1.0, and it is likely that a few edge cases exist where the model is uncertain. Overall, this is a notable AUC and it shows that the model is very good at ranking videos by their likelihood of being fake.

4.5 Confusion Matrix

The confusion matrix of predicted vs actual classes for the test set is used to better understand the distribution of the model's errors. For clarity, the matrix is normalized and shown in percentages.

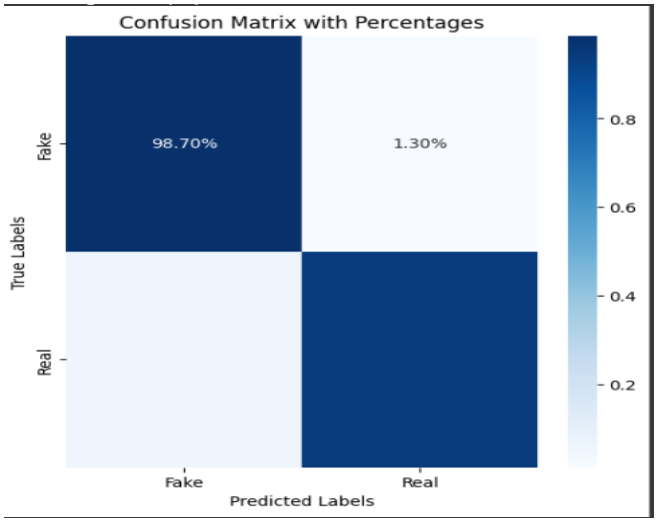


Figure 20. Confusion matrix on the test dataset.

As illustrated in Figure 20, about 98.7% of the deepfake frame sequences were correctly identified by the model, which corresponds to sequences that are actually fake and predicted as fake in the top left cell. Only 1.3% of the top right cell (actual fake, predicted real) shows that only a very small fraction of fakes were undetected (false negatives). Also, looking at the bottom right cell (actual real, predicted real), there is a high percentage (over 98%) that the vast majority of real frame sequences were correctly recognized as real. The bottom left cell (actual real, predicted fake) on the order of approximately 1–2%, which means false positives (real content misclassified as fake) are extremely low.

4.6 Confidence Distribution Scores and Model Predictions Scatter Plot

Understanding not only the accuracy but also the confidence of the decisions of any classification model is an important factor of evaluating the robustness and trustworthiness of a classification model. High confidence predictions are required for the reliable separation of manipulated from authentic video frames, for the purpose of minimizing false flags and missed detections, in the context of deepfake detection.

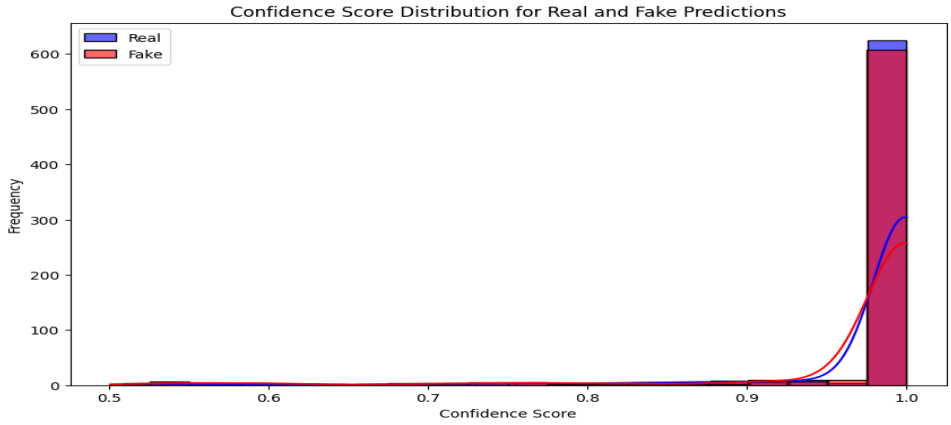


Figure 21. Distribution of the Model's Confidence Scores.

In Figure 21, the distribution of the confidence scores over all the test samples by class (real vs. fake) is illustrated. On the horizontal axis is the predicted confidence score (softmax probability), and on the vertical axis is frequency. The real (blue) and fake (red) distributions are concentrated in the 0.95–1.0 confidence range, indicating that most model predictions are made with a very high level of certainty. This implies the learned features by the model would suffice in separating authentic input frames from tampered input frames since there is minimal overlap between the two distributions.

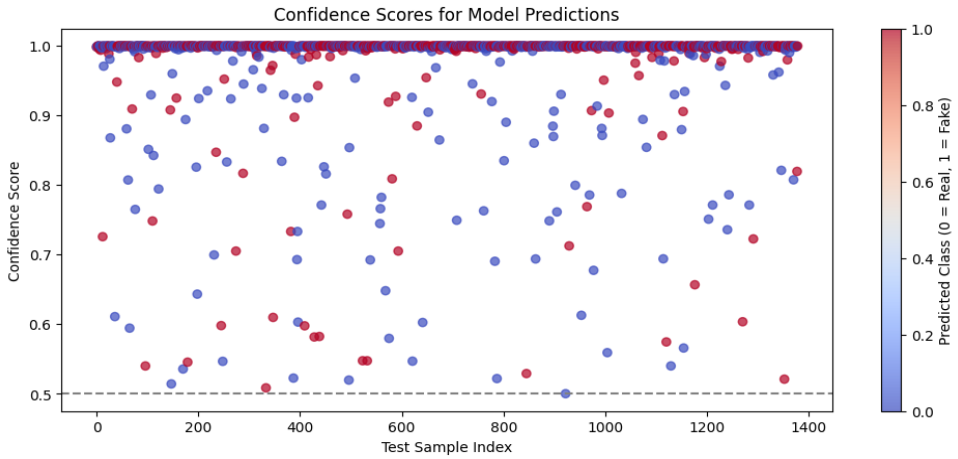


Figure 22. Scatter Plot of Prediction Confidence for Each Test Sample.

The scatter plot of prediction confidence in Figure 22 shows one dot for each test prediction. Confidence scores are on the vertical axis, and the test set's index is on the horizontal axis. Predicted fake samples are indicated by red dots, and predicted real samples by blue dots. The majority of the predictions are clustered very close to the maximum confidence value (1.0), illustrating the model's constant certainty of a wide variety of inputs. Very little ambiguity in the classification process is reflected by a few predictions that fall near the decision boundary (around 0.5–0.7).

Combined, these visualizations present an interpretable way of looking at the model's decision confidence. The learned representations are highly clustered in confidence values around 1.0 and provide sharply defined boundaries, which indicate the reliability of the model's learned representations.

4.7 Explainable AI Interpretation

Quantitative metrics help to understand the model's efficacy. However, the use of explainable AI techniques allows for a complete understanding of the model's decision-making process. On the CNN-BiLSTM model, Grad-CAM and Lime have been used to visualize the most important parts of an input frame that helped in classifying the input frame as fraudulent.

4.7.1 Grad-Cam Heatmap Interpretation

For the interpretability of the model's decisions, Gradient-weighted Class Activation Mapping (Grad-CAM) was applied to CNN based feature extractors end convolutional layers.

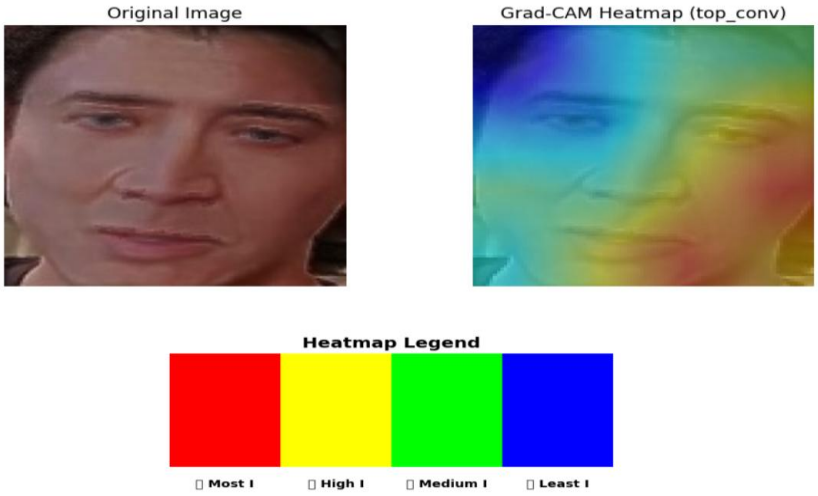


Figure 23. Grad-Cam Visualization of a sample frame.

A sample frame and Grad-CAM heatmap overlay of a deepfake video are shown in Figure 23, with the spatial regions that contributed significantly to the "fake" classification. However, the activation is highly concentrated in critical facial regions, such as the mouth, jawline, and eye contours, as seen in the heatmap. These areas are often linked to contradictions present in deepfakes, like faulty lip synchronization, odd movements in and around cheeks, as well as differences in skin texture around eyes. The model's focused attention is demonstrated through areas of higher activation, such as warmer areas (red and orange hues), and lower activation of background elements and non-facial areas. Therefore, this overlay is used as a validation platform as well as for traceability in cases where transparency is required.

4.7.2 Lime Explanation

To enrich the Grad-CAM based feature space heatmaps with an input space explanation that is generalized across multiple models, Local Interpretable Model Agnostic Explanation (LIME) is also incorporated.

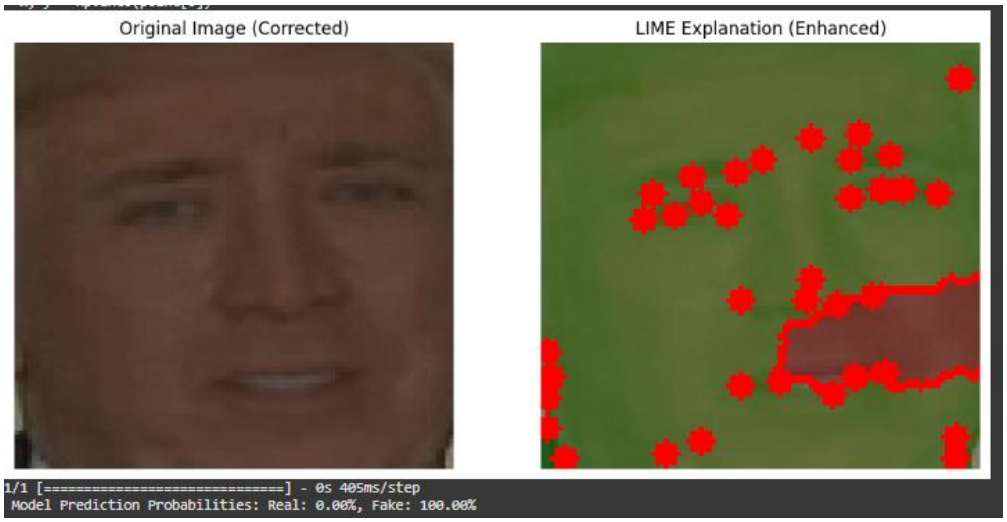


Figure 24. Lime explanation of the model's decision on a fake video frame.

Figure 24 shows that LIME's output highlights the image's superpixels, which help drive the model's decision about whether the image is fake. In this case, specifically, LIME identifies and emphasizes a number of superpixels around the mouth, chin, and lower mouth boundary. These areas correspond to these blending artifacts, misaligned expressions, or subtle distortions of fabricated content. All regions that the red overlays denote contribute positively to the model's prediction of the fake class. In that case, removing these regions will significantly influence prediction confidence.

LIME directly manipulates and analyzes the inputs. This distinction makes LIME particularly suitable for local decision explanation on an individual instance basis.
Aligning with the Grad-CAM explanation using LIME alleviates the model's trustworthiness.

4.8 Example Predictions on Real and Fake Videos

To test the saved model's performance in real-life detection scenarios, sample videos were uploaded to further test the process in a web app. These examples demonstrate how the hybrid modeling (CNN + Bi-LSTM) not only has high accuracy for frame classification but is also interpretable, fostering trust and transparency in the outputs.

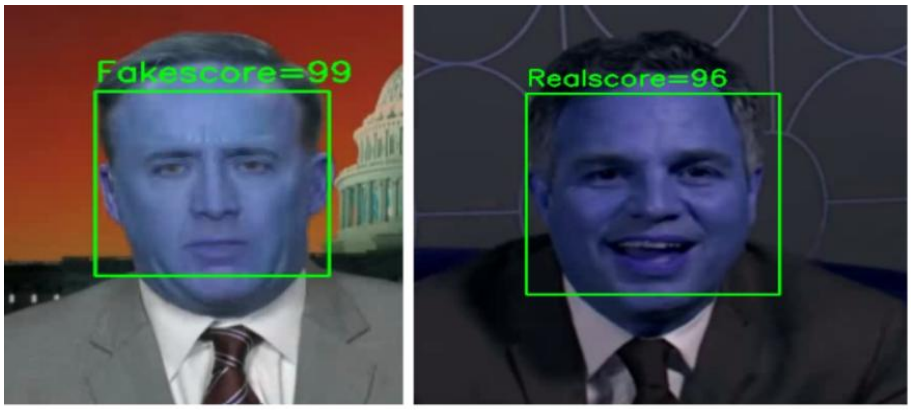


Figure 25. Example of the model output on a deepfake video.

Figure 25 shows two comparative example predictions a test deepfake video and a real video to further show the discriminative capability of the hybrid model. The CNN + Bi-LSTM pipeline has been applied to both videos, and each is then labeled with the model's classification label and the corresponding confidence score. The test results from videos are split into 10 frames, each showing the outcome. In one, the frame of a fake video is the left frame. With a score of 99%, the model confidently detects the face as fake. On the other hand, the right frame has a real, unaltered video with the model correctly scoring it as 96% real. The facial region used for the inference is inside the bounding box, and the relatively high certainty indicates no typical deepfake anomalies.

5 Discussion

In this chapter, the results of the study are fully examined and discussed in light of the three main research questions that were defined at the beginning of the thesis. The first part of the subsection justifies and investigates the efficacy of creating a hybrid deepfake video detection architecture consisting of multiple convolutional neural networks coupled with a single recurrent neural network. The second part of the subsection evaluates and justifies the selection of particular preprocessing methods compatible with the UADFV dataset and their contribution to increasing detection performance. Lastly, the third sub-section discusses the implications of using Explainable Artificial Intelligence (XAI) tools such as Grad-CAM and LIME as an improvement in explainability, trustworthiness, and qualitative evaluation of deepfake detection systems.

5.1 Hybrid CNN-RNN Model Development for Deepfake Detection

This study focused on developing a hybrid deepfake video detection architecture that can detect manipulated video content from genuine ones with good accuracy. The proposed model was the combination of multiple convolutional neural networks (CNNs), including EfficientNetV2B3, XceptionNet, and VGG16, with a single recurrent neural network, in particular Bidirectional Long Short Term Memory (Bi-LSTM). The choice of multiple CNN architectures is due to the uniqueness of the features each network extracts. The spatial representations provided by EfficientNetV2B3 were effective and robust, XceptionNet captured subtle manipulation artifacts with depthwise separable convolutions, and VGG16 was for hierarchical spatial feature extraction. Fusing these diverse spatial features and feeding into the BiLSTM allowed the model to detect inconsistencies that normally occur temporally between manipulated video sequences. Moreover, evaluation results strongly pointed to the effectiveness of this hybrid approach with very high accuracy (96.45%) and better Area Under the ROC Curve (AUC). These outcomes illustrate clear benefits over simpler (CNN or RNN) only models,

supporting spatial and temporal characteristics that should be modeled simultaneously for deepfake detection.

5.2 Effectiveness of Dataset-Specific Preprocessing Techniques

Different preprocessing techniques were chosen carefully to do well with the UADFV dataset's characteristics to optimize model performance and training efficiency. The dataset was moderate in size and varied in quality, so targeted preprocessing methods were required to expose the model to meaningful and robust features during training. Face detection and cropping were used to concentrate only on facial regions by removing background details that have no contribution to facial expression, and also reduce data complexity. Furthermore, a JPEG compression simulation was performed to account for the reality of videos getting compressed in the process of their online dissemination. In order to better enhance robustness, frequency domain preprocessing strategies such as Fast Fourier Transform (FFT) and Structural Similarity Index Measure (SSIM) augmentations were employed to find the subtle but important frequency and structural inconsistencies of deepfake manipulations in the frequency domain. Collectively, these preprocessing techniques were very effective in improving data quality and helping the model generalize from a few training examples.

5.3 Explainable AI Tools for Enhanced Interpretability

An important part of this research was assessing the use of Explainable AI (XAI) tools such as Gradient-weighted Class Activation Mapping (Grad-CAM) and Local Interpretable Model-Agnostic Explanations (LIME) to improve the interpretability, evaluation, and end-user trust of deepfake detection systems. The Grad-CAM visualizations offered a clear diagnosis of where the model made spatial decisions, indicating the facial regions that were most salient for the model's decision pipeline. In addition, these findings were complemented by LIME, which performed localized perturbations of the input data, explicitly showing the image segments that make a difference for accurate predictions.

These interpretability tools, when integrated, greatly enhanced the model's transparency, showing logical and consistent decision making, which made sense to human intuition as well as expert knowledge on facial forgery indicators. The amount of interpretability present was not only something that increased certainty in the model's predictions, but also provided valuable qualitative assessments of what the model was doing above and beyond quantitative metrics (accuracy, or AUC).

6 Conclusion

6.1 Summary of Research Objectives and Contributions

This research was a direct response to the growing concern about deepfake content, the issue of information authenticity, and the digital trust it has presented. The main goal was to develop a deepfake video detection model capable of producing reliable detection that is also transparent and explainable. In approaching this objective, a hybrid deep learning framework was designed and implemented, which integrates both spatial and temporal analysis components. In addition to the technical formulation, this research is important for building an intelligent system that is not only correct but also feasible and interpretable in real applications. The model integrates explainable AI methods in the detection pipeline, hence detecting, classifying, and showing the rationale for predictions. This then leads to more trusting users, and the model becomes an important tool in applications such as cyber forensics, content verification, and platform moderation.

This work contributes meaningfully to the field through its design, evaluation, and interpretation, showing that balancing performance and interpretability is important for generative design. The findings of this study validate that deep learning models can be supported and accountable, indeed providing both solutions and understanding. This study serves as an endeavor to encourage a more responsible approach to fighting the wrong use of synthetic media and as a starting point for research on robust and ethically driven AI systems.

6.2 Limitations

Although promising results were obtained in this research, there are also several limitations. The first concern is that the model is dependent on the UADFV dataset, which has certain representativeness and size limitations. As a result, it may hinder the

generalizability of the findings to other smaller sets or other deepfake generation methods. Moreover, the hybrid architecture has inherent computational complexity that may be difficult to implement for a real-time interactive system or for resource-constrained environments. The study also did not look extensively into potential vulnerabilities to adversarial manipulations designed to bypass detection systems. This is a relevant topic nowadays in cybersecurity.

6.3 Recommendation for Future Study

Several future directions are proposed to address the identified limitations, as well as to increase the impact of this research. An essential next step is to diagnose the model on expanded and more varied datasets like Celeb-DF, FaceForensics++, and DFDC. These datasets provide the necessary variety of manipulation techniques and real-world variability necessary to validate the competence of the model to generalize to conditions not present in the UADFV dataset.

Secondly, optimized, lightweight model variants are also explored. Deployment of the current hybrid architecture on devices with limited processing power, as well as its use in real-time environments is prevented by the high computational demands. This may possibly be done by using model compression techniques, like pruning, quantization or knowledge distillation, with little accuracy loss.

Another promising direction is integrating adversarial training methodologies for making the model more robust to deliberate attempts to bypass detection. With adversarial attacks against detection systems becoming more sophisticated, it is important to reinforce the model's security by giving exposure to such threats during training to ensure long term reliability.

In addition, extending the model to allow for multimodal analysis and including audio features along with the visual cues can profoundly strengthen the accuracy of detection. Moreover, the appearance of synchronized voice modifications in Deepfake content

makes the task of detecting the visual-spatial inconsistencies faced with the audio anomalies more extensive and more reliable.

Consequently, these future developments strive to enhance the technical performance of the model, while also strengthening its practical applicability under dynamic, high-stakes scenarios. This study takes a meaningful step towards building such reliable, interpretable, and deployable deepfake detection systems that are becoming ever more important to protect the ethicality of digital information in an era of artificial content.

References

Abdelkhalki, J. E., Ahmed, M. B., & Abdelhakim, A. (2022). DEEPFAKE DETECTION BASED ON THE XCEPTION. *Journal of Theoretical and Applied Information Technology*, *100*(1), 221–234. https://www.jatit.org/volumes/Vol100No1/19Vol100No1.pdf

Adnan, S. R., & Abdulbaqi, H. A. (2022). Deepfake Video Detection Based on Convolutional Neural Networks. *IEEE International Conference on Data Science and Intelligent Computing (ICDSIC)*, 65–69. https://doi.org/10.1109/ICDSIC56987.2022.10075830

Ajoy, A., Mahindrakar, C. U., Gowrish, D., & A, V. (2021). DeepFake Detection using a frame based approach involving CNN. *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, 1329–1333. https://doi.org/10.1109/ICIRCA51532.2021.9544734

Bouter, M. de L. den, Pardo, J. L., Geradts, Z., & Worring, M. (2023). *ProtoExplorer: Interpretable Forensic Analysis of Deepfake Videos using Prototype Exploration and Refinement* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2309.11155

Broinowski, A., & Martin, F. R. (2024). Beyond the deepfake problem: Benefits, risks and regulation of generative ai screen technologies. *Media International Australia*, 1329878X241288034. https://doi.org/10.1177/1329878X241288034

Caldelli, R., Galteri, L., Amerini, I., & Del Bimbo, A. (2021). Optical Flow based CNN for detection of unlearnt deepfake manipulations. *Pattern Recognition Letters*, *146*, 31–37. https://doi.org/10.1016/j.patrec.2021.03.005

Chauhan, P. S. (2024). Deepfake: Risks and Opportunities. *Computer*, *57*(6), 141–144. https://doi.org/10.1109/MC.2024.3392992

Das, A., & Sebastian, L. (2023). A Comparative Analysis and Study of a Fast Parallel CNN Based Deepfake Video Detection Model with Feature Selection (FPC-DFM). *2023 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA)*, 1–9. https://doi.org/10.1109/ACCTHPA57160.2023.10083340

Das, R., Negi, G., & Smeaton, A. F. (2021). *Detecting Deepfake Videos Using Euler Video Magnification*. https://doi.org/10.48550/ARXIV.2101.11563

Fahad, M., Zhang, T., Iqbal, Y., Ikram, A., Siddiqui, F., Abdullah, B. Y., Muhammad Nauman, M., Zhao, X., & Geng, Y. (2025). Advanced deepfake detection with enhanced Resnet-18 and multilayer CNN max pooling. *The Visual Computer*, *41*(5), 3473–3486. https://doi.org/10.1007/s00371-024-03613-x

Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., & Holz, T. (2020). *Leveraging Frequency Analysis for Deep Fake Image Recognition* (Version 3). arXiv. https://doi.org/10.48550/ARXIV.2003.08685

Gong, L. Y., Li, X. J., & Chong, P. H. J. (2024). Swin-Fake: A Consistency Learning Transformer-Based Deepfake Video Detector. *Electronics*, *13*(15), 3045. https://doi.org/10.3390/electronics13153045

Gowrisankar, B., & Thing, V. L. L. (2024). An adversarial attack approach for eXplainable AI evaluation on deepfake detection models. *Computers & Security*, *139*, 103684. https://doi.org/10.1016/j.cose.2023.103684

Guarnera, L., Giudice, O., Guarnera, F., Ortis, A., Puglisi, G., Paratore, A., Bui, L. M. Q., Fontani, M., Coccomini, D. A., Caldelli, R., Falchi, F., Gennaro, C., Messina, N., Amato, G., Perelli, G., Concas, S., Cuccu, C., Orrù, G., Marcialis, G. L., & Battiato, S. (2022). The Face Deepfake Detection Challenge. *Journal of Imaging*, *8*(10), 263. https://doi.org/10.3390/jimaging8100263

Helode, A., Yadav, A., Verma, V. P., & Srinivasa, K. G. (2024). Fusion of Machine Learning and Deep Learning:A Hybrid Approach for Deepfake Detection. *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1–6. https://doi.org/10.1109/ICCCNT61001.2024.10724874

Heo, Y.-J., Yeo, W.-H., & Kim, B.-G. (2023). DeepFake detection algorithm based on improved vision transformer. *Applied Intelligence*, *53*(7), 7512–7527. https://doi.org/10.1007/s10489-022-03867-9

Iglesias, G., Talavera, E., & Díaz-Álvarez, A. (2023). A survey on GANs for computer vision: Recent research, analysis and taxonomy. *Computer Science Review*, *48*, 100553. https://doi.org/10.1016/j.cosrev.2023.100553

Jiang, Y., Huang, Z., Pan, X., Loy, C. C., & Liu, Z. (2021). Talk-to-Edit: Fine-Grained Facial Editing via Dialog. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 13779–13788. https://doi.org/10.1109/ICCV48922.2021.01354

Jolly, V., Telrandhe, M., Kasat, A., Shitole, A., & Gawande, K. (2022). CNN based Deep Learning model for Deepfake Detection. *2022 2nd Asian Conference on Innovation in Technology (ASIANCON)*, 1–5. https://doi.org/10.1109/ASIANCON55314.2022.9908862

K, J., & M, A. (2024a). Safeguarding media integrity: A hybrid optimized deep feature fusion based deepfake detection in videos. *Computers & Security*, *142*, 103860. https://doi.org/10.1016/j.cose.2024.103860

K, J., & M, A. (2024b). Safeguarding media integrity: A hybrid optimized deep feature fusion based deepfake detection in videos. *Computers & Security*, *142*, 103860. https://doi.org/10.1016/j.cose.2024.103860

Kaddar, B., Fezza, S. A., Akhtar, Z., Hamidouche, W., Hadid, A., & Serra-Sagristá, J. (2024). Deepfake Detection Using Spatiotemporal Transformer. *ACM Transactions on Multimedia Computing, Communications, and Applications*, *20*(11), 1–21. https://doi.org/10.1145/3643030

Kaddar, B., Fezza, S. A., Hamidouche, W., Akhtar, Z., & Hadid, A. (2021). HCiT: Deepfake Video Detection Using a Hybrid Model of CNN features and Vision Transformer. *2021 International Conference on Visual Communications and Image Processing (VCIP)*, 1–5. https://doi.org/10.1109/VCIP53242.2021.9675402

Katarya, R., & Lal, A. (2020). A Study on Combating Emerging Threat of Deepfake Weaponization. *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 485–490. https://doi.org/10.1109/I-SMAC49090.2020.9243588

Khan, S. (2024). *Adversarially Robust Deepfake Detection via Adversarial Feature Similarity Learning*. https://doi.org/10.48550/ARXIV.2403.08806

Khan, S. A., & Dang-Nguyen, D.-T. (2022). Hybrid Transformer Network for Deepfake Detection. *International Conference on Content-Based Multimedia Indexing*, 8–14. https://doi.org/10.1145/3549555.3549588

Kohli, A., & Gupta, A. (2021). Detecting DeepFake, FaceSwap and Face2Face facial forgeries using frequency CNN. *Multimedia Tools and Applications*, *80*(12), 18461–18478. https://doi.org/10.1007/s11042-020-10420-8

Li, Y., Chang, M.-C., & Lyu, S. (2018). *In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking* (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1806.02877

Li, Y., & Lyu, S. (2018). *Exposing DeepFake Videos By Detecting Face Warping Artifacts* (Version 3). arXiv. https://doi.org/10.48550/ARXIV.1811.00656

Li, Y.-H., Harfiya, L. N., Purwandari, K., & Lin, Y.-D. (2020). Real-Time Cuffless Continuous Blood Pressure Estimation Using Deep Learning Model. *Sensors*, *20*(19), 5606. https://doi.org/10.3390/s20195606

Liu, J., Zhu, K., Lu, W., Luo, X., & Zhao, X. (2021). A lightweight 3D convolutional neural network for deepfake detection. *International Journal of Intelligent Systems*, *36*(9), 4990–5004. https://doi.org/10.1002/int.22499

Maheshwari, R. U., & Paulchamy, B. (2024). Securing online integrity: A hybrid approach to deepfake detection and removal using Explainable AI and Adversarial Robustness Training. *Automatika*, *65*(4), 1517–1532. https://doi.org/10.1080/00051144.2024.2400640

Maniyal, V., & Kumar, V. (2024). Unveiling the Deepfake Dilemma: Framework, Classification, and Future Trajectories. *IT Professional*, *26*(2), 32–38. https://doi.org/10.1109/MITP.2024.3369948

Mansoor, N., & Iliev, A. I. (2025). Explainable AI for DeepFake Detection. *Applied Sciences*, *15*(2), 725. https://doi.org/10.3390/app15020725

Mirsky, Y., & Lee, W. (2022). The Creation and Detection of Deepfakes: A Survey. *ACM Computing Surveys*, *54*(1), 1–41. https://doi.org/10.1145/3425780

Nirkin, Y., Wolf, L., Keller, Y., & Hassner, T. (2022). DeepFake Detection Based on Discrepancies Between Faces and Their Context. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(10), 6111–6121. https://doi.org/10.1109/TPAMI.2021.3093446

Pandey, R., & Kushwaha, A. K. S. (2024). Hybrid Deep-Learning Model for Deepfake Detection in Video using Transfer Learning Approach. *National Academy Science Letters*. https://doi.org/10.1007/s40009-024-01480-7

Pang, G., Zhang, B., Teng, Z., Qi, Z., & Fan, J. (2023). MRE-Net: Multi-Rate Excitation Network for Deepfake Video Detection. *IEEE Transactions on Circuits and Systems for Video Technology*, *33*(8), 3663–3676. https://doi.org/10.1109/TCSVT.2023.3239607

Pei, G., Zhang, J., Hu, M., Zhang, Z., Wang, C., Wu, Y., Zhai, G., Yang, J., Shen, C., & Tao, D. (2024). *Deepfake Generation and Detection: A Benchmark and Survey* (Version 4). arXiv. https://doi.org/10.48550/ARXIV.2403.17881

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

Rafique, R., Gantassi, R., Amin, R., Frnda, J., Mustapha, A., & Alshehri, A. H. (2023). Deep fake detection and classification using error-level analysis and deep learning. *Scientific Reports*, *13*(1), 7422. https://doi.org/10.1038/s41598-023-34629-3

Remya Revi, K., Vidya, K. R., & Wilscy, M. (2021). Detection of Deepfake Images Created Using Generative Adversarial Networks: A Review. In M. Palesi, L. Trajkovic, J. Jayakumari, & J. Jose (Eds.), *Second International Conference on Networks and Advances in Computational Technologies* (pp. 25–35). Springer International Publishing. https://doi.org/10.1007/978-3-030-49500-8_3

Salvi, D., Liu, H., Mandelli, S., Bestagini, P., Zhou, W., Zhang, W., & Tubaro, S. (2023). A Robust Approach to Multimodal Deepfake Detection. *Journal of Imaging*, *9*(6), 122. https://doi.org/10.3390/jimaging9060122

Seow, J. W., Lim, M. K., Phan, R. C. W., & Liu, J. K. (2022). A comprehensive overview of Deepfake: Generation, detection, datasets, and opportunities. *Neurocomputing*, *513*, 351–371. https://doi.org/10.1016/j.neucom.2022.09.135

Siddiqui, F., Yang, J., Xiao, S., & Fahad, M. (2025). Enhanced deepfake detection with DenseNet and Cross-ViT. *Expert Systems with Applications*, *267*, 126150. https://doi.org/10.1016/j.eswa.2024.126150

Singh, D., Singh, P., & Bhandari, R. (2024). Enhancing Deepfake Video Detection: A Hybrid CNN-LSTM Approach. *2024 First International Conference on Technological Innovations and Advance Computing (TIACOMP)*, 130–135. https://doi.org/10.1109/TIACOMP64125.2024.00031

Solaiman, M., & Rana, M. S. (2024). Enhancing Global Security: A Robust CNN Model for Deepfake Video Detection. *2024 7th International Conference on Information and Computer Technologies (ICICT)*, 238–243. https://doi.org/10.1109/ICICT62343.2024.00044

Sudarshana, K., & Vamsidhar, Y. (2025). UAM-NET: Robust Deepfake Detection Through Hybrid Attention Into Scalable Convolutional Network. *Expert Systems*, *42*(3), e70009. https://doi.org/10.1111/exsy.70009

Suratkar, S., Johnson, E., Variyambat, K., Panchal, M., & Kazi, F. (2020). Employing Transfer-Learning based CNN architectures to Enhance the Generalizability of Deepfake Detection. *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–9. https://doi.org/10.1109/ICCCNT49239.2020.9225400

Tran, V.-N., Lee, S.-H., Le, H.-S., & Kwon, K.-R. (2021). High Performance DeepFake Video Detection on CNN-Based with Attention Target-Specific Regions and Manual Distillation Extraction. *Applied Sciences*, *11*(16), 7678. https://doi.org/10.3390/app11167678

Trinh, L., Tsang, M., Rambhatla, S., & Liu, Y. (2020). *Interpretable and Trustworthy Deepfake Detection via Dynamic Prototypes* (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2006.15473

Tsigos, K., Apostolidis, E., Baxevanakis, S., Papadopoulos, S., & Mezaris, V. (2024). *Towards Quantitative Evaluation of Explainable AI Methods for Deepfake Detection* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2404.18649

Twomey, J., Ching, D., Aylett, M. P., Quayle, M., Linehan, C., & Murphy, G. (2023). Do deepfake videos undermine our epistemic trust? A thematic analysis of tweets that discuss deepfakes in the Russian invasion of Ukraine. *PLOS ONE*, *18*(10), e0291668. https://doi.org/10.1371/journal.pone.0291668

Venkateswarulu, S., & Srinagesh, A. (2024). DeepExplain: Enhancing DeepFake Detection Through Transparent and Explainable AI model. *Informatica*, *48*(8). https://doi.org/10.31449/inf.v48i8.5792

Wolter, M., Blanke, F., Heese, R., & Garcke, J. (2022). Wavelet-packets for deepfake image analysis and detection. *Machine Learning*, *111*(11), 4295–4327. https://doi.org/10.1007/s10994-022-06225-5

Yan, J., Li, Z., He, Z., & Fu, Z. (2025). *Generalizable Deepfake Detection via Effective Local-Global Feature Extraction* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2501.15253

Yang, X., Li, Y., & Lyu, S. (2019). Exposing Deep Fakes Using Inconsistent Head Poses. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8261–8265. https://doi.org/10.1109/ICASSP.2019.8683164

Yu, Y., Liu, X., Ni, R., Yang, S., Zhao, Y., & Kot, A. C. (2024). PVASS-MDD: Predictive Visual-Audio Alignment Self-Supervision for Multimodal Deepfake Detection. *IEEE Transactions on Circuits and Systems for Video Technology*, *34*(8), 6926–6936. https://doi.org/10.1109/TCSVT.2023.3309899

Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., & Yu, N. (2021). *Multi-attentional Deepfake Detection* (Version 3). arXiv. https://doi.org/10.48550/ARXIV.2103.02406