Using Deep Learning to Recognize Fake Faces

Jaffar Atwan[1], Mohammad Wedyan[2], Dheeb Albashish[3], Elaf Aljaafrah[4], Ryan Alturki[5], Bandar Alshawi[6]
Prince Abdullah bin Ghazi Faculty of Information and Communication Technology
Al-Balqa Applied University, Jordan[1,3,4]
Department of Computer Sciences-Faculty of Information Technology and Computer Science
Yarmouk University, Irbid, 21163, Jordan[2]
Department of Software Engineering-College of Computing
Umm Al-Qura University, Makkah, Saudi Arabia[5]
Department of Computer and Network Engineering-College of Computing
Umm Al-Qura University, Makkah, Saudi Arabia[6]

***Abstract*—In recent times, many fake faces have been created using deep learning and machine learning. Most fake faces made with deep learning are referred to as "deepfake photos." Our study's primary goal is to propose a useful framework for recognizing deep-fake photos using deep learning and transformative learning techniques. This paper proposed convolutional neural network (CNN) models based on deep transfer learning methodologies in which the designed classifier using global average pooling (GAP), dropout, and a dense layer with two neurons that use SoftMax are substituted for the final fully connected layer in the pretrained models. DenseNet201, the suggested framework, produced the best accuracy of 86.85% for both the deepfake and real picture datasets, while MobileNet produced a lower accuracy of 82.78%. The obtained experimental results showed that the proposed method outperformed other state-of-the-art fake picture discriminators in terms of performance. The proposed architecture helps cybersecurity specialists fight deepfake-related cybercrimes.**

***Keywords*—Deep learning; machine learning; deepfake; convolutional neural network; global average pooling**

I. Introduction

Artificial intelligence (AI) is the process of creating devices that mimic human intelligence in terms of behaviour and thought. The term can also refer to any device exhibiting characteristics of the human mind, like problem-solving and learning [1]. An ideal attribute of AI is the capacity to simplify and carry out actions that are most likely to achieve a specific goal. A subset of AI is machine learning (ML). Massive volumes of unstructured data, including text, photos, and videos, are ingested by deep-learning algorithms to allow this autonomous learning. ML aims to replicate how humans learn and increases in accuracy over time by using data and algorithms [2, 3].

ML is a crucial component of the developing field of data science. Algorithms are trained in data-mining projects to categorize, forecast, and unearth important insights using statistical techniques. With the goal of influencing important growth metrics, these insights drive decisions within applications and organizations [4]. As big data continues to grow and improve, data scientists will become increasingly in demand. It should be possible to use ML to find the information needed to answer many important business questions. Deep learning can be classified as a subset of machine learning [5]. Deep learning uses less complex concepts than those employed in ML and uses artificial neural networks that are designed to imitate human brain networks. Previously, the intricacy of neural networks has been limited by computer power. Larger and more complex neural networks are now conceivable due to advancements in big data analytics, which allow computers to see, learn from, and respond to complex events more quickly than people can. Deep learning makes it possible to categorize images, identify faces, translate languages, recognize audio, and determine whether a face is real or fake. It can tackle pattern recognition issues and does not require human intervention [6, 7].

The face is a person's most recognizable feature. The security hazards of facial modification are becoming increasingly more significant because of the rapid development of facial synthesis technology. Several algorithms based on deep-learning techniques can replace one person's face with another person's realistic-looking visage [8]. Additionally, new AI technology called deepfake combines the faces of two different people. A number of methods based on generative adversarial networks (GANs) produce high-resolution deepfake images that are more accurate than previous technologies [9]. This is cause for concern, as deepfake information can circulate quickly due to the rise of mobile phones and the emergence of multiple social networking sites [10]. Initially, deepfake photos could be distinguished by the human eye because of a pixel collapse phenomenon that tends to produce unnatural visual contrasts in skin tones and face features. However, over time and with the development of technology, deepfakes have essentially merged with natural imagery [11].

Deepfake techniques frequently require enormous volumes of audio, video, or image data to produce convincing photographs that look natural. However, while deepfakes represent huge development in technological capability, there are some negatives. There is a prevalence of deepfakes of public people, including athletes, politicians, and celebrities, in the abundance of films and photos that can be found online [12]. Additionally, deepfake technologies can be used to ridicule and humiliate people. Deepfakes are considered to be the most harmful sort of synthetic media. They utilize celebrities' voices and photos without their permission to make political or humorous content about them. Due to the simplicity of the numerous applications making deepfakes, anyone can use this technology to make artificial content that is indistinguishable from actual content. It is not only public people who can be affected by deepfake

technology. One use of deepfake content is cyberbullying, which affects a large population of young people [10]. A number of factors are taken into account in the sophisticated approach of deepfake image detection. The basic steps of image classification include identifying a suitable classification scheme, collecting training patterns, image pre-processing, feature extraction, choosing a suitable evaluation method, and evaluating accuracy.

The remainder of the essay is structured as follows. Section II provides background on deepfakes, GANs, and a summary of a range of studies and previous research on image classification. Section III focuses on research procedures and methods of work. It includes a detailed explanation of the models used. Section IV presents an experimental setup. Section V describes the results of the experiment obtained using the selected dataset on a set of models and makes a comparison between them based on several criteria. Finally, Section VI presents conclusions and suggestions for further work.

II. Literature Review

The development of technology has made life easier in many respects. However, there have also been instances where technology has been abused, which has resulted in some serious issues. One example is digital image technology. There are many tools and software that make it is easy to modify any digital image. For example, anyone with even a basic understanding of Photoshop can quickly and simply create a fake image of another person [13].

There has been a lot of recent research on the use of these kinds of forgeries. Advancements in the disciplines of AI means that people may now alter a raw image and use it in both positive and harmful ways because, crucially, these techniques can provide incredibly life-like outcomes. This introduced us to the realm of deepfake pictures [14]. For example, [15] uses deep learning as a technology that creates face recognition and can determine whether a profile image is authentic or not, with the aim of finding a reliable method to distinguish between actual and phony. This study included real and fake face detection utilizing deep learning methods built on neural networks in two image datasets. They chose the ResNet50 model as the best match and used a trained dataset of 9,000 photos. The training accuracy was 99.18%. The research in [16], transfer learning methods from previously trained depth models like ResNet50 and VGG16 were used in the proposed model and three benchmark datasets were used to assess the proposed model. The findings collected demonstrate that the suggested model outperforms the current models. The study in [17] used enhanced datasets for real and fake face identification to compare the most popular modern face-recognition classifiers, including Custom convolutional neural network (CNN), VGG19, and DenseNet-121. They found performance can be increased while using fewer computational resources due to data augmentation. According to the authors preliminary findings, VGG19 outperforms all other examined models and has a maximum accuracy of 95%. To create ensemble-like multi-attention networks for detecting deep fake media, this work attempts to provide a complete examination of the mentioned methods, structures, and mechanisms.

The research in [18] attempts to address the difficulty of differentiating between real and fake pictures by developing an algorithm that can distinguish between real and fake pictures. The algorithm used in [18] seeks to differentiate between real images and deep fakes. The dataset was tested against five transfer learning methods as well as an 18-layered bespoke CNN model that was described in the research. The proposed model was able to test with an accuracy of 98.77%, whereas InceptionV3 produced the best results of the transfer learning models with a testing accuracy of 97.10%. Comparing deepfake and real photos, the unique CNN model performed better than any other model previously employed. The main goal of [19] was to develop a reliable and accurate method for recognizing deepfake images. The significance of this work lies in obtaining positive outcomes while utilizing the CNN architecture. This study employed eight CNN architectures to identify deep-fake images from big datasets. The findings were accurate and dependable. For some criteria, like F1 score, precision, and area under the Receiver operating characteristic ROC curve, the custom model used in this investigation performed marginally better than VGG Face in terms of recall.

The research in [20] provided a pipeline for categorizing and recognizing human faces from input visual samples. The second stage employed a number of deep learning (DL)-based techniques to calculate deep features from the returned faces. A support vector machine (SVM), a type of classifier, was trained on these characteristics to assess whether the data was real or fake. They compared the performance of numerous feature extractors based on their published results and found that DenseNet169 and its SVM classifier surpassed the competition. Table I summarizes the previously mentioned studies.

III. Materials and Methods

In order to detect fake faces, this work builds a group of pre-trained models with fine-tuning. A final choice is made for a testing image by fine-tuning five pre-trained models (DenseNet201, MobileNet, InceptionV3, ResNet50, and Xception) and fusing their projected probabilities. The pre-trained models use transfer learning to reduce their weights so that they can perform a similar classification problem. For the classification of faces, ensemble learning of previously trained models achieves greater results.

A. Pretrained Dense Net

A variation on the ResNet design is the densely linked convolutional network (DenseNet) architecture suggested by [21]. In this architecture, layers are connected to one another using the summation technique. In comparison to the ResNet design, the summing operation aids in further improving generalization ability and better resolving the issue of the vanishing gradient. The features that are taken from each layer are used as input for the following layers in this method. Reusing feature maps could help the overall performance be improved even further. The architecture of DenseNet201 contains 201 layers, hence the name. In this paradigm, high performance can be attained with little memory and little computational expense. DenseNet comes in a variety of sizes, including 121, 169, 201, and 264.

B. Pretrained MobileNet

The Google research team created the MobileNet architecture [22] for object identification on portable devices.

TABLE I. SUMMARY OF THE MOST IMPORTANT CLASSIFICATION STUDIES ON FAKE FACES

Authors	Dataset used	CNN architectures
Maher Salman et al. [15]	Real and fake faces detection	VGG16, ResNet50 InceptionV3, MobileNet
Taeb et al. [17]	Real and fake face detection 140K real and fake faces	VGG19, DenseNet121
Sharma et al. [16]	140k real and fake faces Fakefaces Real and fake face detection	VGG16, ResNet50
Dhar [18]	140K real and fake faces	VGG16, DenseNet121 InceptionV3, VGG19, ResNet50
Shad et al. [19]	140K real and fake faces	DenseNet201, DenseNet169, ResNet50, VGG16, VGG19, VGGFace
Masood et al. [20]	DeepFake Detection Challenge (DFDC)	VGG16, VGG19, ResNet101, Inception V3, DenseNet-169, InceptionResV2, XceptionNet, MobileNetv2, EfficientNet, NASNetMobile

MobileNet architecture presented a depth-wise separable convolution along with 11 point-wise convolution layers, having 32 times fewer parameters compared to conventional convolutions. MobileNet architecture outperformed VGG16 achieving higher accuracy during training on ImageNet dataset and requiring 27 times less computational power. Through depth-wise convolution, one depth-wise kernel was employed all the input channel. Point-wise convolution utilizes 11-bit kernel size CONV layer to calculate a linear combination of several input channels. The preceding method reduces the feature maps dimensionality significantly.

C. Pretrained Resnet

The ResNet50 network has a lot of depth. With it, more complicated networks can be constructed (which might refer to as networks inside networks) utilizing common network components known as residual modules and train them using stochastic gradient descent (SGD). The ResNet architecture [23] was groundbreaking work that demonstrated how residual modules can be used to train very deep networks using regular SGD. By applying identity-mapping techniques to update the residual coefficients, accuracy can be attained. Its architecture drastically reduces its size by using a global average pooling layer rather than a fully linked layer. This network is called ResNet50 because the architecture has 50 levels.

D. Pretrained Xception

Xception architecture, which stands for extreme inception and was introduced by François Chollet [24], is an improvement on the Inception design. In this architecture, the initialization modules from the Inception design are replaced by residual connections and depth-wise separable convolutions. It is possible that the depth-wise separable convolution will lower memory and processing expenses. The Xception architecture consists of 14 modules, each with 36 convolutional layers. All connections between modules, except for the first and last, are created via linear residual connections.

E. Pretrained Inception

The third iteration of the Inception model, the Inception V3 architecture [24] has a total of 159 layers. Instead of utilizing a single kernel size (such as 3x3 or 5x5), the Inception module uses several convolution sizes, such as 1x1, 3x3, and 5x5 filter sizes. The fundamental concept behind using various convolution sizes is that it enables the extraction of multi-level characteristics from the input image during each convolution process. Pointwise 11 convolution is also employed in this architecture to cut down on the number of parameters. The computational cost is decreased by the pointwise convolutions. The network has undergone numerous iterations due to its ongoing evolution. InceptionV1, InceptionV2, InceptionV3, InceptionV4, and Inception-Resent are common variants. Table II shows the summary of the deep architectures employed in this study.

F. Experimental Design

The proposed method for identifying fake or real faces based on the CNN architecture is described in this section. By using five different models, this study attempts to create a deep-learning model for face classification. The entire workflow of suggested solution is depicted in Fig. 1. The diagram illustrates the three basic steps of the model. The first phase is loading the dataset and image processing, the second is using the pre-trained model to extract features, and the third is using the selected features and classifying images. The proposed model uses datasets as input, and the final output is to classify images and evaluate and visualize the results.

Five different deep learning models – ResNet50, Inception V3, DenseNet201, Xception, and MobileNet – have been used as the base models and pre-trained for classification using the ImageNet dataset. An approach called transfer learning is used to train these models. In transfer learning, a pre-trained network performs better than a network that was trained from scratch. As shown, constructing classification solutions with transfer learning is quicker and more effective than doing it without. CNN also plays a fascinating role in classification. Two components make up each model: a feature extractor and a classifier. The classifier is used to categorize the collected features, whereas the feature extractor works to extract features using a convolutional base layer. In order to determine if the output is a fake face or a real face, The convolutional base layers and adapt the final classification layer are kept by adding new sets of layers such as global average pooling (GAP), dropout, and the dense layer.

IV. EXPERIMENTAL SETUP

A. Datasets

The proposed model on a deepfake and real images dataset acquired from the Kaggle website is tested. https://www.kaggle.com/datasets/manjilkarki/deepfake-and-real-images. Five CNN models were trained to distinguish between fake and real images. The dataset is divided into a training set and a test set. The training set has 4,700 images, of which 2,500 are real, and the rest are fake. The testing set has 540 images, of which

TABLE II. A SUMMARY OF THE DEEP ARCHITECTURES EMPLOYED IN THIS STUDY

Architecture	Convolutional layer count	Count of face centred cubic (FCC) layers	Parameter count for training
DenseNet201	199	2	20.2 million
MobileNet	53	3	3.4 million
ResNet50	48	2	25.6 million
Xception	70	1	22.9 million
InceptionV3	42	1	23.9 million

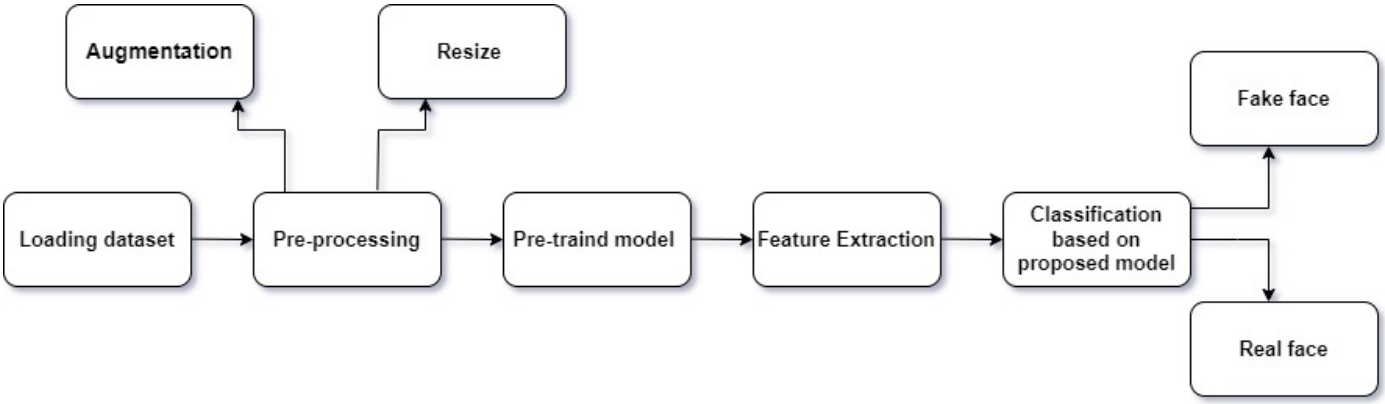

Fig. 1. Proposed experimental design.

300 are real and the rest are fake. Both real and altered photos can be found in this dataset. The faces, which are produced using a variety of techniques, are modified images. To extract the most value from these photos, this dataset was processed. Each picture is a 256x256 jpg image of a real or fake human face.

B. Pre-processing

The most important part of the model is the pre-processing method. To minimize overfitting, data augmentation was implemented. A 224x224x3 image was provided as the last input for the recommended model. Image augmentation is the process of creating new training samples from existing ones. To create a new pattern, slightly modify the original image; for example, you can make the new image slightly brighter or crop part of the original image. The original image can be mirrored to produce a new one [25]. There are various techniques to help increase the number of data points, such as rotation, shift, zooming, and horizontal fling. Augmented datasets were used for these experiments. To make the expanded dataset better fit the trained models, and they were scaled it and added horizontal flips by added a shifting of 0.1, a zoom range of 0.5, and a 45-degree rotation to the datasets.

C. Extraction of Features

In the feature extraction approach, the network of convolutional and pooling layers that serve as the extraction of features were kept while removing the fully connected layers of a pre-trained CNN model. The feature extractor can be expanded with fully linked layers and machine-learning classifiers. As a result of the dataset being more appropriate for this model, the network's performance on it is improved. Also, the final fully connected layer and retrieved features were kept with the trained models ResNet50, Inception V3, DenseNet201, Xception, and MobileNet.

D. Classification

Deep features were extracted and sent through the ResNet50, Inception V3, DenseNet201, Xception, and MobileNet models before being transferred to user-specific layers that were specifically designed for them. Deep features that had been concatenated were scaled in one-dimension (1D) form using GAP, producing feature maps that were appropriate for the succeeding two completely connected layers. Two fully connected layers and introduced dropout (0.5) in the midst of the fully connected layers were used to improve efficiency and generalize learning. The activation function and the output are ultimately produced by a dense layer with two neurons that uses the SoftMax activation function for binary classification.

E. Evaluation Criteria

In this study, the TensorFlow package, Keras API, and Python programming were used to implement all the pre-trained models (DenseNet201, MobileNet, ResNet50, Xception, and Inception V3). Additionally, Google Colab Pro was used for all tests. The model is trained and optimized using the Adam optimizer. A cycle of updating network weights using all the training data is known as an epoch. A model's performance will advance over time as the number of epochs rises. All models were tested across 25 epochs with a learning rate of 0.001 and a batch size of 32. Dropout was introduced to expedite training, enhance learning, boost precision, and avoid overfitting. The inputs used to train the model are shown in the Table III.

1) Accuracy: The percentage of correctly categorized images is what is meant by accuracy. TP + TN / (TP + TN + FP + FN).
2) Precision: It is the proportion of positively anticipated categories to positively classed categories that were effectively recognized. TP/ (TP + FP).

TABLE III. HYPERPARAMETERS USED IN THE SUGGESTED TRANSFER LEARNING MODELS

Hyperparameters	Value
Image size	224 x 224
Optimizer	Adam
Learning rate	0.001
Batch size	32
Dropout	0.5
Number of epochs	25
Activation function	SoftMax

3) Recall: The recall rate is the proportion of subjects who were correctly classified out of all positively classified subjects. TP / (FN + TP).
4) F1 score: The F1 score is typically employed to make it possible to measure both precision and recall simultaneously. The harmonic mean is used in place of the arithmetic mean. As a result, the penalize extreme values more. 2*(precision*recall)/ (precision + recall)

V. THE RESULTS

The proposed methods were used to test out a number of pre-trained deep-learning models that were available. Performance in various models was enhanced using different optimizers. A number of CNN models (see Table IV) were implemented using the deepfake and real images dataset. This demonstrated good facial image classification accuracy. Additionally, the figures of each proposed model were shown and explained using the dataset. The automatic identification and classification of faces are presented in depth in this part, along with the results of the studies. To create a reliable classifier, numerous trials were carried out with list models, InceptionV3, DenseNet201, MobileNet, ResNet50, and Xception. This study's primary objective is to evaluate the effectiveness of deep learning architectures. On the basis of performance metrics for precision, recall, and F1 score, the five designs employed in the study were assessed. The experimental results attained for each model are shown in Table IV. The results table show that the classifier performs well for each class.

Table IV and Fig. 2 show the results for the accuracy, precision, and F1-score recall of the deep fake and real images dataset, which includes two classes of fake faces and real faces using five of the pre-training models with optimizer Adam, anumber of epochs of 25 for each model with a SoftMax activation function, and a batch size of 32. The model that achieved the highest accuracy was DenseNet201, with a rate of 86.58% and the highest recall of 0.86, a precision of 0.87, and an F1 score of 0.87, while ResNet50 had an accuracy of 83.33%. The accuracy for Xception was 84.07% and, 85.0% for Inception V3. The MobileNet model provided relatively low accuracy, sensitivity, precision, and F1-score values for all classes.

Graph (A) from Fig. 2 shows the accuracy of the model DenseNet201 throughout training and validation over a period of 25 epochs. As the number of epochs rises, the accuracy of training and validation appears to increase. However, there are some variations in the validation accuracy over time. The validation accuracy fell below 65% in the first three epochs. However, the results approached a score of 86% by the 25th epoch, while the validation loss fluctuated, eventually falling to zero across the remaining epochs.

Fig. 3 (A) shows the accuracy of the model MobileNet throughout training and validation over a period of 25 epochs. As the number of epochs rises, the accuracy of training and validation appears to increase. However, there are some variations in the validation accuracy over time. The validation accuracy fell below 66% in the first 15 epochs. However, the results approached a score of 82% by the 25th epoch, while the validation loss fluctuated, eventually falling to zero across the remaining epochs.

Fig. 4 graph (A) shows the accuracy of the model ResNet50 throughout training and validation over a period of 25 epochs. As the number of epochs rises, the accuracy of training and validation appears to increase. However, there are some variations in the validation accuracy over time. The validation accuracy fell below 55 in the first five epochs. However, the results approached a score of 83% score by the 25th epoch, while the validation loss fluctuated, eventually falling to zero across the remaining epochs.

Fig. 5 graph (A) shows the accuracy of the model Xception throughout training and validation over a period of 25 epochs. As the number of epochs rises, the accuracy of training and validation appears to increase. However, there are some variations in the validation accuracy over time. The validation accuracy fell below 72% in the first 10 epochs. However, the results approached a score of 84% by the 25th epoch, while the validation loss fluctuated, eventually falling to zero across the remaining epochs.

Fig. 6 graph (A) shows the accuracy of the model InceptionV3 throughout training and validation over a period of 25 epochs. As the number of epochs rises, the accuracy of training and validation appears to increase. However, there are some variations in the validation accuracy over time. The validation accuracy fell below 68% in the first 15 epochs. However, the results approached a score of 85% by the 25th epoch, while the validation loss fluctuated, eventually falling to zero across the remaining epochs.

A. Performance Evaluation Metrics

There is a concept known as a confusion matrix in the context of machine learning, deep learning, and, more specifically, the issue of statistical classification. A table that summarizes how well a classification model works on a collection of test data or real values from the set is known as a confusion matrix. A result, the algorithm's performance can be assessed and commonalities between classes can be quickly found. In further detail, the confusion matrix is a clear account of the outcomes of a categorization task that contains a summary of the right and wrong predictions. The true negative (TN) condition occurs when the model predicts the negative class accurately. The negative type in this instance relates to an actual face. A false negative (FN) occurs when the model forecasts the negative class inaccurately and incorrectly predicts that the face was real. A false positive (FP) occurs when the model forecasts the positive class inaccurately; that is, it predicted the face to be a fake but it was incorrect. When the model accurately predicts the positive class, it is said to be a true

TABLE IV. THE EXPERIMENTAL RESULTS OBTAINED ON THE DEEPFAKE AND REAL IMAGES DATASET USING MODELS

Pretrained models	Accuracy	Precision	Recall	F1-score
DenseNet201	86.58%	0.87	0.86	0.87
MobileNet	82.78%	0.83	0.83	0.83
ResNet50	83.33%	0.83	0.83	0.83
Xception	84.07%	0.85	0.83	0.84
InceptionV3	85.00%	0.87	0.84	0.84

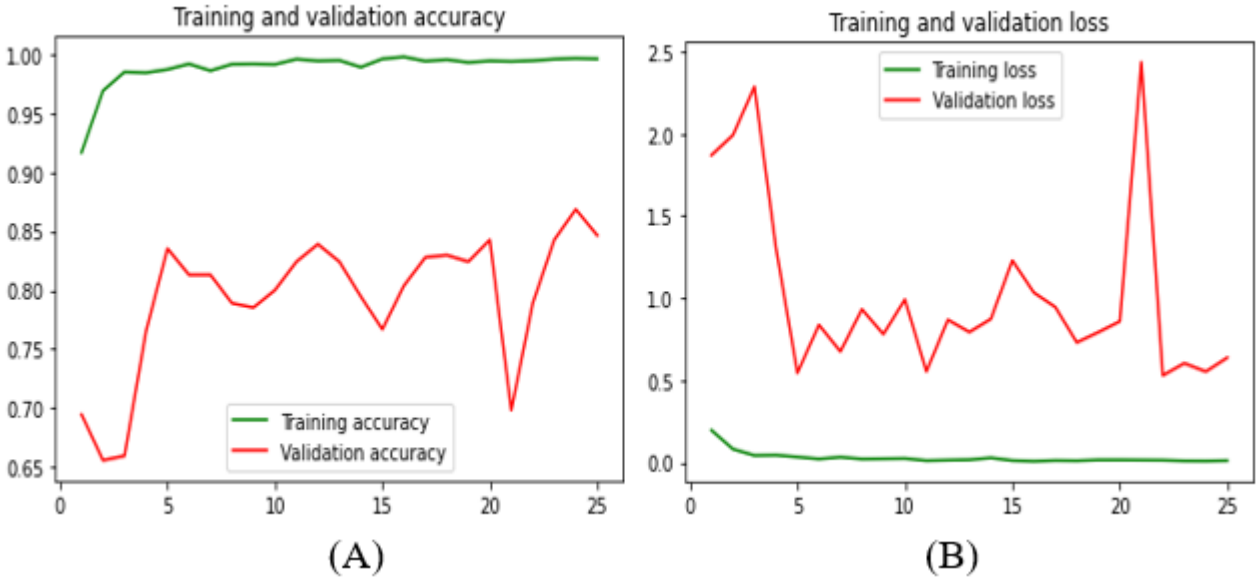

Fig. 2. The accuracy (A) and Loss (B) of the DenseNet201 model during training and validation.

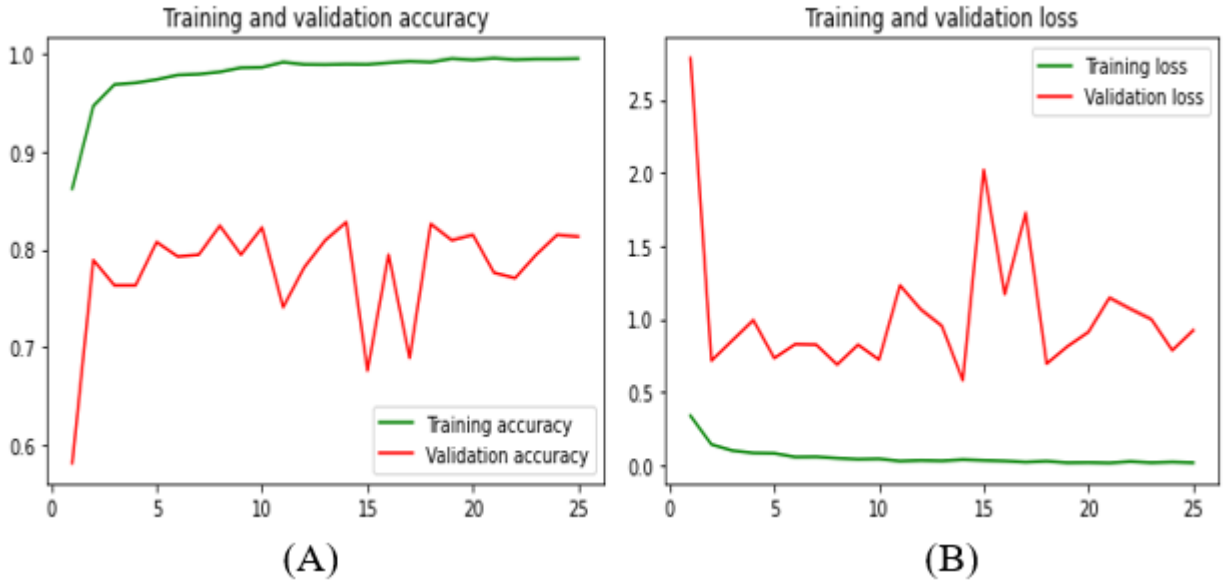

Fig. 3. The accuracy (A) and Loss (B) of the MobileNet model during training and validation.

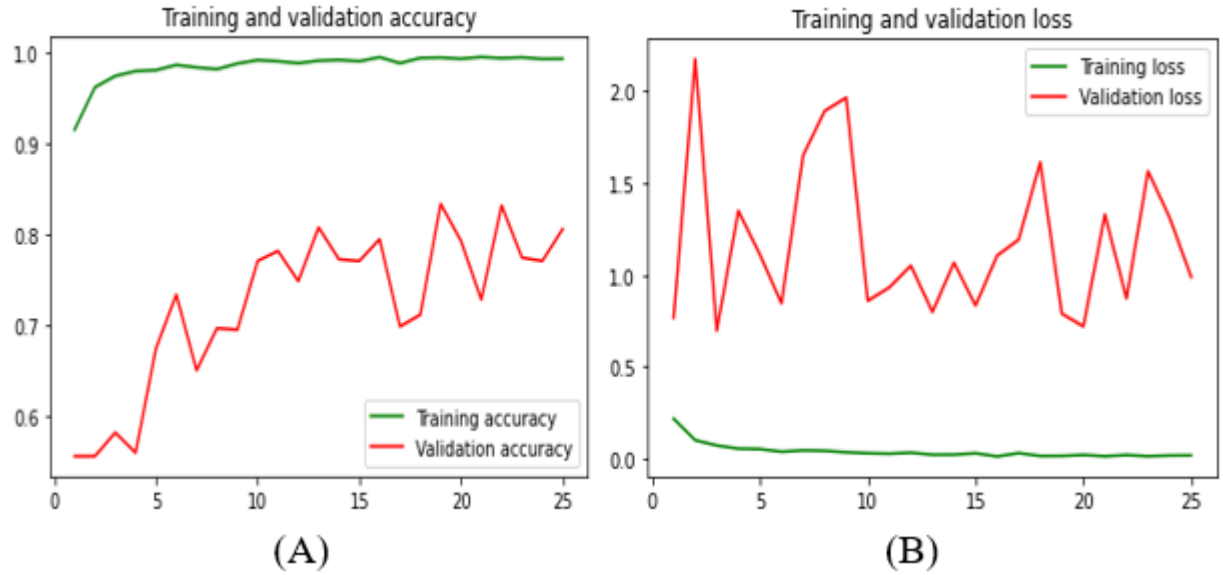

Fig. 4. The accuracy (A) and Loss (B) of the ResNet50 model during training and validation.

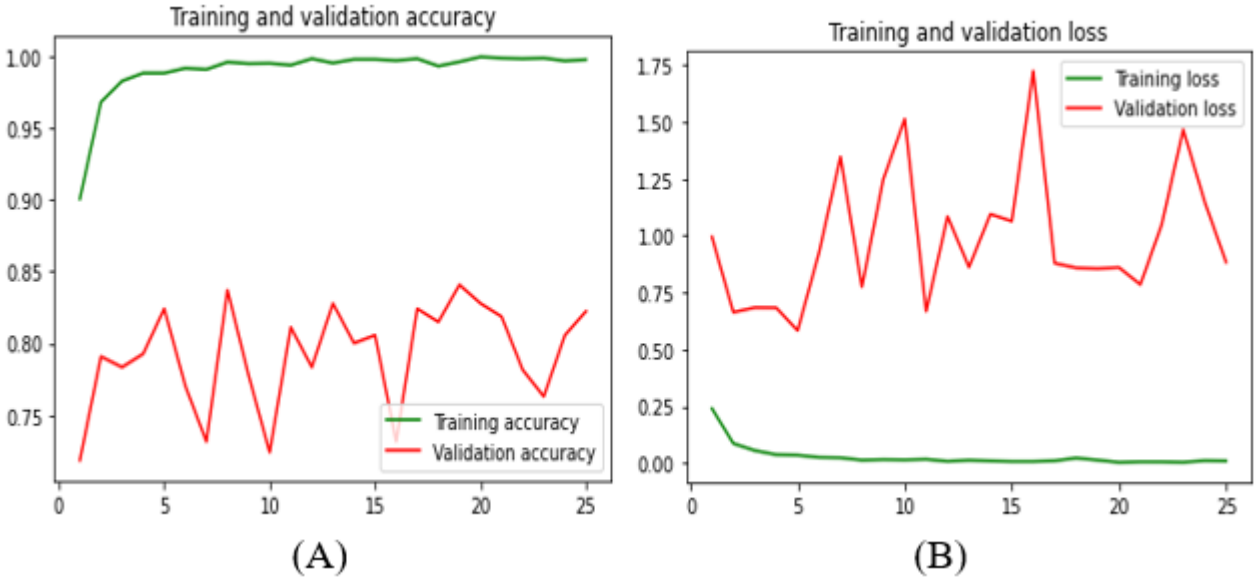

Fig. 5. The accuracy (A) and Loss (B) of the Xception model during training and validation.

positive (TP). The positive category in this instance refers to a fake face.

Fig. 7 displays the confusion matrix for the DenseNet201 model for the deepfake and real images dataset. The number of images is 540, divided into 240 fake images and 280 real images. Forty-five images were incorrectly labeled as fake when they were real faces and 26 images were real but incorrectly labeled as fake. Furthermore, 195 of the photographs were accurately identified as fake, while 274 of the images were correctly identified as real.

Fig. 8 displays the confusion matrix for the MobileNet model for the deepfake and real images dataset. The number of images is 540, divided into 240 fake images and 280 real images. Forty-five images were incorrectly labeled as fake when they were real faces and 48 images were real but incorrectly labeled as fake. Furthermore, 195 of the photographs were accurately identified as fake, while 252 of the images were correctly identified as real.

Fig. 9 displays the confusion matrix for the ResNet50 model for the deepfake and real images dataset. The number of images is 540, which were divided into 240 fake images and 280 real images. Forty-seven images were incorrectly labeled as fake when they were real faces and 43 images were real but incorrectly labeled as fake. Furthermore, 193 of the photographs were accurately identified as fake, while 257 of the images were correctly identified as real.

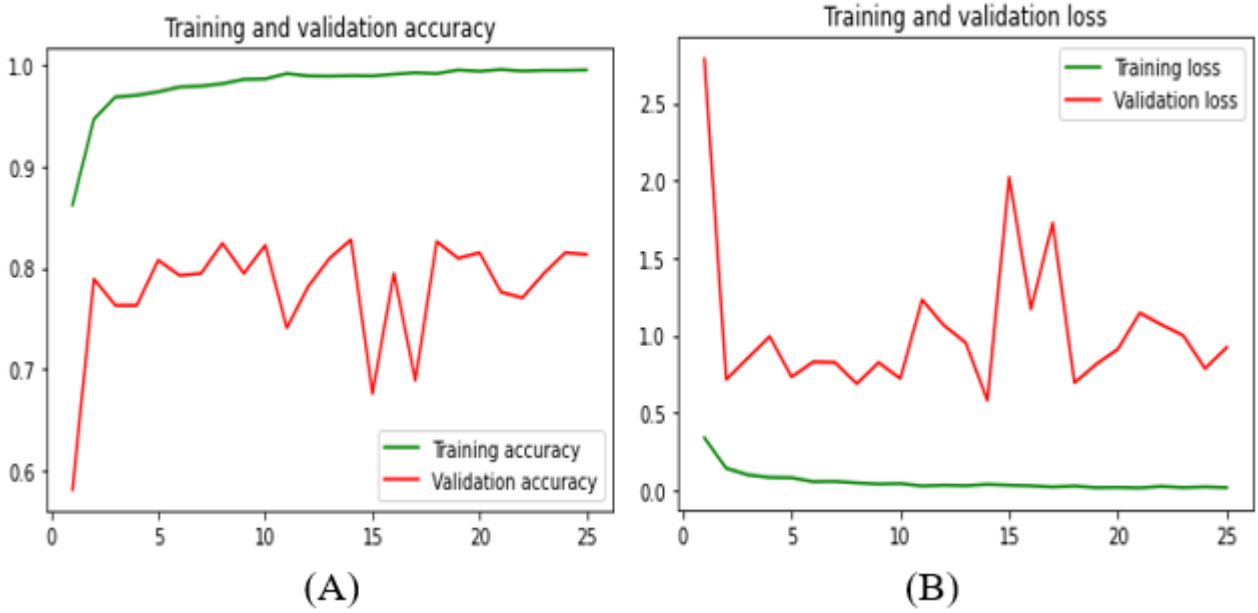

Fig. 6. The accuracy (A) and Loss (B) of the InceptionV3 model during training and validation.

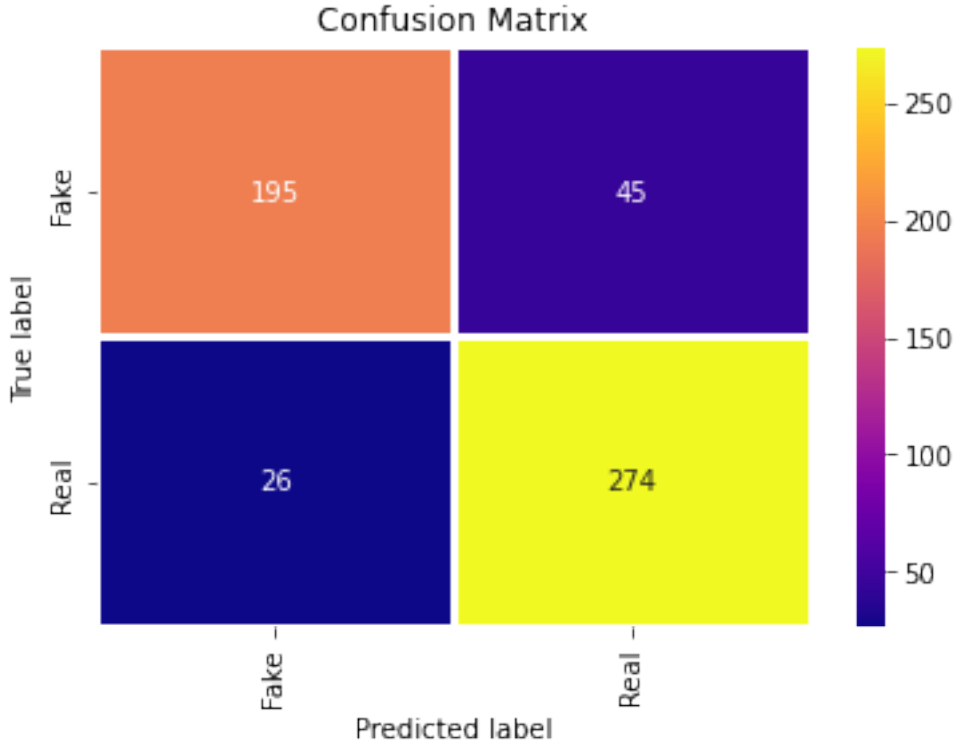

Fig. 7. The result of the prediction of the DenseNet201.

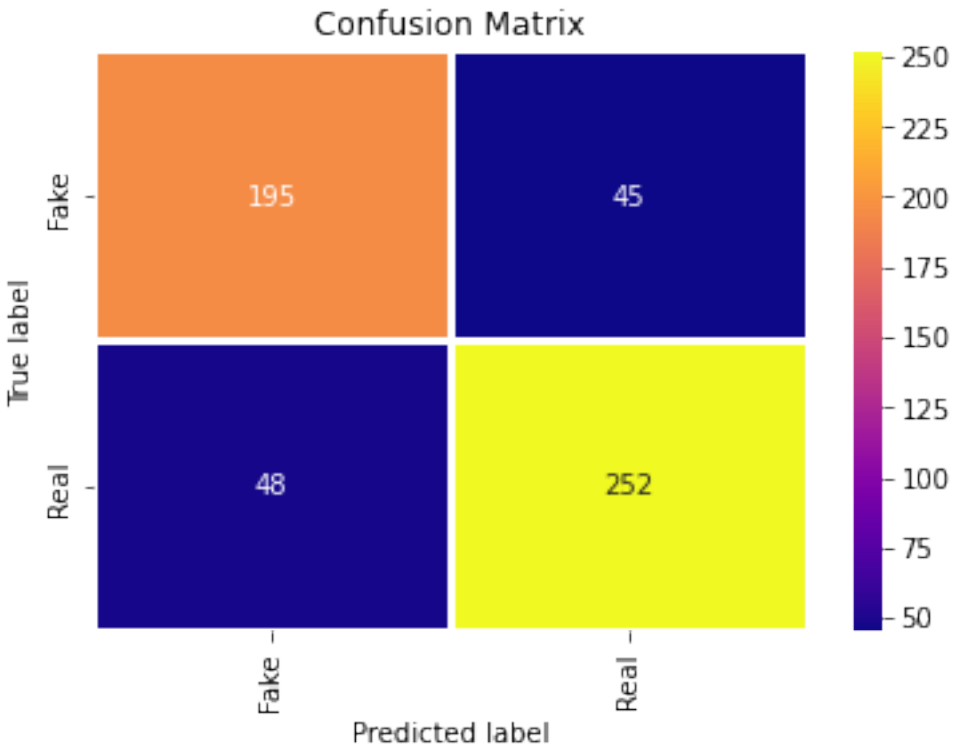

Fig. 8. The result of the prediction of the MobileNet.

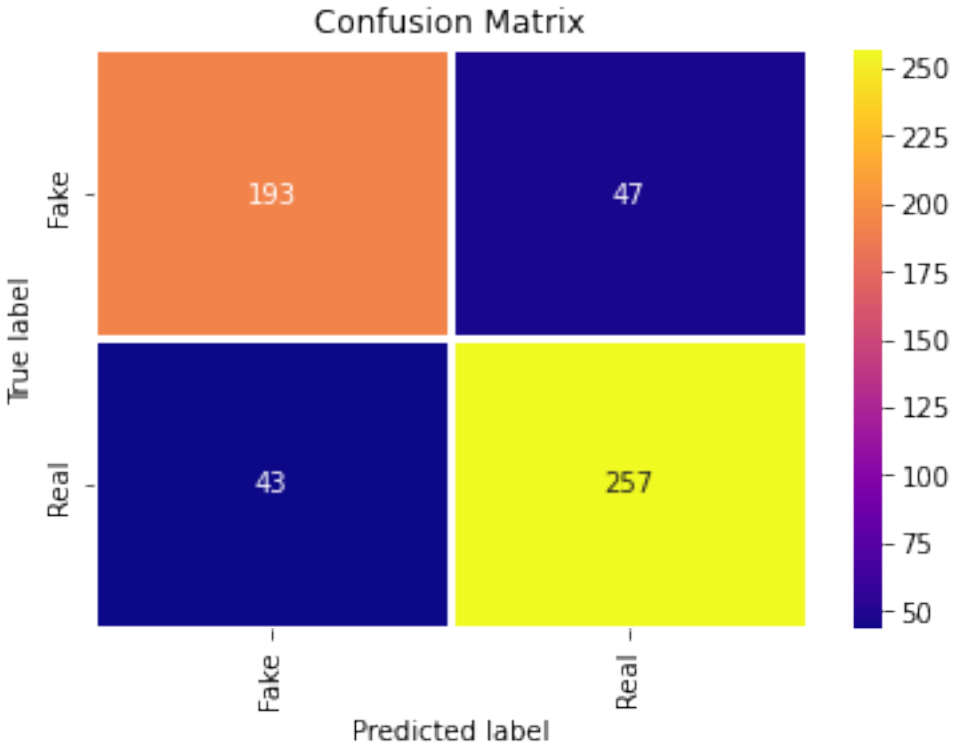

Fig. 9. The result of the prediction of the ResNet50.

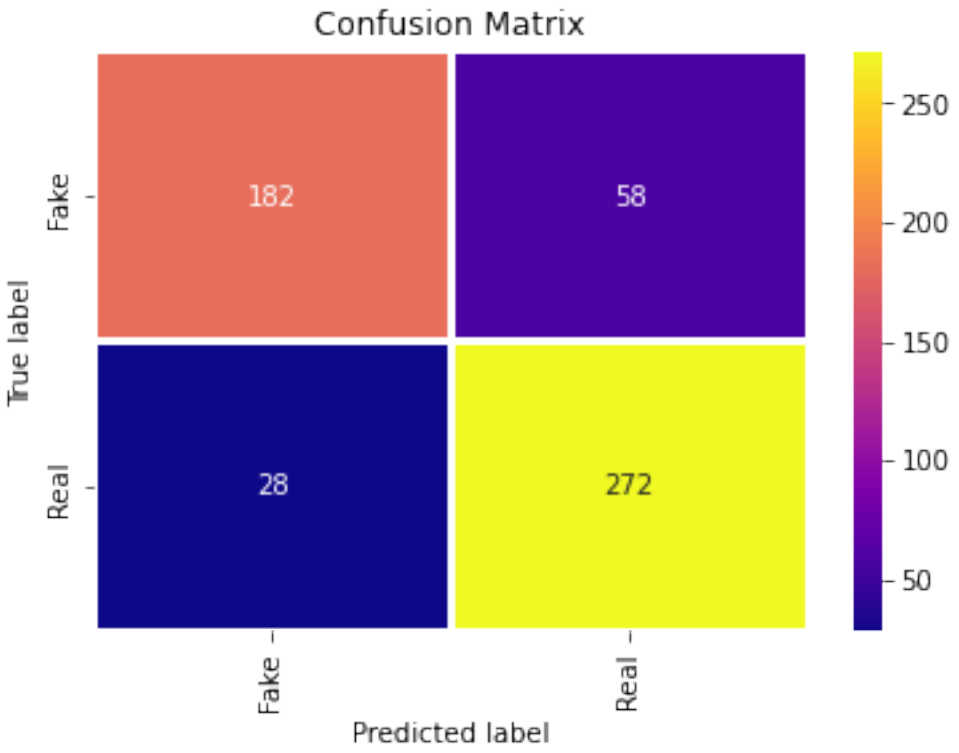

Fig. 10. The result of the prediction of the Xception.

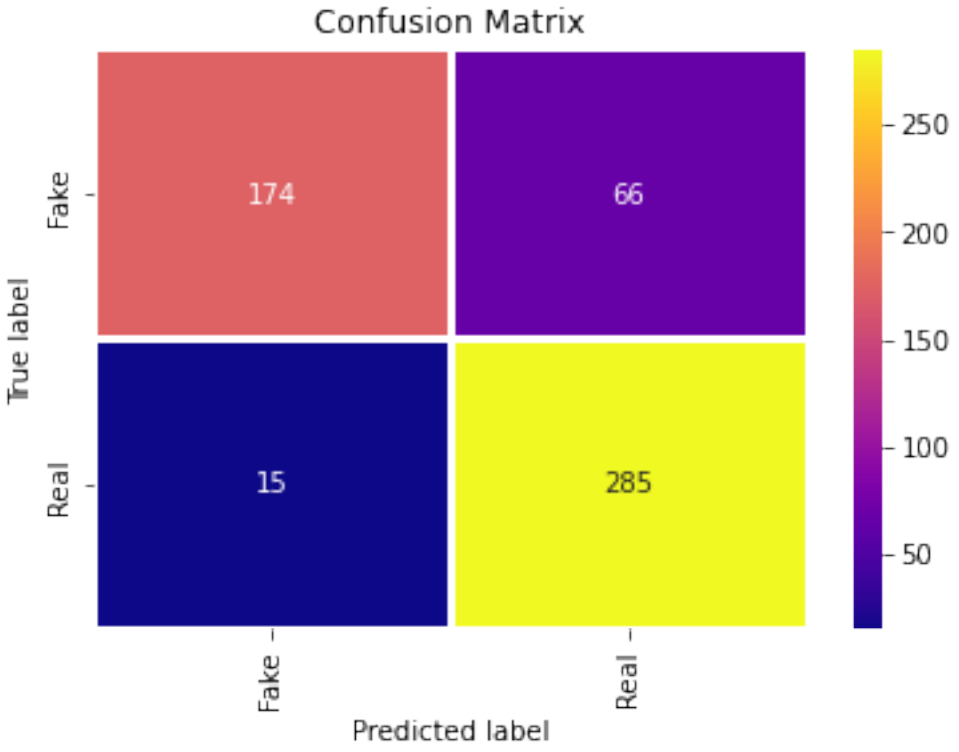

Fig. 11. The result of the prediction of the InceptionV3.

Fig. 10 displays the confusion matrix for the Xception model for the deepfake and real images dataset. The number of images is 540, which were divided into 240 fake images and 280 real images. Fifty-eight images were incorrectly labeled as fake when they were real faces and 26 images were real but incorrectly labeled as fake. Furthermore, 182 of the photographs were accurately identified as fake, while 272 of the images were correctly identified as real.

Fig. 11 displays the confusion matrix for the Inception V3 model for the deepfake and real images dataset. The number of images is 540, which were divided into 240 fake images and 280 real images. Sixty-six images were incorrectly labeled as fake when they were real faces and 16 images were real but incorrectly labeled as fake. Furthermore, 174 of the photographs were accurately identified as fake, while 285 of the images were correctly identified as real.

VI. Conclusion and Future Works

A new technique called "deepfake" is being employed to uses AI to generate realistic but fake images of people, particularly public figures. While not all fake information is harmful, some of it genuinely threatens the global community and should be identified. The main goal of this research was to develop a reliable and accurate method for spotting phony pictures. Researchers have used a number of techniques to find deep-fake content. However, the significance of this work lies in obtaining positive outcomes while utilizing the CNN architecture. In this study, the paper employed transfer-learning techniques in the proposed framework to enhance the accuracy of detection and reduce execution time. Also, the paper applied the proposed model to several pre-trained models, and a comparison was made in terms of accuracy, sensitivity, recall, and F1 score. This research used five pre-trained models — Resnet50, Inception V3, DenseNet201, Xception, and MobileNet — to detect deep-fake images using public datasets. The dataset contained deepfake and real images, with 4,700 training images and 540 test images. The final fully connected layer in the pre-trained models was eliminated in this study and replaced with a classifier that uses dropout, GAP, and a dense layer with two neurons that employs SoftMax. Image augmentation techniques were also used, with the help of the optimizer Adam. Some improvements can be made to the deep-learning framework used in this paper, such as applying the framework to different datasets, performing experiments using pre-trained models different from those used in this paper, and merging two CNN models with each other. The aim of this research was to design an application that detects deepfakes and gives an accurate and automatic performance evaluation. Additionally, we intend to evaluate this work on low-resolution, low-light imagery and extend it to real and fake video recognition.

References

[1] J. McCarthy, "From here to human-level AI," *Artificial Intelligence*, vol. 171, no. 18, pp. 1174–1182, 2007.

[2] J. Atwan, M. Wedyan, Q. Bsoul, A. Hamadeen, R. Alturki, and M. Ikram, "The effect of using light stemming for Arabic text classification," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, 2021.

[3] I. E. Naqa and M. J. Murphy, "What is machine learning?, in machine learning in radiation oncology," *machine learning in radiation oncology, Cham: Springer*, pp. 3–11, 2015.

[4] R. F. Murray, "Classification images: A review," *Journal of vision*, vol. 11, no. 5, pp. 2–2, 2011.

[5] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70–90, 2018.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] M. Ibrahim, M. Wedyan, R. Alturki, M. A. Khan, and A. Al-Jumaily, "Augmentation in healthcare: Augmented biosignal using deep learning and tensor representation," *Journal of Healthcare Engineering*, vol. 2021, 2021.

[8] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5325–5334.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[10] T. T. Nguyen, Q. V. H. Nguyen, D. T. Nguyen, D. T. Nguyen, T. Huynh-The, S. Nahavandi, T. T. Nguyen, Q.-V. Pham, and C. M. Nguyen, "Deep learning for deepfakes creation and detection: A survey," *Computer Vision and Image Understanding*, vol. 223, p. 103525, 2022.

[11] T. Jung, S. Kim, and K. Kim, "DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern," *IEEE Access*, vol. 8, pp. 83 144–83 154, 2020.

[12] M. Westerlund, "The emergence of deepfake technology: A review," *Technology innovation management review*, vol. 9, no. 11, 2019.

[13] B. U. Mahmud and A. Sharmin, *Deep Insights of Deepfake Technology : A Review*, vol. 5, 2020.

[14] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," *15th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pp. 1–6, 2018.

[15] F. M. Salman and S. S. Abu-Naser, "Classification of Real and Fake Human Faces Using Deep Learning," *International Journal of Academic Engineering Research*, vol. 6, 2022.

[16] J. Sharma, S. Sharma, V. Kumar, H. S. Hussein, and H. Alshazly, "Deepfakes Classification of Faces Using Convolutional Neural Networks," *Traitement du Signal*, vol. 39, pp. 1027–1037, 2022.

[17] M. Taeb and H. Chi, "Comparison of Deepfake Detection Techniques through Deep Learning," *Journal of Cybersecurity and Privacy*, vol. 2, pp. 89–106, 2022.

[18] A. Dhar, P. Acharjee, L. Biswas, S. Ahmed, and A. Sultana, "Detecting deepfake images using deep convolutional neural network," 2021.

[19] H. S. Shad, "Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network," *Comput Intell Neurosci*, vol. 2021, 2021.

[20] M. Masood, "Classification of Deepfake Videos Using Pre-trained Convolutional Neural Networks," *2021 International Conference on Digital Futures and Transformative Technologies*, 2021.

[21] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger. [Online]. Available: https://github.com/liuzhuang13/DenseNet

[22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[23] K. He, X. Zhang, S. Ren, and J. Sun. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[24] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[25] X. Qiu, "Pre-trained models for natural language processing: A survey," *Sci China Technol Sci*, vol. 63, pp. 1872–1897, 2020.