DOI: 10.36108/ujees/4202.60.0111
UNIOSUN Journal of Engineering and Environmental Sciences. Vol. 6 No. 1. March. 2024

DEEPFAKE FACE RECOGNITION THROUGH MODIFIED AND IMPORVED DEEP TRANSFER LEARNING

Sobowale, A. A., Adetona, B. J., Soladoye, A. A., Omodunbi, B. A.

Abstract The manipulation of photo, audio, and video content has been a topic of interest for many years, as people uses fake faces to indulge in various immoral act like pornography, fraud, and defamation. In the early days, classification of real and fake faces was done using traditional methods such as editing frames by frame or using chroma keying, this traditional approach is time consuming and lacks enough editors that have the technical skills to do the frame-by-frame edition or use the chroma keying. With technological advancement, new techniques have been developed that allow for much more sophisticated and realistic manipulations, one such technique is deepfakes. Deepfakes are created using deep learning algorithms to swap or replace faces in videos or images. This can be done with a high degree of realism, making it difficult to distinguish between real and fake content. This research aims to develop a deep fake detection system using deep transfer learning (modified VGG19 and ResNet50 models), these two models were chosen over other CNN architectures due to their proven better performance, faster recognition time and lesser memory usage. The research modified the original VGG19 and ResNet architectures by replacing the last five layer with a customized dense layers that will help with faster and accurate recognition of faces. A balanced dataset comprising 70,000 real faces from the Flickr dataset and 70,000 fake faces generated by StyleGAN was utilized. This research employed hold-out evaluation method. VGG19 gave an accuracy, f1score of 91.59% and 91.47% respectively while RestNet50 gave an average accuracy and F1score of 96.61% and 96.59% respectively on the testing dataset. This shows that ResNet 50 gave the best performance both on the training, validation and testing dataset. The developed system was also compared with other state-of-art methods and they were all outperformed.

Keywords:

I. Introduction

The manipulation of photographic, auditory, and audiovisual material has been a subject of interest for a considerable period of time. Initially, this endeavor was accomplished through conventional means such as manually altering individual frames or implementing chroma keying techniques. Nonetheless, the emergence of artificial intelligence (AI) and machine learning (ML) has given rise to novel methodologies that enable more intricate and realistic manipulations. One of such method is known as deepfakes. Nonetheless, the emergence of artificial intelligence (AI) and machine learning (ML) has given rise to novel methodologies that enable more intricate and realistic manipulations. One of such method is known as deepfakes. Deepfakes involve the utilization of deep learning algorithms to interchange, or substitute faces within videos or images. This process can be executed with a high level of verisimilitude, making it arduous to discern between authentic and counterfeit content. In addition, deepfakes boast several potential applications, encompassing the realms of entertainment, education, and research. Nonetheless, they also carry the potential for misuse, including the dissemination of disinformation or the fabrication of news. Deepfakes pertain to synthetic media that has

Sobowale, A. A., Adetona, B. J., Soladoye, A. A., Omodunbi, B. A.
(Department of Computer Engineering, Federal University, Oye-Ekiti, Nigeria)
Corresponding Author: afeez.soladoye@fuoye.edu.ng
Phone Number:

been digitally altered in order to convincingly replace one individual's likeness with that of another. The manipulation of facial features through deep generative methods defines the concept of deepfakes [1].

While the act of fabricating counterfeit content is not novel, deepfakes leverage the formidable capabilities of machine learning and artificial intelligence to manipulate or generate visual and auditory content that can more effectively deceive [2]. Deepfakes have garnered widespread attention due to their potential for generating illicit material involving child sexual abuse, celebrity pornography, revenge pornography, disinformation, hoaxes, cyberbullying, and financial fraud [3]. The potential employment of deepfakes encompasses a range of criminal activities, such as tarnishing the reputation of a prominent figure by assuming the guise of a family member [4]. The Federal Bureau of Investigation (FBI) has identified deepfake technology as an emergent peril, cautioning that malicious actors will exploit synthetic content for cyber and foreign influence operations. Notably, deepfakes possess the capacity to undermine public trust through the propagation of misinformation campaigns, exert influence on political elections, compromise national security, disrupt the stock market, facilitate corporate espionage, and more [4].

Sensity AI, a research firm that has diligently monitored online deepfake videos since December 2018, has consistently discovered that between 90% and 95% of these videos are nonconsensual pornographic content. A staggering 90% of this content specifically targets women [5]. Deepfakes possess the potential to manipulate elections by disseminating detrimental videos of candidates. If properly timed, this manipulation could alter the outcome of an electoral process. The perpetrators of such actions could be the opposing campaign team, foreign governments, or even individuals. Over time, this detrimental practice could undermine the democratic process by instilling doubt regarding its legitimacy. Additionally, deepfakes have the capacity to erode trust in institutions. For instance, a deepfake video featuring the leader of an immigration institute making racist remarks could inflict significant harm upon the immigration system. The research will yield various advantages in multiple domains such as Entertainment, Education, and Research. For instance, photographs and videos have been employed as substantial evidence in police investigations and courtrooms. Creating detection tools help law enforcement agencies and other organizations to identify and prevent the spread of fakes images.

Many researchers have used various Traditional machine learning and Deep learning approach for detection of faces, distinguishing a real image from fake ones, differentiating between AI generated and real images in videos and pictures. Subsequently, different studies carried out by these various and numerous research are reviewed in this section, to give insight on start-of-art in facial recognition and detection.

[6] also propose the detection of counterfeit faces in AI generated images and videos. Their study aims to mitigate the errors that may arise from the creation of Deep fakes through slicing, which can be exposed when 3D head poses are estimated from the face images. The Levenberg-Marquardt algorithm is employed for 3D head pose estimation, while the SVM classifier with Radial Basis Function (RBF) kernels is utilized for classification. The study was only evaluated

using DARPA GAN Challenge dataset and only SVM was used as the classifier without comparing its performance with other classifiers for validation.

[7] conducted a study to explore the potential of Generative Adversarial Networks (GANs) to imprint specific characteristics on the images they generate. The researchers employed various well-known GAN models, including Cycle-GAN, Pro-GAN, and Star-GAN, for their experiments. Similar to the photo-response non-uniformity (PRNU) pattern, the authors adopted a pipeline to extract the imprinted characteristics, referred to as fingerprints. The study demonstrated that each GAN model does indeed leave a distinct fingerprint on the generated images. In addition, the authors performed source identification experiments and successfully utilized the fingerprints to reliably determine the source of the images. [8] conducted a similar study to detect images generated by GANs. They proposed a method for identifying unique fingerprints left by different GAN models in the generated images. A deep convolutional neural network was employed to train an attribution classifier capable of predicting the image source. Furthermore, the authors introduced three variations of the network to analyze which components of the images contain the fingerprints. The study's findings confirmed that GANs do indeed leave distinguishable and consistent fingerprints in their generated images, which can be used for image attribution. These two studies are similar just that the first went further to detect the source of the image which would really be a good advantage for the analyzer to know where the image was generated from and evaluate the source for future use.

[9] presented an extensive investigation into the detection of manipulated facial images. The authors introduced an automated benchmark for facial manipulation detection, utilizing four state-of-the-art methods: DeepFakes, Face2Face, FaceSwap, and NeuralTexturesm. The study employed a convolutional neural network (CNN) for facial detection, which significantly outperformed human observers in automatically and reliably detecting such manipulations. However, it is important to note that the detection method was implemented and evaluated on a specific dataset, without assessment of its performance on other datasets or real-world scenarios. Moreover, the method is specifically designed to detect manipulations created by certain techniques, and its effectiveness against other manipulation techniques remains unknown. [10] took an approach to detect synthetic content in portrait videos as a preventive measure against the emerging threat of deep fakes. The authors initially employed several signal transformations for the pairwise separation problem, achieving an accuracy of 99.39%. Subsequently, they utilized these findings to develop a generalized classifier for fake content by analyzing the proposed signal transformations and their corresponding feature sets. Additionally, the authors generated novel signal maps and utilized a Convolutional Neural Network (CNN) to enhance their traditional classifier for detecting synthetic content. As part of their evaluation process, the authors compiled an 'in the wild' dataset of fake portrait videos. They evaluated their detection method, FakeCatcher, on multiple datasets, achieving accuracies of on Face Forensics, Face Forensics++, CelebDF, and their new Deep Fakes Dataset, respectively. However, it is important to recognize that the authors' work is limited as it heavily relies on

detecting artifacts specific to certain models and is therefore not suitable as a general mechanism for detecting synthetic portrait videos.

[11] presents a methodology that aims to identify DeepFake images, specifically counterfeit faces, by utilizing a straightforward technique for extracting features based on frequency domain analysis. To identify these fabricated elements, the authors propose a technique that entails a classical frequency domain analysis followed by a rudimentary classifier. The frequency domain analysis is executed through the application of the Discrete Fourier Transform (DFT) and the Azimuthal Average, which compresses the 2D information into a 1D representation. Logistic Regression, Support Vector Machines, and K-Means Clustering were employed as the classifiers. This methodology was assessed using a newly developed high-resolution dataset called Faces-HQ, which was created by merging various public datasets containing authentic and counterfeit faces. This methodology achieves a very promising classification accuracy even when trained on a minimal number of 20 annotated samples. Moreover, it also accomplishes a perfect accuracy in a supervised setting (LR, SVM) as well as in an unsupervised setting (K-means) using the medium-resolution CelebA dataset. Lastly, when evaluating low-resolution video sequences from the FaceForensics++ dataset, the methodology attains 90% accuracy in detecting manipulated videos.

[12] introduces a novel method for representing images called face X-ray, which aims to detect forgery in face images. The authors note that many existing face manipulation techniques involve blending the altered face with a background image. The face X-ray technique produces a greyscale image that can determine if the input image is composed of a blend of two images from different sources. Through extensive experimentation, the authors demonstrate that face X-ray remains effective in detecting forgery generated by unseen face manipulation techniques, while other existing algorithms for face forgery or deepfake detection experience a significant decrease in performance. However, the authors acknowledge that their approach has limitations. It assumes the presence of a blending step and does not rely on knowledge of the specific artifacts associated with a particular face manipulation technique. While this level of generality encompasses most existing face manipulation algorithms, it may not be effective for all types of face forgery.

[13] investigate the use of deep learning techniques for detecting deepfake videos, particularly in scenarios involving high compression. Their approach focuses on enhancing the feature space distance between clusters of real and fake video embedding vectors to improve the binary classification of deepfakes. The authors employ Multitask Cascaded Convolutional Networks (MTCNN) to extract faces from frames and utilize the XceptionNet architecture for video classification in their dataset. Additionally, they experiment with recurrent neural networks and convolutional 3D networks to enhance classification in low-resolution videos. However, their most successful approach involves metric learning, where semi-hard triplets are utilized to differentiate between fake and real video embedding vectors. The authors validate their method on two datasets: the Celeb-DF dataset and the FF++ dataset. They achieve a state-of-the-art Area under the Curve (AUC) score of 99.2% on the Celeb-DF dataset

and an accuracy of 90.71% on a highly compressed Neural Texture dataset.

[14] presents an innovative technique for detecting fake face images generated by various face image manipulation (FIM) techniques. The authors propose the Adaptive Residuals Extraction Network (AREN), a pre-processing module designed to suppress image content and emphasize tampering artifacts. AREN employs an adaptive convolution layer to predict image residuals, which are then incorporated into subsequent layers to maximize manipulation artifacts by adjusting weights during the back-propagation process. The authors combine AREN with a convolutional neural network (CNN) to create a fake face detector called ARENnet. The results demonstrate that ARENnet achieves an average accuracy of up to 98.52% in detecting fake face images generated by various FIM techniques, surpassing existing state-of-the-art methods. When faced with detecting face images with unknown post-processing operations, the detector still achieves an average accuracy of 95.17%. However, the authors acknowledge that the performance of ARENnet may degrade or become completely ineffective when detecting face images with unknown post-processing operations. Hence, further research is necessary to enhance the generalization capability of the detector.

[15] investigate the utilization of spatiotemporal convolutional networks for the purpose of identifying deepfake videos. The authors posit that while most existing methods for detecting deepfakes rely on individual video frames and fail to take advantage of temporal information, their approach employs spatiotemporal convolutional techniques to identify deepfakes. The authors utilize the Celeb-DF dataset as a benchmark for their methods, which surpass the performance of state-of-the-art frame-based detection methods. Multiple network architectures, including RCN, R3D, ResNet Mixed 3D-2D Convolution, ResNet (2+1)D, and I3D, are employed along with some pre-processing techniques to eliminate extraneous information from the videos. The authors discover that their methods achieve high ROC-AUC scores and accuracies, with the R3D network outperforming the other networks. However, the authors acknowledge that the performance of the networks may be impacted by the imbalance between positive and negative samples in the training set.

[16] introduce an innovative approach for the detection of deepfakes in videos. The authors propose a network structure consisting of two branches that isolate digitally manipulated faces by enhancing artifacts while suppressing the high-level face content. One branch of the network structure propagates the original information, while the other branch suppresses the face content and enhances multi-band frequencies using a Laplacian of Gaussian (LoG) as a bottleneck layer. Additionally, the authors introduce a novel cost function that compresses the variability of natural faces and distances unrealistic facial samples in the feature space. When compared to previous work, the authors' method demonstrates promising results on the FaceForensics ++, Celeb-DF, and Facebook's DFDC preview benchmarks. The authors' method processes sequences of aligned faces from a video, extracts discriminative features using the backbone, and employs bi-directional long short-term memory (LSTM) for recurrent modeling under the supervision of their new loss. The entire network is trained end-to-end to enable the recurrent model to back-propagate to the feature extractor. The authors' method exhibits favorable video-level

performance in terms of video-level AUC when cross-testing.

[17] present a novel technique for detecting face swapping and other identity manipulations in single images. The authors put forth an approach that incorporates two networks: a face identification network that takes into account the face region bounded by a precise semantic segmentation, and a context recognition network that considers the context surrounding the face (e.g., hair, ears, neck). The authors describe a method that utilizes the recognition signals from these two networks to identify discrepancies, thereby providing a supplementary detection signal that enhances the performance of conventional real vs. fake classifiers commonly used for detecting fake images. Deep learning techniques and the Xception architecture are employed by the authors for their face and context recognition networks. The networks are trained on images from the VGGFace2 dataset. The authors discover that their method achieves state-of-the-art results on the FaceForensics++, Celeb-DF-v2, and DFDC benchmarks for face manipulation detection, and is even able to generalize to detect fakes produced by previously unseen methods.

[18] Introduced a methodology for identifying deepfake videos through the utilization of a Convolutional Vision Transformer (CViT). The CViT is an amalgamation of a Convolutional Neural Network (CNN) and a Vision Transformer (ViT). The CNN serves the purpose of extracting learnable characteristics from the input data, while the ViT takes these acquired features as input and classifies them by means of an attention mechanism. The researchers trained their model on the DeepFake Detection Challenge Dataset (DFDC) and achieved a competitive outcome, with an accuracy rate of 91.5%, an AUC value of 0.91, and a loss value of 0.32. The authors contend that their model represents a significant contribution to the field due to its incorporation of a CNN module into the ViT architecture, a novel undertaking. Additionally, the authors underscore the importance of data preprocessing in deepfake detection and propose an extensive data preprocessing pipeline.

[19] critically scrutinize the existing methods for detecting synthetic images, particularly those generated by Generative Adversarial Networks (GANs). The researchers employ an assortment of machine learning techniques and algorithms in their analysis, including deep learning-based methods, convolutional neural networks (CNNs), and autoencoder-based architectures. They also delve into the utilization of augmentation and training strategies to enhance the generalization capability of the detectors. The findings of the study disclose that while certain detectors perform admirably under optimal circumstances, their performance significantly deteriorates when confronted with real-world challenges such as image compression and resizing. The authors conclude that although the detection of GAN images is not an insurmountable task, it remains far from trivial and necessitates further research and development.

From all the related works above, it will be observed that though most of the studies employed or incorporated Convolutional Neural Networks (CNN) into their methodology but none of them employed a VGG19 or ResNet50 architecture of CNN and for the little ones that actually employed this architecture, the modifications done on the

architectures in this research were not done by them. So this research aimed to fill that research gap of using a generic VGG19 and ResNet50 architectures.

II. Materials and Methods

In this section, the implementation of deepfake face detection using transfer learning was discussed, along with the comprehensive methodology employed for detecting fake faces using transfer learning. This approach employed in this study can be classified as an undirected classification method. Specifically, two models were developed in this study to address the issue of deepfake face detection. One of these models is a modified version of VGG19, while the other is a modified version of ResNet50. The performance of these models was evaluated and compared using various metrics, including Accuracy, Precision, Recall, F-Score, and AUC. This methodology is systematically represented in Figure 1 for simplicity and clarification.

A. Data acquisition

A dataset containing 70,000 fake faces and 70,000 real faces was obtained from Kaggle, a public data repository. This dataset comprises of 70,000 real faces from the Flickr dataset, collected by Nvidia, and 70,000 fake faces sampled from the 1 million fake faces generated by StyleGAN and provided by Bojan. Each image has dimensions of 256 by 256. The dataset was divided into training, validation, and test sets. The training set consists of 100,000 images, the validation set contains 20,000 images, and the test set comprises 20,000 images. The data is organized into three folders, referred to as data sources. These folders are named train, valid, and test, corresponding to the training set, validation set, and test set, respectively. Within each data source, there are two folders named fake and real, representing the folders for fake images and real images, respectively. The fake and real folders contain jpg images of faces

B. Image pre-processing

Preprocess input function was used for both the developed VGG-19 and ResNet-50 models. The purpose of this function is to convert the input images, represented in RGB (Red, Green, Blue), into BGR (Blue, Green, Red). Following this conversion, the function proceeds to center each color channel in relation to the ImageNet dataset, without scaling. This pre-processing step is crucial as the pre-trained VGG-19 and ResNet-50 models were trained on the ImageNet dataset, which adheres to a specific distribution of color values. By centering the input data with respect to this dataset, we ensure that the input data exhibits a similar distribution to the data used during model training. Consequently, such an adjustment can potentially enhance the model's performance.

C. Modified VGG-19 architecture

VGG-19 is a variant of the VGG model, consisting of 19 layers, including 16 convolutional layers, 3 fully connected layers, 5 MaxPool layers, and 1 SoftMax layer. It is a deep convolutional neural network that has been trained on over a million images from the ImageNet database. The network is capable of classifying images into 1000 object categories, encompassing items such as keyboards, mice, pencils, and various animals. Consequently, the network has acquired comprehensive feature representations for a wide range of images [20] (Sudha & Ganeshbabu, 2020). To execute transfer learning on the pre-trained VGG 19 model, this study conducted the removal of the

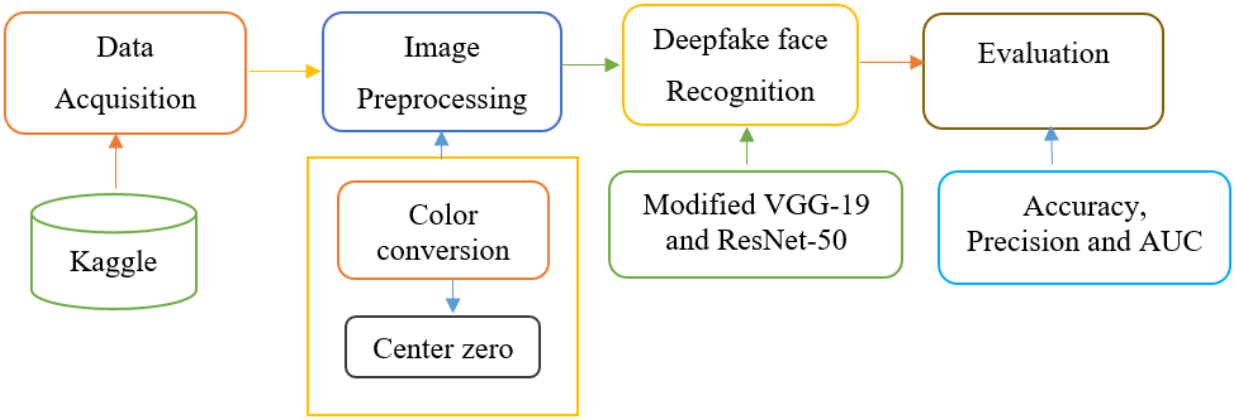

Figure 1: Overview of research methodology

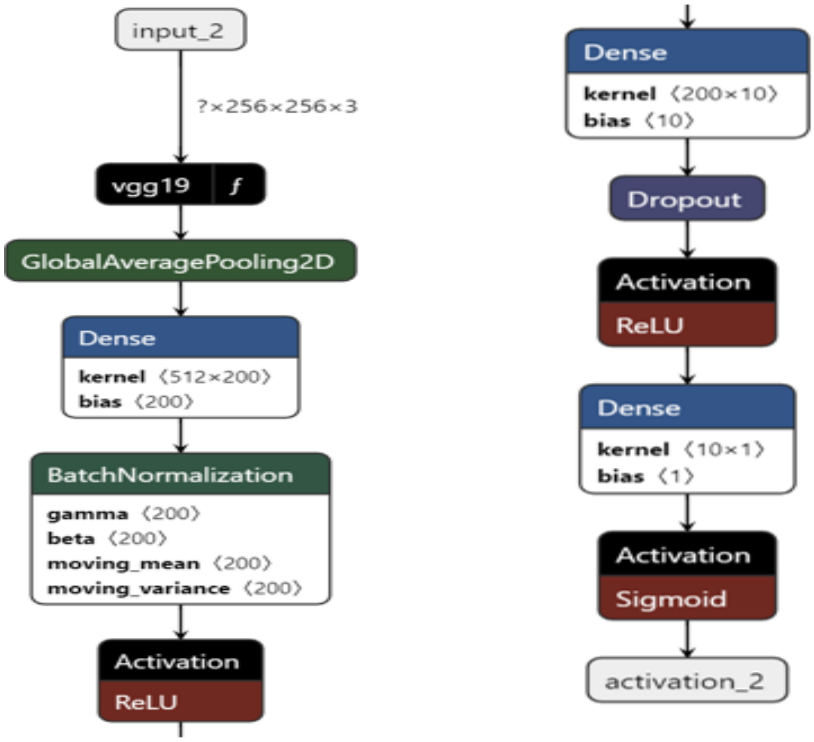

Figure 2: Flowchart of modified VGG 19 architecture

final 5 layers and introduced a customized set of layers, which encompassed a Global average pooling layer, Batch normalization, and three dense layers comprising of 200, 10, and 1 neurons respectively. Additionally, a dropout rate of 0.2 was incorporated. The batch normalization and output layers employed the ReLu and Sigmoid activation functions respectively. Furthermore, the alteration of the input shape from (224, 224, 3) to (256, 256, 3) was implemented. The subsequent section presents the architecture of the adjusted VGG 19 model. This modified architecture is shown in Figure 2

D. Modified ResNet-50 architecture

ResNet 50, a convolutional neural network (CNN), boasts a depth of 50 layers. It was initially introduced in the publication "Deep Residual Learning for Image Recognition" by He et al. (2015). ResNet 50 is widely recognized as one of the most prominent CNNs utilized for image classification, having achieved state-of-the-art outcomes across a diverse range of image classification benchmarks. Functioning as a residual network, ResNet 50 consists of a sequence of residual blocks. Each residual block comprises of two convolutional layers, followed by a shortcut connection. The inclusion of this shortcut connection allows the residual block to learn residual information, which signifies the disparity between the input and output of the block. The residual blocks within ResNet 50 are organized in a hierarchical manner. The initial residual blocks are responsible for capturing low-level features such as edges and textures, whereas the deeper residual blocks are tasked with acquiring high-level features encompassing objects and scenes. The original ResNet 50 encompasses a total of 25,583,592 trained parameters.

In this study, the researchers modified the pre-trained ResNet 50 by substituting its final layer with 8 novel layers. Furthermore, custom layers were added to the Modified ResNet, including three dense layers consisting of 50, 10, and 1 neurons respectively. A batch normalization layer was introduced alongside the ReLu activation function, and a dropout rate of 0.2. Subsequently, Sigmoid was implemented as the activation function for the output layer. Additionally, the input shape was adjusted from (224, 224, 3) to (256, 256, 3). This modified architecture is represented in Figure 3.

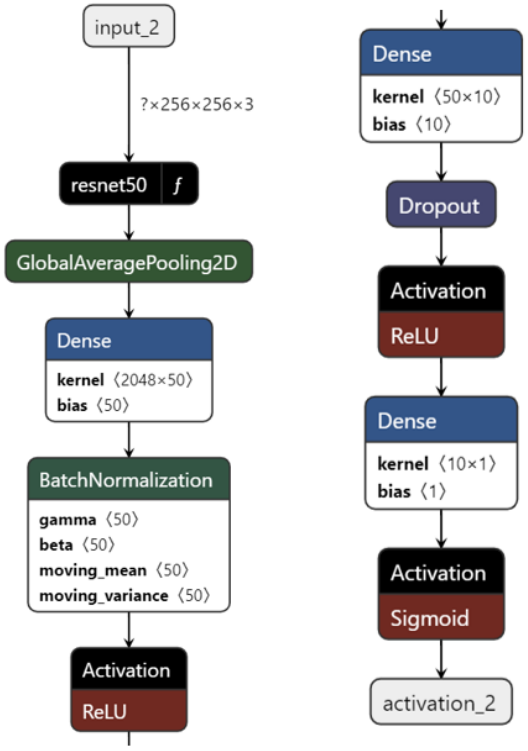

Figure 3: Flowchart of the modified ResNet 50

E. Experimental setup

The construction of all models was carried out using TensorFlow version 2.10.1, with Python serving as the programming language. The training of the models took place on a Windows 11 PC featuring the following specifications: 32GB RAM, an NVIDIA GeForce RTX 3070 Ti GPU equipped with 24GB of memory, and a 12th Gen Intel Core i7 processor. This formidable hardware configuration facilitated the efficient training of the models. Jupyter Notebook was used as the Integrated Development Environment (IDE). The Jupyter Notebook used is part of the Anaconda Navigator software.

F. Evaluation metrics

To assess and compare the performance of the two proposed models, we computed the subsequent metrics: Accuracy, Precision, Recall, F-Score, and Area Under ROC Curve.

i. Accuracy: It is the most fundamental evaluation metric, quantifies the percentage of accurate predictions made by a model. It is derived by dividing the number of correct predictions by the total number of predictions made by the model. This metrics is represented in equation 1.

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

ii. Precision: This measures the proportion of true positive predictions among all the positive predictions made by the model. It is calculated by dividing the number of true positives by the sum of true positives and false positives. Formula for calculating precision is represented by equation 2.

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

iii. Recall: It is also known as sensitivity, measures the proportion of true positive predictions among all the actual positive samples in the dataset, specifically the deepfake faces in our context. It is calculated by dividing the number of true positives by the sum of true positives and false negatives.This is represented by equation 3.

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

iv. F1 Score: It is a weighted average of Precision and Recall with equal weights, serves to balance the trade-off between precision and recall. F1 score is represented in equation 4.

$$F1 = \frac{2 \times precision \times recall}{precision+recall} \quad (4)$$

v. Area Under ROC (AUC): It is a performance measurement for classification problems under various thresholds. It utilizes the Receiver Operating Characteristic (ROC), which is a probability curve, to quantify the degree or measure of separability. A higher AUC value indicates a greater capability of the model to distinguish between classes.

III. Results and Discussion

The results obtained through different stages of experimentations were discussed in this section for clarification. This section cut across some sub sections like the description of the hyperparameters used, the experimental results from implementing the two models and their

comparison with existing studies on deepfake face detection.

A. Optimal hyper-parameter selection

Hyper-parameters play an important role in getting optimal results. The key hyper-parameters are epoch, batch size, optimization algorithm, learning rate of the optimization algorithm. Several hyper-parameters were explored before discovering those which yielded better performance in terms of convergence rate and accuracy.

The epoch used in this experiment is 10. This implies that during the training of each model, the entire dataset was processed 10 times. We noticed that after the 10th epoch, the model never improved in any significant way.

We used a batch size of 50. Using a batch size of 50 implies that there are 2000 batches per epoch since the training dataset has 100,000 images. Choosing 50 as batch size was largely influenced by memory constraint. We noticed that using any batch size over 100 resulted in memory overflow. We did not notice any significant changes in performance by varying the batch size.

In the weights of the model, we employed the Adaptive Moment Estimation (Adam) optimization algorithm. Adam is particularly suitable for problems with a substantial number of trainable parameters, such as our own. It is also well-suited for handling sparse or noisy gradients, which frequently occur when the number of batches per epoch is large. An important hyper-parameter, the learning rate determines the convergence of the optimization algorithm. We experimented with various learning rates and determined that a value of 0.001 yielded optimal convergence.

B. Experimental results from the developed system

As stated in the research methodology, two architectures of CNN namely VGG19 and ResNet50 were modified by adding some custom layers to ensure they give a better performance and recognition accuracy when they are implemented. These modified architectures were implemented, and series of experimental results were obtained. These results were presented in the following section with their comparison to report the architecture with the best recognition performance.

i. Modified VGG19 results

The results obtained from the Modified VGG19 model for detecting deepfake images from real images show considerable promise. The model underwent evaluation on three distinct datasets: training, validation, and testing. The performance metrics employed for evaluation encompass Accuracy, Precision, Recall, F1 score, and AUC (Area Under the Curve). As illustrated in Table 1, the training dataset yielded an accuracy of 96.53%, precision of 97.57%, recall of 95.42%, F1 score of 96.49%, and AUC of 96.53%. These results reflect the model's adeptness in learning from the training data and its ability to correctly classify a significant proportion of deepfake images. The high precision score indicates a low false positive rate, meaning the model seldom misclassifies a real image as a deepfake. Although the recall score is slightly lower, it remains relatively high, signifying the model's capability to accurately identify most of the deepfake images in the dataset.

Also from Table 1, the validation dataset results show an accuracy of 91.43%, precision of 92.63%, recall of 90.00%, F1 score of 91.30%, and AUC of 91.43%. These results are slightly lower than the training results, which is to be expected as the model has not seen this data during training. However, the results are still quite high, indicating that the model is generalizing well and can correctly classify a

high percentage of deepfake images it has not seen before.

The test dataset results show an accuracy of 91.59%, precision of 92.79%, recall of 90.19%, F1 score of 91.47%, and AUC of 91.59%. These results are very similar to the validation results, further indicating that the model is generalizing well and can correctly classify a high percentage of unseen deepfake images.

The performance metrics results gotten for the training dataset are slightly higher than those for the validation and testing datasets. The expectation is that the model, having undergone thorough training on the data, would yield predictable results. Nonetheless, the disparity is minimal, implying that the model is not excessively tailored to the training data and is capable of effectively extrapolating to unfamiliar data.

The performance measures for both the validation and testing datasets exhibit remarkable similarity. This observation implies that the model's performance remains consistent across various datasets, indicating its potential to perform comparably on novel, unobserved data.

According to Figure 4, the accuracy of the dataset's training and validation is demonstrated over 15 epochs. Although the validation curve slightly exceeds the training curve, the variance remains within the realms of control and remains acceptable. The difference between the two accuracies were not up to 2%, this shows that our model was well trained without experiencing any over fitting. The training accuracy depict a good accuracy starting as the curve started from a lower accuracy compared to that of validation data, with these wider values were taken into consideration with the training dataset. This is among the factors that resulted in better average training accuracy than average validation accuracy as depicted in Table1. This can be supported with the high precision and recall values given by the training dataset.

.

Table 1: Experimentation results of modified VGG19 with various metrics

Metrics	Training Set	Validation Set	Test Set
Accuracy	0.96526	0.91425	0.91590
Precision	0.97575	0.92632	0.92788
Recall	0.95424	0.90010	0.90190
F1	0.96487	0.91302	0.91471
AUC	0.96526	0.91425	0.91590

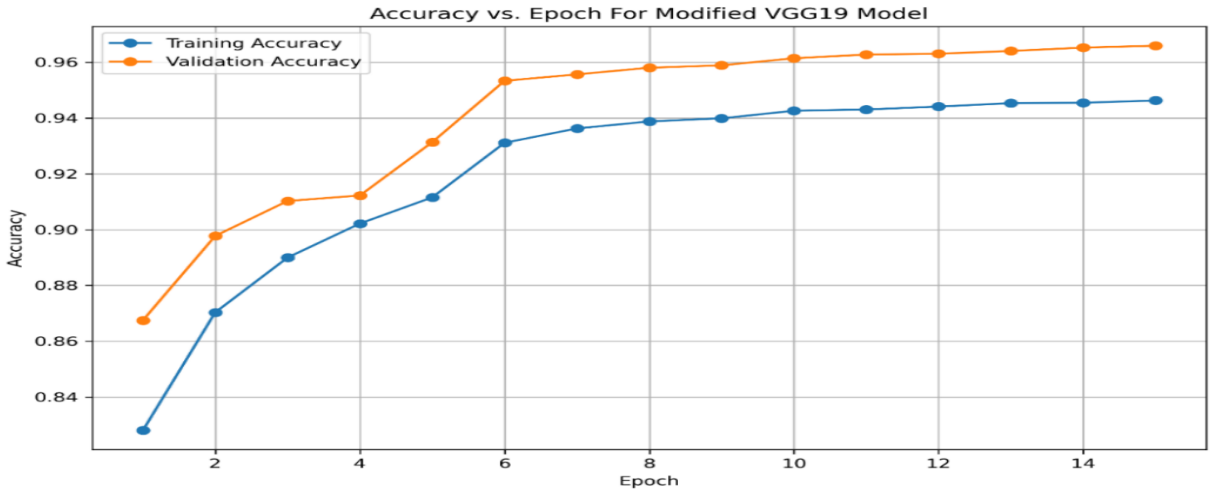

Figure 4: Modified VGG19 training and validation accuracies graph

As earlier denoted, the training and validation losses were plotted and represented by Figure 5, from this graph, the validation loss graph falls below that of the training loss, as in a normal training-validation loss relationship, the training loss ought to be lower than the validation loss. However the difference in that average validation and training loss is less than or equal 0.03. This shown difference is still within the acceptable difference as this does not imply that there is any under fitting due to the small loss difference resulting in lower validation los than the training loss.

Figure 6 shows that graphical relationship of the Training, Validation and Testing dataset in relation to their Trues and false positive values. As shown in the Figure the three datasets gave a very good true positive rates implying that there was little or no misclassification of the real and fake faces. The training dataset gave the best true positive rate as the rate was almost approaching perfect rate at the earliest stage. Figure 6 shows the good classification given by the datasets

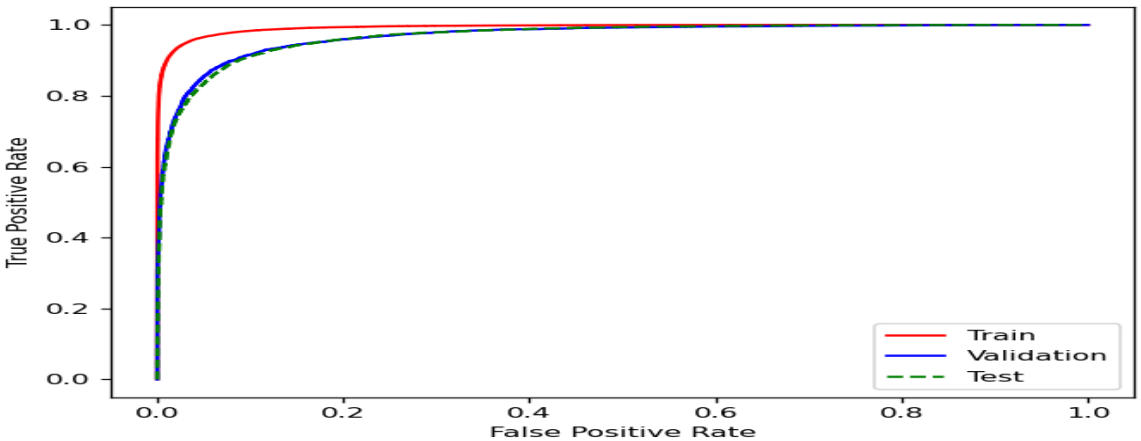

Figure 6: Modified VGG19 true and false positive rate raph

ii. Modified Resnet 50 results

The outcomes of the Modified Resnet 50 model, employed for the detection of deepfake images from real ones, exhibit great promise. The model was assessed using three distinct datasets: training, validation, and testing. The evaluation employed performance metrics such as Accuracy, Precision, Recall, F1 score, and AUC (Area Under the Curve). As illustrated in Table 4.2, the model achieved an accuracy of 99.26% on the training set, 96.83% on the validation set, and 96.61% on the test set. These figures indicate that the model performs admirably, correctly classifying a significant proportion of images as either real or deepfake. The model's precision is also commendable across all datasets, registering values of 99.53% for the training set, 97.21% for the validation set, and 97.57% for the test set. This suggests that the model is efficient in avoiding false classification of genuine images as deepfakes.

The model's recall is slightly lower, with values of 98.98% for the training set, 96.05% for the validation set, and 95.97% for the test set. This implies that the model is somewhat less effective in identifying all deepfake images within the datasets. The F1 score, which gauges the model's equilibrium between precision and recall, is high across all datasets, registering values of 99.26% for the training set, 96.81% for the validation set, and 96.59% for the test set. This indicates that the model strikes a fine balance between precision and recall. The AUC, which measures the model's capacity to differentiate between positive and negative classes, is also substantial across all datasets, with values of 99.26% for the training set, 96.83% for the validation set, and 96.61% for the test set. This implies that the model excels in distinguishing between real and deepfake images, as evidenced by the training AUC nearing 100% and the test set's convincing 96.61% AUC.

Table 2: Experimentation results of modified RestNet50 with various metrics

Metrics	Training Set	Validation Set	Test Set
Accuracy	0.9926	0.9683	0.9661
Precision	0.9953	0.9757	0.9721
Recall	0.9898	0.9605	0.9597
F1	0.9926	0.9681	0.9659
AUC	0.9926	0.9683	0.9661

The model's performance on the training set is marginally superior to that on the validation and test sets. This disparity is unsurprising, given that the model is trained on the training set and therefore possesses more information about this particular data. Nevertheless, the discrepancy in performance is not substantial, indicating that the model is not excessively tailored to the training data and exhibits a commendable ability to generalize to unseen data.

The performance on the validation and test sets is very similar, which indicates that the model is consistent in its predictions and is not affected by the specific split of the data.

Figure 7 represents the training and validation accuracies graph of the varying training and validations accuracies obtained while training the modified RestNet50 model. As shown in Figure 4.4, the validation accuracies are a bit bigger than the training accuracies but in the long run the average training accuracy was 99.23% while the average validation accuracy was 96.83% which implies that the training accuracy was later bigger than the validation accuracy resulting in normal training of the model without any over-fitting or resulting deficiency in the training. The difference in the accuracies throughout the 15 epoch is not up to 2%, which is actually very acceptable difference between the two accuracies.

Moreover, Figure 8 represent the training and validation losses as obtained during the training process of the modified ResNet50 model. The validation loss graph will be seen to be lower than the training loss's, however, the training loss started from higher loss values ensuring that all values were taken into consideration, and this helped in producing a better average training accuracy as presented earlier. Additionally, the difference in these losses were not that much as it is below 0.03.

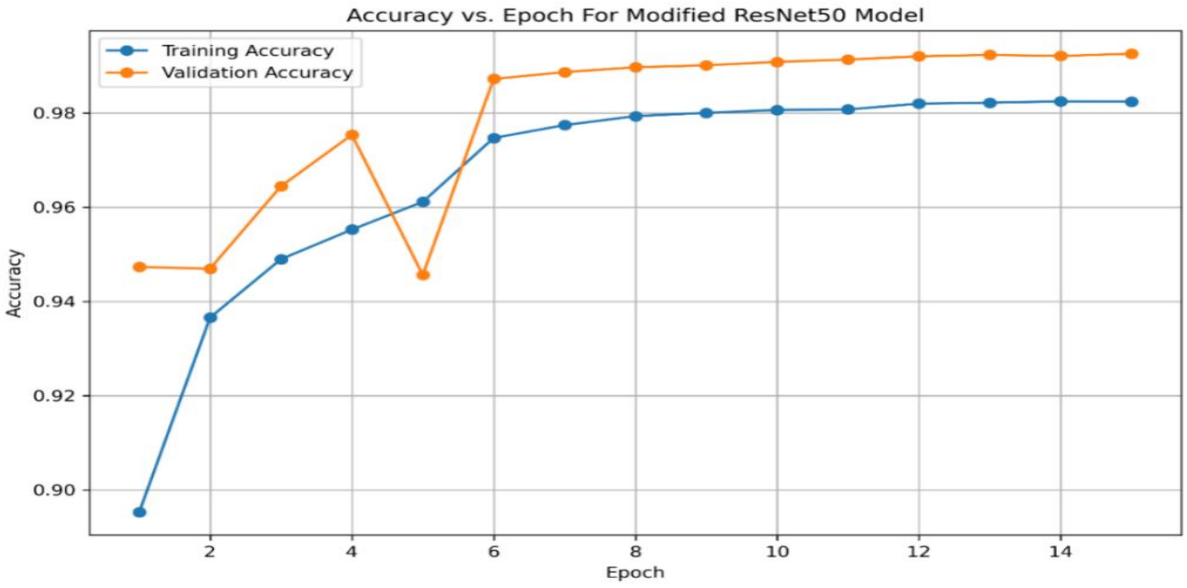

Figure 7: Modified ResNet50 training and validation accuracies graph

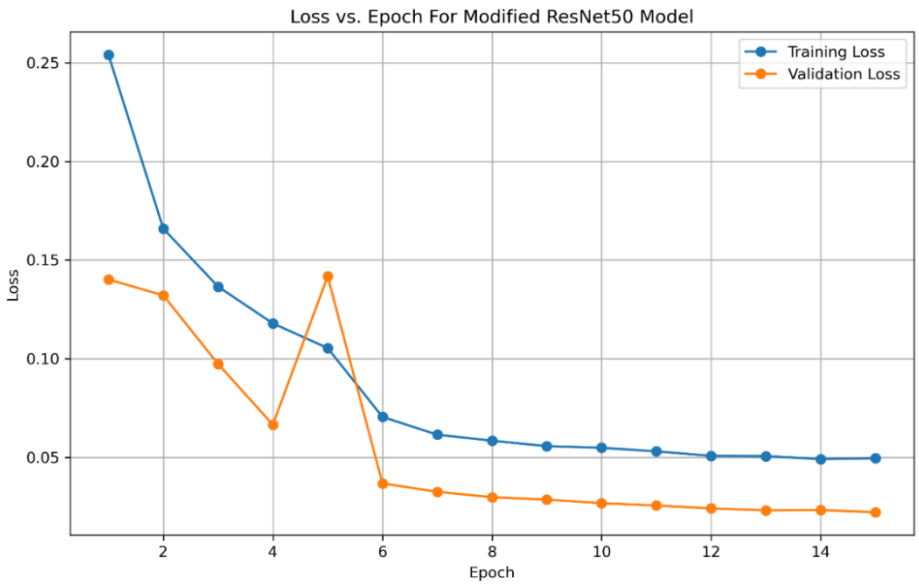

Figure 8: Modified ResNet50 training and validation losses graph

Figure 9 shows that graphical relationship of the Training, Validation and Testing dataset in relation to their Trues and false positive values obtained from the performance results of the Modified ResNet50. As shown in the Figure the three datasets gave a very good true positive rates implying that there was little or no misclassification of the real and fake faces. The training dataset gave the best true positive rate as the rate was almost approaching perfect rate at the earliest stage. Figure 9 shows the good classification given by the datasets.

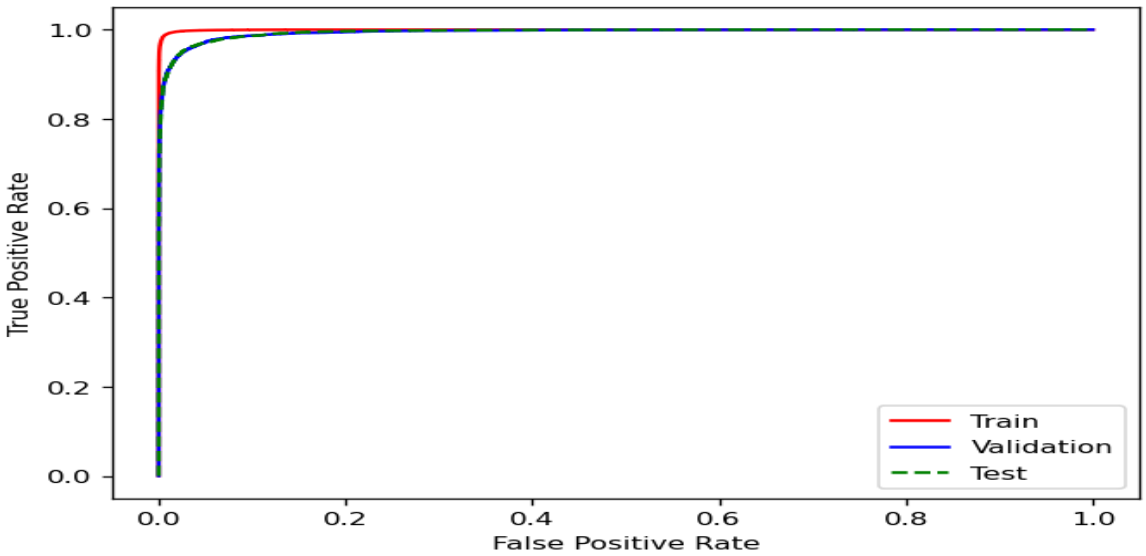

Figure 9: Modified ResNet50 true and false positive rate graph

iii. Results of the comparative Analysis of Modified VGG19 and ResNet50

The findings suggest that both the VGG19 and Resnet 50 models have exhibited commendable performance in the identification of deepfake images. Nevertheless, it is noticeable that the Resnet 50 model has surpassed the VGG19 model in all aspects and datasets.

In terms of accuracy, the Resnet 50 model demonstrates a superior accuracy rate across the three datasets (Training, Validation, and Test) when compared to the VGG19 model. The accuracy of the Resnet 50 model is recorded at 97.746% for the training set, 95% for the validation set, and 94.95% for the test set. Conversely, the VGG19 model achieves an accuracy of 95.46% for the training set, 90.71% for the validation set, and 90.195% for the test set.

The precision of the Resnet 50 model is also higher than that of the VGG19 model across all datasets. The Resnet 50 model exhibits a precision rate of 99.908% for the training set, 98.839% for the validation set, and 98.721% for the test set. In contrast, the VGG19 model demonstrates a precision rate of 98.088% for the training set, 93.559% for the validation set, and 92.948% for the test set.

Furthermore, the recall, F1 score, and AUC of the Resnet 50 model surpass those of the VGG19 model in all datasets. Based on the obtained results, it can be concluded that the Resnet 50 model is the superior choice for the detection of deepfake images. It exhibits higher levels of accuracy, precision, recall, F1 score, and AUC across all datasets. This suggests that the Resnet 50 model is more dependable and efficient in the identification of deepfake images. It strikes a better balance between precision (minimizing false positives) and recall (minimizing false negatives), as evidenced by its higher F1 score. Additionally, the higher AUC indicates that the Resnet 50 model achieves a better trade-off between sensitivity (true positive rate) and specificity (true negative rate). This is tabulated in Table 3 for simplicity and clarification.

Table 3: Result of the modified VGG19 and RestNet50 comparison results

Metrics	Training		Validation		Testing	
	VGG19	ResNet50	VGG19	ResNet50	VGG19	ResNet50
Accuracy	0.9653	0.9926	0.9143	0.9683	0.9159	0.9661
Precision	0.9757	0.9953	0.9263	0.9757	0.9279	0.9721
Recall	0.9542	0.9898	0.9001	0.9605	0.9019	0.9597
F1	0.9649	0.9926	0.9130	0.9681	0.9147	0.9659
AUC	0.9653	0.9926	0.9143	0.9683	0.9159	0.9661

iv. Results of the Comparison with the Existing Systems on Deepfake

As stated in the last objective of this research that the developed systems will be compared with other existing system that employed the same methodology for the detection and recognition of real and fake faces. The researcher conducted a comparison between the Area Under Curve (AUC) performance of the developed systems and an existing system that employed VGG16, RestNet50, and other cutting-edge techniques for real face recognition.

As shown in Table 4, not much research employed VGG19 and ResNet50 for real and fake face recognition, but some other state-of-art methods were compared like Neural Network, Meso-4 and MesoInception-4, VGG16 and RestNet50 employed by various researchers. The results gotten from high quality images employed by these researchers were compared with the performance results gotten from the developed system. As shown in Table 4, the developed system out-perform all other presented methods with the modified VGG19 and ResNet50 achieving the best performance.

From the result presented in Table 4, it shows that this study outperformed some of the existing studies for deepfake face detection conducted by many researchers, as the AUC given by this study outperformed other obtained from the compared studies even with those that employed the same methodology and dataset.

Table 4: Result of the comparison with the existing systems and state-of-art methods

Author	Method	AUC (%)
Peng *et al.* (2017)	Two stream NN	63.5
Daris *et al.* (2018)	Mesonet-4	68.4
Li and Lyu (2019)	VGG16	57.4
Li and Lyu (2019)	RestNet50	93.3
Developed system	Modified VGG19	91.59
Developed system	Modified ResNet50	96.61

IV. Conclusion

In conclusion, it can be observed that both the VGG19 and Resnet 50 models have exhibited commendable performance in detecting deepfake images subsequent to the modification of these two architectures. The modification entailed the replacement of the final five layers in the standard VGG19 with an additional four custom layers. Moreover, RestNet outperformed VGG-19 possibly because it relies on the use of Residual layer as it solves the problem of vanishing gradient using skip connected which is based on stacking of multiple identity mappings. With the result obtained from this study, impersonation using fake images generated by AI to constitute crime can be easily detected and such criminals can be prosecuted instead of causing harm to innocent souls. Future works should focus on employing different fake faces generated from different AI platforms that generates faces will help in catering for the peculiarities or limitations that each platform might be experiencing and employed deepfakes generated for black faces.

References

[1] Mirsky and Lee, W. "The Creation and Detection of Deepfakes: A Survey". *ACM Computing. Surveys, 2020. 1(1)*

[2] Somers, M. "Cybersecurity : Deepfakes, explained". *Ideas Made To Matter*. 2020. Retrieved on July 20, 2022 from https://mitsloan.mit.edu/ideas-made-to-matter/deepfakes-explained

[3] Agarwal, A.; Singh, R.; Vatsa, M and Noore. A. "SWAPPED! Digital face presentation attack detection via weighted local magnitude pattern". *In the proceedings of IEEE International Joint Conference on Biometrics (IJCB). 2017. IEEE, 659–665.*

[4] Smith, A. "Deepfakes are the most dangerous crime of the future, researchers say". *Independent UK Edition*. 2020. Retrieved on July 21, 2022 from https://www.independent.co.uk/tech/deepfakes-dangerous-crime-artificial-intelligence- a9655821.html

[5] Hao, K. "Tech Policy: Deepfake porn is running women's lives. Now the law may finally ban it". *MIT Technology Review, 2021*. Retrieved on July 21, 2022 from https://www.technologyreview.com/2021/02/12/1018222/deepfake-revenge-porn- coming-ban/

[6] Yang, X.; Li, Y. and Lyu, S. "Exposing Deep Fakes Using Inconsistent Head Poses". *arXiv preprint arXiv:1811.00661. 2018*

[7] Marra, F.; Gragnaniello, D.; Verdoliva, L. and Poggi, G. "Do GANs leave artificial fingerprints?"*arXiv preprint arXiv:1812.11842. 2018*

[8] Yu, N.; Davis, L. and Fritz, M. "Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints". *arXiv preprint arXiv:1811.08180. 2019*

[9] Andreas, R.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J. and Nießner, M. "Faceforensics: Learning to detect manipulated facial images". *In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019. pp. 1–11*

[10] Ciftci, U. A.; Demir, I. and Yin, L. "FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals" *IEEE Transactions on Pattern Analysis and Machine Intelligence, X(X),2020. 1-2.*

[11] Durall, R., Keuper, M., Pfreundt, F. J., and Keuper, J. "Unmasking DeepFakes with simple Features" *arXiv preprint arXiv:1911.00686. 2020*

[12] Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F. and Guo, B." Face X-ray for More General Face Forgery Detection" *arXiv preprint arXiv:1912.13458. 2020*

[13] Kumar, A., and Bhavsar, A. "Detecting Deepfakes with Metric Learning" *arXiv preprint arXiv:2003.08645. 2020*

[14] Guo, Z., Yang, G., Chen, J., and Sun, X. "Fake Face Detection via Adaptive Residuals Extraction Network". *arXiv preprint arXiv:2005.04945. 2020*

[15] De Lima, O., Franklin, S., Basu, S., Karwoski, B. and George, A. "Deepfake

Detection using Spatiotemporal Convolutional Networks" *arXiv preprint arXiv:2006.14749. 2020*

[16] Masi, I., Killekar, A., Mascarenhas, R. M., Gurudatt, S. P. and AbdAlmageed, W. "Two-branch Recurrent Network for Isolating Deepfakes in Videos". *arXiv preprint arXiv:2008.03412. 2020*

[17] Nirkin, Y., Wolf, L., Keller, Y. and Hassner, T. "DeepFake Detection Based on Discrepancies Between Faces and their Context". *arXiv preprint arXiv:2008.12262. 2020*

[18] Wodajo, D. and Atnafu, S. "Deepfake Video Detection Using Convolutional Vision Transformer" *ArXiv preprint arXiv:2102.11126. 2021.*

[19] Gragnaniello, D., Cozzolino, D., Marra, F., Poggi, G., & Verdoliva, L."Are GAN Generated Images Easy to Detect? A Critical Analysis of the State-of-the-Art". *arXiv preprint arXiv:2104.02617. 2021.*

[20] Sudha V., Ganeshbabu T. "Convolution Neural Network C lassifier VGG-19 Architecture for Lesion Detection and Grading in Dia betic Retinopathy Based on Deep Learning". *Computers, Materials & Continua, 2020. DOI:10.32604/cmc.2020.012008*