ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2026.0579

PERFORMANCE ANALYSIS OF DEEPFAKE VIDEO DETECTION USING DEEP LEARNING MODEL

K. Thulasimani and J. Pooja
Department of Computer Science and Engineering, Government College of Engineering, Tirunelveli, India

Abstract

Now a days, due to advances in computer editing tools, the creation and manipulation of fake content of images, videos, and audio has become quickly and easily accessible. Deepfakes are synthetic media created using Artificial Intelligence (AI), often with advanced deep learning models, that depict events or individuals that do not exist or did not occur in reality. Typically, deepfakes involve face swapping or full face generation, which creates highly realistic but fabricated content. While such technology has creative and entertainment applications, its misuse has raised serious concerns, including the spread of misinformation, manipulation of digital evidence, and identity-related crimes. The widespread use of masks after the COVID-19 pandemic has further complicated detection, as hidden facial features make it difficult to distinguish real from altered content. Traditional detection methods often struggle with masked or morphed faces, highlighting the need for more robust and general solutions. Addressing these challenges, a deep learning detection framework based on ResNet152V2 is introduced. The system was trained and tested on a combined UADFV and DFFMD dataset consisting of masked and unmasked models. The ResNet152V2 architecture uses residual connections and block normalization to improve learning efficiency, feature representation, and classification accuracy. Experimental findings confirmed the effectiveness of the proposed system, interms of training accuracy is achieved 0.9847 confirming its ability to effectively learn discrimination features. Validation results showed strong generalization to unseen frames, even in cases of hidden or partially covered faces.

Keywords:

Deepfake Detection, Artificial Intelligence (AI), ResNet152V2, Masked Faces, Image and Video Manipulation

1. INTRODUCTION

For decades, people have been intrigued by modifying photographic, audio, and video content. In the beginning, videos were edited frame by frame to create special effects in films, and different creative methods were used to manipulate content. Although photo editing has been available for a long time through tools like Adobe Photoshop, video editing also became easier with software such as Adobe After Effects allows for the alteration of video clips and the crafting of inventive visual tricks. Today, with the rise of personal computing, basic video editing is within reach of the general public. Advances in Artificial Intelligence (AI) and Machine Learning (ML) have led to the development of more advanced methods for editing photos, audio, and video content.

Deepfake technology has emerged as one of the most notable in this field. It initially gained prominence through Face Swap ML techniques, which were frequently misused to overlay celebrity faces onto explicit videos. With the development of advanced editing tools, the tasks of producing and modifying media content have become faster and more intuitive. In today's digital era, with platforms such as Twitter, With Facebook, Instagram, and WhatsApp functioning as major sources of information, deepfakes pose a significant threat by spreading misinformation, manipulating evidence, and influencing public opinion. Deepfakes are no longer limited to celebrities and public figures; they can target any individual, especially since many people post photos and videos online. One infamous example is a manipulated video of President Obama, created to showcase how deepfakes can make individuals appear to say things they never did. Deepfakes are a form of synthetic media generated using AI, typically through Generative Adversarial Networks (GANs). These models learn from massive datasets to produce highly realistic but entirely fabricated content. This involves changing facial expressions, vocal tones, gestures, or entire video segments to give them an authentic appearance. Deepfake generation relies on two core components: a generator, which creates fake content, and a discriminator, which evaluates the content's realism. Popular open-source tools for creating deepfakes include FaceSwap, DeepFaceLab, and DFaker.

Fig.1. Deepfake Sample (Source : SpringerLink)

In the Star-GAN [11] framework, image translation is extended to multiple domains, allowing a model to learn how to convert images from one source category into various target categories using a one-to-many mapping. By adopting this strategy, StarGAN enables image translation across multiple domains in a seamless manner, eliminating the need for explicitly paired datasets for each transformation. Star-GAN enables the model to transform images by changing multiple facial features, such as hairstyle and eye color, while also adapting to shifts in the background or environment. within the same framework.

The ProGAN [13] Deepfake method utilizes a progressive growth strategy throughout the training process. Both the generator and discriminator are first trained at a coarse resolution of 4×4 pixels in this method. In this approach, both the generator and discriminator networks grow progressively by adding new layers and resolution during training. This incremental growth allows for a smoother and more stable learning process.

The StyleGAN [8] framework represents a powerful approach to generative modeling, enabling the creation of high-quality, lifelike images. It represents an improvement over traditional GANs, offering enhanced control and superior quality in the generated images. One of the main innovations of StyleGAN is its ability to distinguish Between various aspects of image

generation, including high-level features (e.g., pose, identity, semantic information) and low-level details (e.g., texture and style). This separation enables more precise control over image generation, allowing processing at multiple levels without compromising image quality. The First Order Motion Model for Image Animation [9] adopts a novel approach that utilizes motion data from a source video to animate a target image. This operates by capturing motion features from both the source video and the target image. The extracted features are then used to construct a motion-driven representation of the target image, producing a realistic animation effect. This process transfers facial expressions, movements, and gestures from the video to the image, leading to natural and seamless animation.

Since the COVID-19 pandemic, video conferencing applications have become more popular. Attackers have utilized Deepfake models to generate fake virtual identities during online video conferences. During the COVID-19 pandemic, attackers employed video manipulation techniques to evade detection and erase evidence of their crimes from recorded videos. Moreover, the widespread use of face masks has greatly facilitated the creation of Deepfakes while making their detection more difficult. This is because of only the forehead, and eyes are visible, while other facial features remain hidden. This prompted criminals to wear facemasks to conceal their actions, manipulating surveillance footage to evade identification and bypass the criminal justice system. As a result, there was a growing concern about the misuse of Deepfake technology in spreading misinformation and manipulating people's opinions, identity theft, and the potential for harm in various fields, including politics, business, and entertainment. Identifying artificially generated and fake media content has become a more difficult challenge today.

The First Order Motion Model animates a target image by transferring motion from a source video, enabling realistic face animation from a single image, though with limited detail retention. Since the COVID-19 pandemic, video conferencing apps have become more common, and attackers have used deepfake models to create fake virtual identities. Additionally, the widespread use of face masks has made deepfake detection more difficult because only limited facial regions (like the eyes and forehead) are visible. This has been misused by criminals, who wear masks to obscure their identities and edit surveillance footage to avoid being recognized.

The challenges are addressed by a deepfake detection system specifically designed to detect masked-face deepfakes in videos. This system utilizes the both UADFV and DFFMD dataset which includes both real and manipulated video samples. The videos are first converted into individual frames, and facial regions are extracted and preprocessed (e.g., resized) to match model input requirements. The extracted frames are subsequently processed through deep Convolutional Neural Networks (CNNs) utilizing the ResNet50V2 and ResNet152V2 architectures. This approach focuses on feature extraction and classification using deep CNN models. The system is built to enhance deepfake detection, even in difficult situations where only partial facial features are observable.

2. RELATED WORK

Combining CNN with RNN, Eye-Blink Detection, and Grayscale Histogram analysis methods have been proposed for detecting Deepfakes [7]. In the first approach, image sequences of size 256 × 256 pixels were passed through a 12-layer CNN, followed by a 4-layer LSTM. This base model was applied directly to raw video data without any preprocessing. OpenCV was used to identify facial landmarks and track eye blinks, with the Eye Aspect Ratio (EAR) calculated to measure blink duration. A script was created to count blinks in the training videos, and this data was used by a KNN classifier to predict whether a test video was a Deepfake. Each input video was converted into 300 grayscale histograms, which were subsequently split into batches of 10 to maintain the temporal order before being processed. The batches were fed into an LSTM layer, and its outputs were subsequently processed through two dense layers with 128 neurons, followed by 64 ultimately producing the final classification.

The reported detection accuracies for the three methods were: CNN with RNN – 82.20%, Eye-Blink detection – 81.67%, and Grayscale Histogram – 85.71%. Optical Flow based CNN Optical flow [10] refers to a set of vectors calculated between two sequential frames, allowing the extraction of visible movement between the observer and the scenes. Optical flow is capable of identifying motion discrepancies between artificially generated frames and those naturally recorded by a video camera. The optical flow analysis with CNN to develop an effective Deepfake detection mechanism. The CNN architecture processes the optical flow information extracted to discern patterns characteristic of manipulated videos. There are two CNN model are VGG16 and ResNet50 is used train the model. The VGG16 has achieves 81.61% and ResNet50 has 75.46% of accuracy.

The Time Distributed Approach [5] utilizes spatio-temporal features to identify DeepFake videos. The experiments showcased the model's capability to effectively identify the most pertinent spatio-temporal features crucial for detecting DeepFake content within the DFDC dataset. The DFDC dataset was used to evaluate the proposed manipulation detection method, with the data split into training, validation, and test sets. For testing, 30 frames were taken from each input video and processed by the trained model. Extensive testing was performed across multiple iterations, where fake videos were shuffled and sampled to match the number of real videos. Following several rounds of testing, each model's final score was determined by selecting the value that occurred most often across all results, achieving an overall accuracy of 97.6%.

Texture Based Detection a novel approach utilizing conventional machine learning techniques has been introduced to identify Deepfake video due to their deficiency in capturing facial texture details [2]. At the initial stage, texture features are obtained from the facial region of each frame using image gradients, standard deviation, gray-level co-occurrence matrices, and wavelet transforms. These features precisely capture the subtle details of facial texture.

Subsequently, a Support Vector Machine (SVM) is applied to detect Deepfake videos based on the extracted texture features. The image gradient describes variations in the grayscale of individual pixels within their proximity, serving as an indicator of the texture level in the image. Areas rich in texture, like edges in an image, display significant variations in gray levels, leading to higher gradient values. The gray- level co-occurrence matrix characterizes texture by statistically analyzing the spatial distribution of each pixel in the image. Taking into account both direction and distance, it determines the likelihood of two adjacent pixels with specific gray levels occurring in the image in a defined spatial relationship. The gray-level co-occurrence matrix is created by compiling the probabilities across different gray levels. Using wavelet transform to decompose the image in both horizontal and vertical directions yields low-frequency sub-bands, along with horizontal, vertical and diagonal high-frequency sub-bands. Analyzing the statistics of these sub-bands enables the extraction of the image's texture level, facilitating the categorization of images based on different levels of texture richness. The selected features are subsequently employed for Deepfake video detection using Support Vector Machines . The proposed approach attains an accuracy of 94.4% for high-quality videos, but only 2.6% for low-quality videos.

Automatic Face Weighting Detection is an innovative model structure integrates a Convolutional Neural Network and a Recurrent Neural Network (RNN) for precise identification of facial manipulations in videos [6]. The network automatically selects the most reliable frames for detecting these manipulations, using a weighting system combined with a Gated Recurrent Unit (GRU) to produce a final probability indicating the process of identifying whether a video is real or manipulated consists of three key steps: first, detecting faces across multiple frames using MTCNN (Multi-task Cascaded Convolutional Networks); next, extracting the relevant features from these detected faces. Lastly, the features extracted from the detected faces are utilized to classify the video and determine its authenticity Finally, the extracted features are analyzed to classify the video as authentic or manipulated. from these detected faces; and third, classifying the features to make the final decision. Lastly, the features extracted from the detected faces are examined to classify the video and reach the features obtained from the detected faces are subsequently classified with a CNN to decide if the video is a Deepfake .Third, classifying these features to determine whether the video is a Deepfake, finally estimating predictions through a layer denoted as Automatic Face Weighting (AFW) combined with a Gated Recurrent Unit (GRU) to incorporate temporal information. The GRU prediction obtaining accuracy of 92.61% in the detection.

Eff-Ynet DeepFake Detection and Segmentation is a conventional, 2D Convolutional Neural Network is transformed by incorporating it as the backbone in a U-Net architecture [3].In this setup, the U-Net encoder is implemented with EfficientNet, a CNN architecture recognized for delivering better performance than other models of comparable size. The U-Net decoder mirrors the encoder's structure and produces a segmentation mask for the input image. Incorporate a classification branch at the end of the encoder, producing a classification output for the input image. Simultaneous training is conducted for the two tasks of segmentation and classification. This integration allows the detection process to concentrate on modified pixels, improving the classifier's training, while ensuring that the model's segmentation mask predictions are consistent with the classification results. When the model does not predict a mask, it classifies the input image as real. Conversely, the presence of a substantial predicted mask indicates that the model is confident the image is fake. The model calculates ROC (Receiver Operating Characteristic) curve of 98.7%.

FaceAVCeleb is an unique audio-video deepfake dataset comprising both deepfake videos (Vonly) and their corresponding lip-synced fake audios (Aonly) [14]. The method is helps to create dataset by using most common deepfake techniques for develop a high-quality video and audio deepfake dataset that can be used to detect both audio and video deepfakes simultaneously. FakeAVCeleb is a novel audio-video Deepfake dataset that contains both Deepfake videos and the corresponding synthesized lip-synced fake audios. This method uses carefully selected real YouTube videos of celebrities from four ethnic backgrounds to create a more realistic multimodal dataset, aimed at mitigating racial bias and further supporting develop multimodal deepfake detectors. The dataset performed several experiments using state-of-the-art detection methods to evaluate deepfake dataset and demonstrate the challenges and usefulness of our multimodal Audio-Video deepfake dataset. FaceAVCeleb is a new challenging large-scale dataset, CelebDF [15], includes 5,639 high-quality Deepfake videos of celebrities produced using an enhanced synthesis technique The process conducts a comprehensive evaluation of DeepFake detection methods and datasets to highlight the heightened difficulties introduced by Celeb-DF. The greater challenges posed by Celeb-DF in Deepfake detection, the study performed an extensive evaluation of current detection algorithms and datasets. Different approaches were applied using their recommended dataset for training.

CNN Based Approach of Deepfake detection has gained significant attention in recent years due to the increasing misuse of synthetic media. Traditional approaches rely on handcrafted features, such as facial landmarks and texture irregularities, but these methods often fail when applied to masked or partially occluded faces. The widespread use of face masks after the COVID-19 pandemic has further complicated this problem, requiring the development of more robust and generalizable detection frameworks. Alnaim *et al.* [17] introduced the Deepfake Face Mask Dataset (DFFMD) and proposed a baseline CNN architecture to evaluate detection performance. Their CNN consisted of three convolutional layers with ReLU activation, max-pooling operations, flattening, and fully connected layers with dropout for classification. Using the DFFMD dataset, the model achieved an accuracy of 77.80% on an 80:20 train-test split. The training and validation curves demonstrated gradual improvements across epochs, but the CNN required more iterations to converge and exhibited lower performance compared to deeper transfer learning models. These findings highlight the limitations of shallow CNNs in capturing complex facial manipulations, particularly when faces are masked or occluded. More advanced architectures such as ResNet, VGG, and Inception-based networks have since been explored to overcome these challenges, providing improved generalization and higher accuracy across diverse Deepfake scenario.

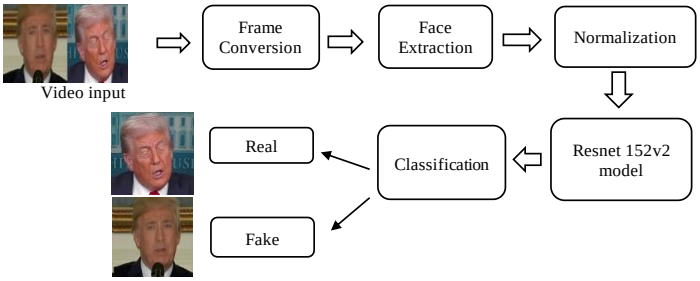

Fig.2. Proposed System Block Diagram

3. METHODOLOGY

3.1 PROPOSED SYSTEM

The primary objective of the proposed system is to accurately detect fake faces in videos, including challenging scenarios where individuals are wearing face masks. This system utilizes a deep learning framework based on the ResNet152V2 architecture, designed to enhance feature representation and boost classification accuracy for Deepfake detection. The method operates on a merged UADFV and DFFMD dataset, featuring videos with and without facial masks. Initially, input videos are broken down into individual frames, and facial regions are identified using the Haar Cascade face detection algorithm. The extracted facial images are then resized to 224×224 pixels (to match ResNet152V2 input requirements) and normalized to ensure consistent model input. For classification, the ResNet152V2 model is trained and tested to differentiate between real and fake faces. The architecture utilizes residual connections and batch normalization, enabling the model to efficiently learn both low- and high-level features, which is particularly effective for handling occluded regions such as masked faces. The system outputs a prediction indicating whether the detected face in each frame is genuine or fake. Experimental evaluation demonstrates the model's robustness, achieving high accuracy across the combined datasets and confirming its potential as a unified framework for reliable Deepfake detection in real-world scenarios.

3.2 DATASET

This system leverages the UADFV and DFFMD datasets, which together provide real and fake videos intended for evaluating Deepfake detection models. A subset of 20 videos from the UADFV dataset and selected samples from the DFFMD dataset were used, which were converted into a total of 3,373 frames for processing. Initially, each video is split into individual frames, and the Haar Cascade classifier is applied to detect facial regions. The detected facial regions are then resized to 224×224 pixels to match the input size required by the ResNet152V2 model. The processed frames from both datasets are combined and divided into different training and testing splits to assess the robustness of the model are 80% Training and 20% Testing,70% Training and 30% Testing,50% Training and 50% Testing. These frames are then used to train and evaluate the ResNet152V2 model for classifying each frame as real or fake. By utilizing residual connections and batch normalization, the architecture effectively captures both low- and high-level features. This improves the model's performance in detecting manipulated facial content across different data splits, even in challenging scenarios such as occlusion due to masks.

3.3 HAAR CASCADE ALGORITHM

The Haar Cascade is a machine learning-based technique that involves to using a substantial set of positive and negative data point images for training the classifier. Regions that include a face are considered positive data points, while those that do not are treated as negative data points. Using these positive and negative examples, a classifier can be trained to decide whether a given region of an image contains a face.

Algorithm :

1: function DetectFaces(InputImage)
2: Input: Video frame or image
3: Output: Detected face regions with bounding boxes
4: GrayImage ← ConvertToGrayscale(InputImage)
5: IntegralImg ← ComputeIntegralImage(GrayImage)
6: for each DetectionWindow in Image do
7: Features ← Extract HaarFeatures(DetectionWindow, IntegralImg)
8: WeakResponses ← PassThroughWeakClassifiers(Features)
9: StrongResponse ← CombineWeakClassifiersAdaBoost (WeakResponses)
10: end for
11: CandidateRegions ← ApplyCascadeOfClassifiers(Strong Response)
12: for each Stage in Cascade do
13: Discard non-face regions
14: Pass potential face regions to the next stage
15: end for
16: MultiScaleDetection(CandidateRegions)
17: for each scale do
18: Resize image and slide detection window
19: Repeat Steps 8–18
20: end for
21: for each Region in CandidateRegions do
22: if Region passes all stages then
23: Mark region as detected face
24: Draw bounding box around Region
25: end if
26: end for
27: return Detected Faces
28: end function

3.3.1 Resnet152V2:

ResNet152V2 is a high-performance deep convolutional neural network, built as an upgraded version of ResNet152. It is particularly designed to achieve efficient image classification and strong feature extraction. Boasting 152 layers it is much deeper than ResNet50V2, allowing it to extract more complex features from large datasets. The increased depth of the network allows it to learn more abstract and complex features, which improves

accuracy, especially in challenging image analysis tasks like deepfake detection. Like ResNet50V2, ResNet152V2 employs residual learning, commonly referred to as skip connections to effectively overcome the problem of vanishing gradients. In residual learning the input to a layer is added to the output of the layer (via skip connections), allowing the network to learn the difference between the input and output instead of learning the entire transformation. This facilitates training of deeper models and enhances gradient flow leading to more accurate and stable learning. In ResNet152V2, pre-activation residual blocks are used, meaning that batch normalization and ReLU activation occur before the convolutional layers. This slight adjustment from earlier ResNet versions like ResNet50 enhances training efficiency and promotes quicker convergence. The architecture of ResNet152V2 in Fig.3 is similar to ResNet50V2, but with more layers, which allows it to capture even more detailed features. The architecture consists of multiple stages with each stage made the network is made up of multiple residual blocks that progressively reduce the spatial resolution of the input image. After traversing these blocks, the network produces a feature representation map. The resulting feature map is passed through a global average pooling layer followed by a fully connected layer to perform the classification task.

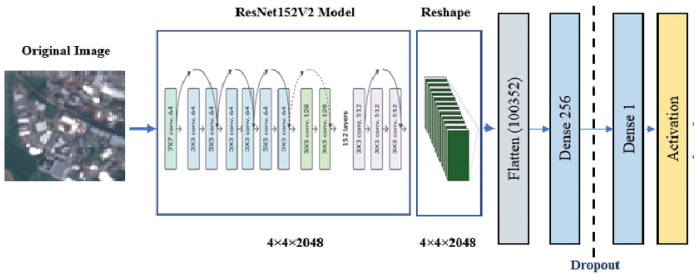

Fig.3. Resnet152V2 Architecture(Source: Research Gate)

4. RESULT AND DISCUSSION

4.1 CONFUSION MATRIX

A confusion matrix is a summary of prediction results for a classification problem. It shows the counts of actual vs. predicted classes for binary classification (e.g., real vs. fake).The confusion matrix is defined by four key components that are True Positives (TP) is a instances where fake faces are correctly identified as fake, True Negatives (TN) is a Instances where real faces are accurately identified as real, False Positives (FP is incorrectly predicted real faces as fake and False Negatives (FN) is instances where the system fails to detect fake faces and identifies them as real. The confusion matrix in a binary classification scenario is represented as follows.

Table.1. Resnet152V2 Confusion Matrix
(80% Training & 20% Testing)

Training	Testing	Actual/Predicted	Fake	Real
80%	20%	Fake	391	2
		Real	13	319
70%	30%	Fake	513	38
		Real	9	450
50%	50%	Fake	799	45
		Real	41	657

Using the confusion matrix, can compute various performance metrics.

4.2 PERFORMANCE METRICS FROM CONFUSION MATRIX

Accuracy represents the ratio of correctly classified instances to the total number of cases.

$$\text{Accuracy}=(TP+TN)/(TP+TN+FP+FN) \quad (1)$$

Precision Indicates the proportion of true positive results out of all predicted positives, reflecting the model's ability to avoid false positives.

$$\text{Precision}=TP/(TP+FP) \quad (2)$$

Recall Indicates the fraction of true positive cases accurately identified, reflecting the model's effectiveness in reducing false negatives.

$$\text{Recall}=TP/(TP+FN) \quad (3)$$

4.3 ROC CURVE

A Receiver Operating Characteristic (ROC) curve is a graphical representation of the performance of a classification model at all classification thresholds. The ROC curve illustrates the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) . True Positive Rate is commonly referred to as Recall or Sensitivity and False Positive Rate is the fraction of genuine faces that are mistakenly classified as fake.

$$TPR=TP/(TP+FN) \quad (4)$$

$$EPR=FP/(FP+TN) \quad (5)$$

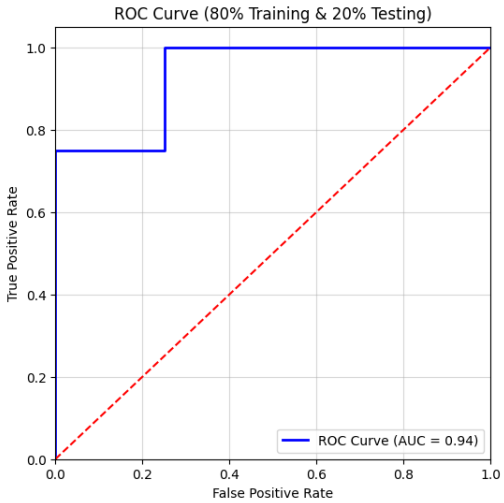

Fig.4. Resnet152V2 ROC Curve(80% Training & 20% Testing)

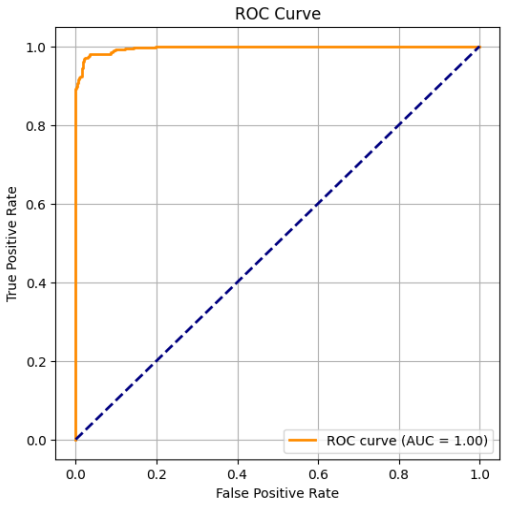

Fig.5. Resnet152V2 ROC Curve(70% Training & 30% Testing)

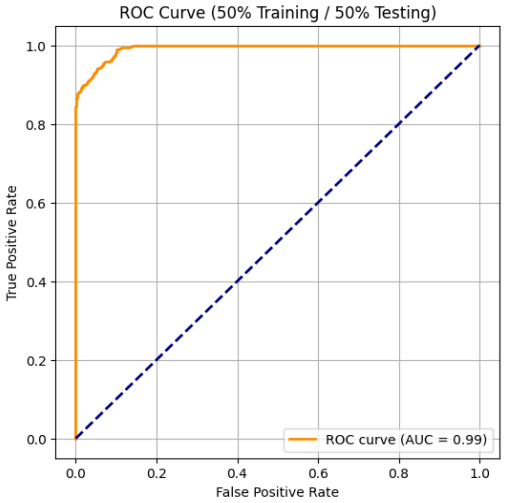

Fig.6. Resnet152V2 ROC Curve(50% Training & 50% Testing)

4.4 CLASSIFICATION REPORT

Table.2. Resnet152V2 classification report

Training	Testing	Class	Precision	Recall	Accuracy
80%	20%	Real	0.97	0.99	0.98
		Fake	0.99	0.96	
70%	30%	Real	0.98	0.93	0.95
		Fake	0.92	0.98	
50%	50%	Real	0.95	0.95	0.94
		Fake	0.94	0.94	

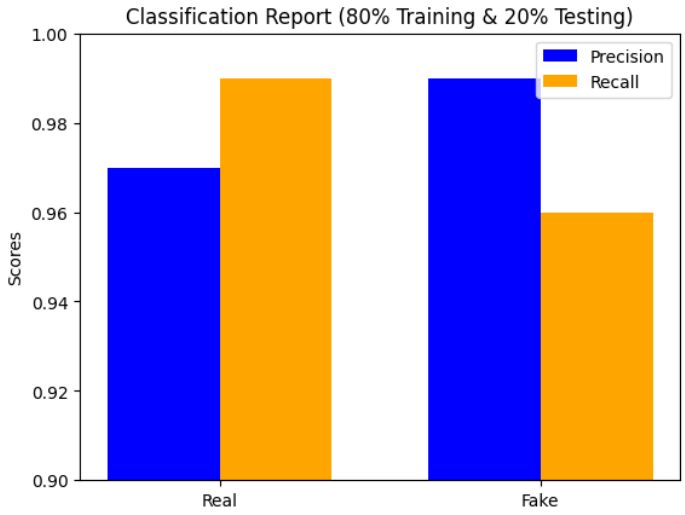

Fig.7. Resnet152V2(80% Training & 20% Testing)

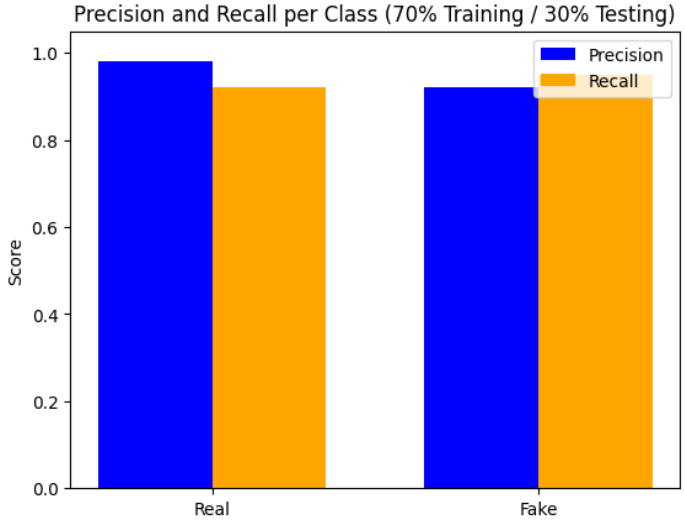

Fig.8. Resnet152V2(70% Training & 30% Testing)

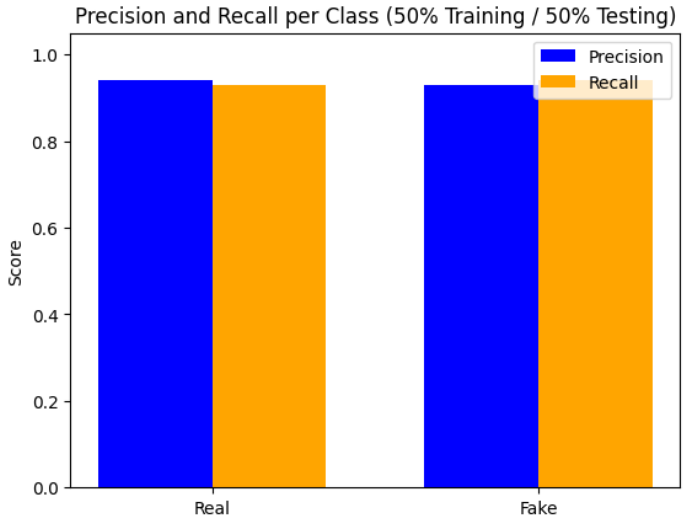

Fig.9. Resnet152V2(50% Training & 50% Testing)

4.5 DETECTED FACE FRAMES

The face area extracted for fake videos and real videos are shown as in Fig.10 and Fig.11.

Fig.10. Fake Video Frames

Fig.11. Real Video Frames

The extracted face images are converted to BGR format to facilitate their transformation into matrix form.

4.6 RESULT DISCUSSION

The ResNet152V2 model was evaluated using standard metrics such as accuracy, precision, and recall. The model was configured with include_top=False, allowing the convolutional layers to act as feature extractors while custom classification layers were added on top for Deepfake detection. Pre-trained ImageNet weights enabled broad visual feature awareness, and the model was fine-tuned on the combined UADFV and DFFMD datasets.

Input frames were resized to 224×224 pixels to match the model's requirements. The convolutional layers of ResNet152V2 effectively capture both low- and high-level features retaining refined facial signs indicative of manipulation. Residual connections prevent vanishing gradient issues ensuring stable learning. The model was evaluated across three train-test splits which are 80:20, 70:30, and 50:50, confirming its adaptability under different data distributions.

Experimental results show that ResNet152V2 can reliably differentiate real and fake faces, even under challenging scenarios like masked or unmasked or partially occluded faces. The combined datasets enhance generalization across diverse Deepfake manipulations. With deep residual learning and batch normalization, the model captures complex hierarchical features achieving accuracies of 98%, 95%, and 94% for the 80:20, 70:30, and 50:50 splits respectively, demonstrating robustness and effectiveness as a unified Deepfake detection framework.

For comparison, Alnaim *et al.* [17] reported that their baseline CNN model on the DFFMD dataset achieved only 77.80% accuracy, with slower convergence and lower overall performance compared to transfer learning models. This highlights the superior generalization and feature extraction capability of deeper architectures like ResNet152V2 over shallow CNNs for Deepfake detection, particularly in scenarios involving masked or unmasked occluded faces.

Table.3. Performance Comparison between CNN (DFFMD) and ResNet152V2(UADFV+DFFMD)

Model	Dataset	Split	Accuracy
CNN (Alnaim et al. [17])	DFFMD	80:20	77.80
ResNet152V2 (Proposed)	UADFV+DFFMD	80:20	98.00
ResNet152V2 (Proposed)	UADFV+DFFMD	70:30	95.00
ResNet152V2 (Proposed)	UADFV+DFFMD	50:50	94.00

4.7 TRAINING AND VALIDATION ACCURACY

During the initial epochs, the model showed moderate learning, with accuracy fluctuating between ~0.52–0.61 across all splits. After fine-tuning, final training accuracies reached 0.9847 (80:20), 0.9812 (70:30) and 0.9735 (50:50) indicating effective learning of discriminative features from the combined UADFV and DFFMD datasets.

Early fluctuations ranged between ~0.55–0.70, reflecting initial challenges in generalization. With training, validation accuracy improved and stabilized at 0.9793 (80:20), 0.9760 (70:30), and 0.9648 (50:50) showing successful adaptation to unseen frames, including masked or unmasked or partially occluded faces.

4.8 TRAINING AND VALIDATION LOSS

Loss steadily decreased across epochs, reaching 0.0534 (80:20), 0.0561 (70:30), and 0.0620 (50:50) demonstrating that the model minimized errors and effectively captured key facial features. Initially fluctuating between 0.4957–0.2250, validation loss eventually stabilized at 0.0539 (80:20), 0.0572 (70:30) and 0.0635 (50:50) confirming robust generalization to new data.

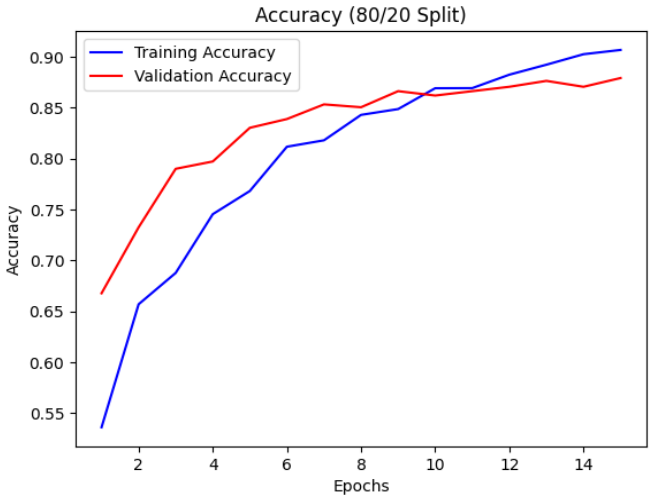

Fig.12. The curve of training Accuracy & validation Accuracy of Resnet152v2 for 80 % Training & 20% Testing.

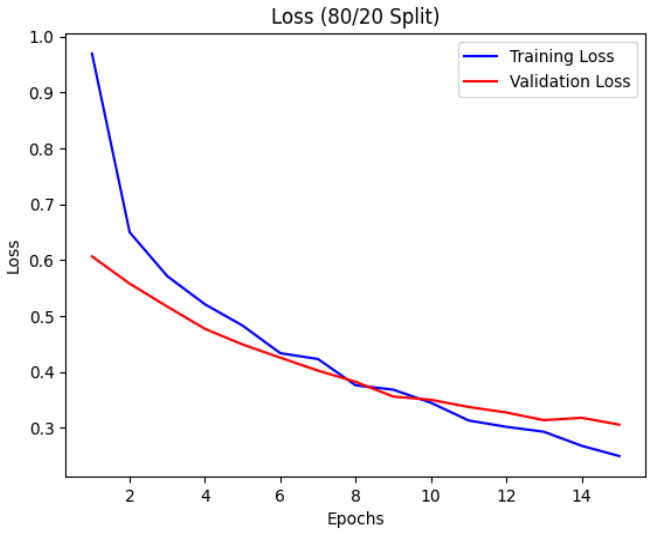

Fig.13. The curve of training Loss & validation Loss of Resnet152v2 for 80 % Training & 20% Testing.

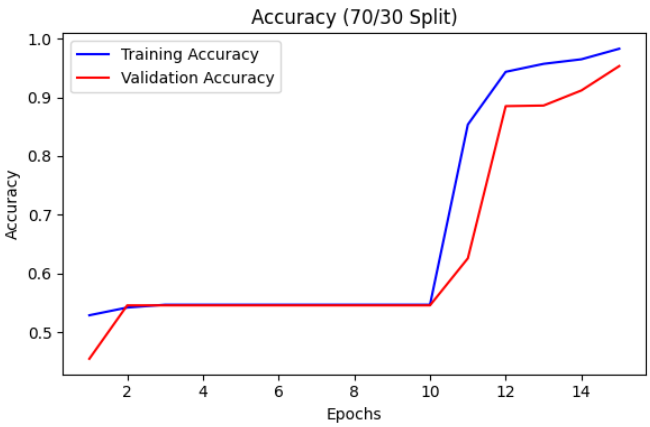

Fig.14. The curve of training Accuracy & validation Accuracy of Resnet152v2 for 70 % Training & 30% Testing.

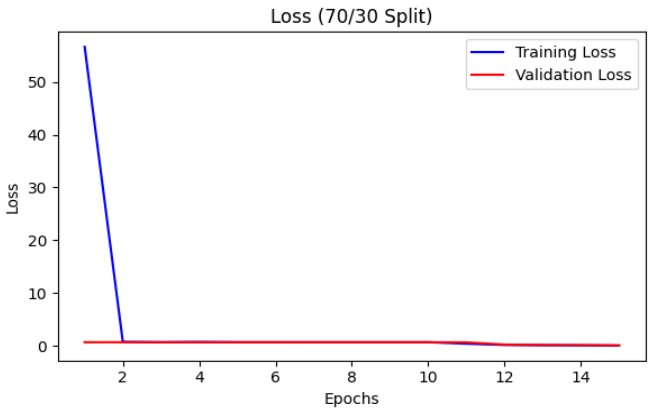

Fig.15. The curve of training Loss & validation Loss of Resnet152v2 for 70 % Training & 30% Testing.

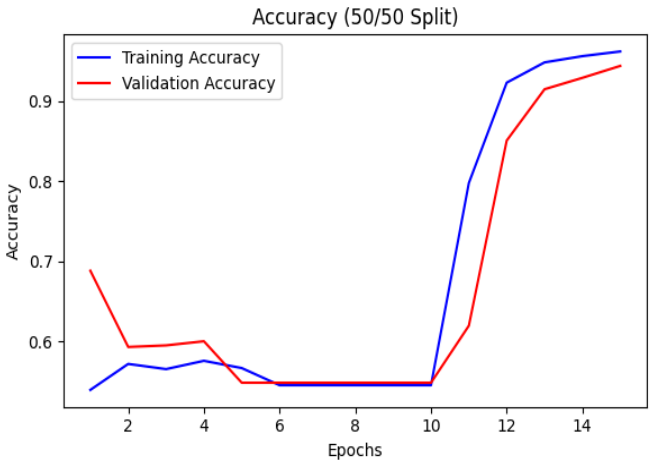

Fig.16. The curve of training Accuracy & validation Accuracy of Resnet152v2 for 50 % Training & 50% Testing.

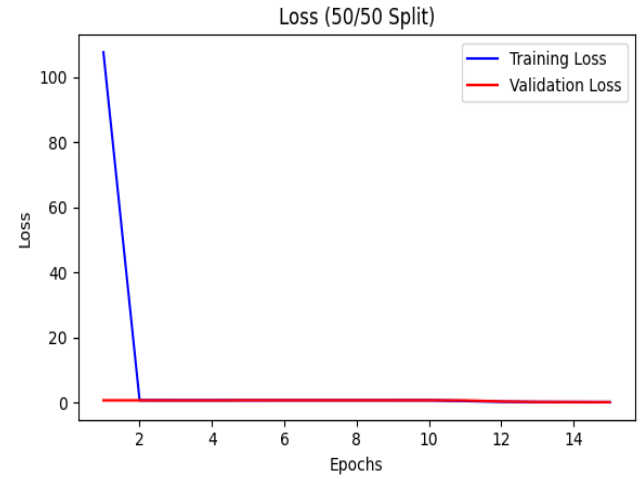

Fig.17. The curve of training Loss & validation Loss of Resnet152v2 for 50 % Training & 50% Testing.

5. CONCLUSION

Deepfake technology is rapidly evolving, producing highly realistic synthetic videos that pose serious threats to privacy, security, and public trust. The widespread use of face masks during the COVID-19 pandemic added challenges, as occluded facial features make it difficult for traditional algorithms to distinguish authentic from manipulated content. This study developed a deep learning framework using ResNet152V2 to detect deepfake videos, including masked faces. The model was trained and evaluated on a combined UADFV and DFFMD dataset containing real and fake videos with and without masks. Preprocessing included frame extraction, face detection with Haar Cascade, resizing to 224×224 pixels, and normalization. Experimental results showed that ResNet152V2 performed robustly across different train-test splits. Based on training accuracy, the model achieved 0.9847 for the 80:20 split, 0.9812 for the 70:30 split, and 0.9735 for the 50:50 split, demonstrating its ability to effectively learn discriminative features from both real and fake frames. Validation results confirmed that the model generalized well to unseen data, including frames with masked or

morphed or partially occluded faces. Despite the relatively small dataset, these results highlight the potential of deep residual networks for reliable deepfake detection. Future work will focus on expanding the dataset to cover diverse deepfake scenarios and exploring alternative model architectures and training strategies to enhance detection accuracy, robustness, and generalization. These advancements are crucial for maintaining the authenticity of digital media in the era of rapidly evolving deepfake technology.

REFERENCES

[1] D. Wodajo and S. Atnafu, "Deepfake Video Detection using Convolutional Vision Transformer", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, Vol. 6, pp. 1-9, 2021.

[2] B. Xu, J. Liu, J. Liang, W. Lu and Y. Zhang, "DeepFake Videos Detection based on Texture Features", *Computers, Materials and Continua*, Vol. 68, No.1, pp, 1375-1388, 2021.

[3] E. Tjon, M. Moh and T.S. Moh, "Eff-YNet: A Dual Task Network for DeepFake Detection and Segmentation", *Proceedings of International Conference on Ubiquitous Information Management and Communication*, Vol. 2, pp. 1-8, 2021.

[4] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales and J. Ortega-Garcia, "DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, Vol. 8, pp. 1-23, 2020.

[5] A. Singh, A.S. Saimbhi, N. Singh and M. Mittal, "DeepFake Video Detection: A Time-Distributed Approach", *SN Computer Science*, Vol. 1, No. 4, pp. 1-8, 2020.

[6] D.M. Montserrat, H. Hao, S.K. Yarlagadda, S. Baireddy, R. Shao, J. Horvath, E. Bartusiak, J. Yang, D. Guera, F. Zhu and E.J. Delp, "DeepFakes Detection with Automatic Face Weighting", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 668-669, 2020.

[7] A. Pishori, B. Rollins, N. Van Houten, N. Chatwani and O. Uraimov, "Detecting Deepfake Videos: An Analysis of Three Techniques", *Proceedings of International Conference on Machine Learning*, Vol. 6, pp. 1-11, 2020.

[8] T. Karras, S. Laine and T. Aila, "A Style-based Generator Architecture for Generative Adversarial Networks", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 4401-4410, 2019.

[9] A. Siarohin, S. Lathuiliere, S. Tulyakov, E. Ricci and N. Sebe, "First Order Motion Model for Image Animation", *Proceedings of International Conference on Neural Information Processing Systems*, Vol. 32, pp. 1-11, 2019.

[10] I. Amerini, L. Galteri, R. Caldelli and A.D. Bimbo, "Deepfake Video Detection through Optical Flow based CNN", *Proceedings of International Workshop on Computer Vision and Pattern Recognition*, pp. 1-3, 2019.

[11] Y. Choi, M. Choi, M. Kim, J.W. Ha, S. Kim and J. Choo, "Star-GAN: Unified Generative Adversarial Networks for Multi-Domain Image to Image Translation", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 8789-8797, 2018.

[12] Y. Li and S. Lyu, "Exposing DeepFake Videos by Detecting Face Warping Artifacts", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 1-7, 2018.

[13] T. Karras, T. Aila, S. Laine and J. Lehtinen, "Progressive Growing of GANs for Improved Quality, Stability and Variation", *Proceedings of International Conference on Neural and Evolutionary Computing*, Vol. 7, pp. 1-26, 2018.

[14] M. Kim, and S.S. Woo, "FakeAVCeleb: A Novel Audio Video Multimodal Deepfake Dataset", *Proceedings of International Conference on Multimedia*, pp. 1-22, 2021.

[15] Y. Li, X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-DF: A Large-Scale Challenging Dataset for Deepfake Forensics", *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 3207-3216, 2020.

[16] Martin Schicklgruber, "Deepfake Detection", Master Thesis, Department of Computer Science, Johannes Kepler University, pp. 1-88, 2022.

[17] N.M. Alnaim, Z.M. Almutairi, M.S. Alsuwat, H.H. Alalawi, A. Alshobaili and F.S. Alenezi, "DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era with Deepfake Detection Algorithms", *IEEE Access*, Vol. 11, pp. 16711-16722, 2023.