IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 39, No. 3, June, 2022, pp. 1027-1037
Journal homepage: http://iieta.org/journals/ts

Deepfakes Classification of Faces Using Convolutional Neural Networks

Jatin Sharma[1], Sahil Sharma[1], Vijay Kumar[2], Hany S. Hussein[3,4], Hammam Alshazly[5*]

[1] Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala 147004, India
[2] Computer Science and Engineering Department, National Institute of Technology, Hamirpur, Himachal Pradesh 177005, India
[3] Electrical Engineering Department, College of Engineering, King Khalid University, Abha 62529, Saudi Arabia
[4] Electrical Engineering Department, Faculty of Engineering, Aswan University, Aswan 81528, Egypt
[5] Faculty of Computers and Information, South Valley University, Qena 83523, Egypt

Corresponding Author Email: hammam.alshazly@sci.svu.edu.eg

https://doi.org/10.18280/ts.390330

Received: 7 May 2022
Accepted: 8 June 2022

Keywords:
deep learning, fake faces, deepfakes, transfer learning, deep neural networks

ABSTRACT

In the recent years, petabytes of data is being generated and uploaded online every second. To successfully detect fake contents, a deepfake detection technique is used to determine whether the uploaded content is real or fake. In this paper, a convolutional neural network-based model is proposed to detect the fake face images. The generative adversarial networks and data augmentation are used to generate the face dataset for real and fake face classification. Transfer learning techniques from pretrained deep models such as VGG16 and ResNet50 are employed in the proposed model. The proposed model is evaluated on three benchmark datasets, namely 140k Real and Fake Faces, Real and Fake Face Detection, and Fake Faces. The proposed model attained accuracies over the three datasets are 95.85%, 53.25%, and 88.63%, respectively. Moreover, to improve the obtained results of the proposed model, we combine it with other pretrained models of VGG16 and ResNet50 to construct deep ensembles. The overall performance is greatly improved with the ensemble model achieving accuracies on the three datasets as 98.79%, 75.79%, and 95.52%, respectively. Furthermore, the obtained results also show that the proposed models have superior performance than existing models.

1. INTRODUCTION

Deepfakes refers to synthetic media and a deep learning-based technology for creating false videos by replacing one person's face in an existing image or video with another [1]. This approach often requires a lot of image and video data to train models. It has the potential to produce false impressions of the presence of a person and behaviors which do not exist in actuality, with significant political and social issues, economical as well as legal ramifications [2]. Motivated from the fake faces that are created using deep learning models such as Generative Adversarial Networks (GANs) and autoencoders that are commonly employed in the field of computer vision. The real and fake face images are discriminated through various deep learning techniques [3]. Moreover, in the detection of deep fakes, Convolutional Neural Networks (CNNs) provide a significant performance boost [4]. Inspired by the previous studies, in this work, various CNN-based models are used to detect deep fake faces.

Deepfake may generate a synthetic bridge over a river when there is not one in the real world, baffling military specialists [5]. A Deep Translation-based Change Detection Network (DTCDN) was developed for optical and Synthetic Aperture Radar (SAR) images [6]. Deep translation initially transfers pictures from one area (e.g., optical) to another area (e.g., SAR) in the same feature space using a cyclic structure. They get analogous as a result of their similar traits after deep translation. Unlike most earlier studies, the translation results are sent into a supervised change detection network, which uses deep context information to distinguish between unmodified and changed pixels. 3D models are transformed to voxel sizes 43, 83, and 163 in the V3DOFR method [7]. Many open-source face-swapping programs and applications led to a slew of deep fake videos sprouting on social media, posing a significant technological challenge for identifying and riddling such content. Table 1 describes some useful tools, features, and associated links. As stated, deep faking has certain drawbacks, but we do have a good example of beneficial use, such as a video developed to begin a campaign to end Malaria. The Malaria-Must-Die campaign made a video with David Beckham, a great soccer player, in which he appears to speak nine languages with the aid of deepfake technology. David's voice shifts from masculine to female, yet his lips remain precisely in sync with the words. This technology created a visual representation of him uttering each syllable by manipulating his facial motions. Fake faces do have an impact on public perception.

Deepfakes are considered the most dangerous type of synthetic media. They can produce entertaining videos of anyone doing anything, anywhere, even though their most well-known application to transplant celebrities' heads onto actors' bodies in obscene flicks. Deepfake photos appear to be genuine content, with the person generating the phony films performing some sort of activity. Many images of the targeted subject from various perspectives are used to overlay over the original face. Deepfake has both advantages and

disadvantages. Faces and other body parts are combined with images to give them a unique appearance. Table 2 summarizes the pros and cons of deepfake technology.

Table 1. A summary of some existing deepfake tools

Ref.	Tools	Links	Key Features
[8]	FaceSwap	https://github.com/deepfakes/faceswap	There are two encoder-decoder pairs in use. The parameters of the encoders are also separated.
[9]	FSGAN	https://github.com/YuvalNirkin/fsgan	The face-swapping as well as re-enactment system that can be used on any pair of faces without requiring any prior training.
[10]	Transformable Bottleneck Network	https://github.com/kyleolsz/TB-Networks	Using a transformable bottleneck architecture, apply spatial transformation to the CNN model [11].
[12]	MarioNETte	https://hyperconnect.github.io/MarioNETte/	A handful of shot face reproduction structures retain the sack identification.
[13]	StyleRig	https://gvv.mpi-inf.mpg.de/projects/StyleRig/	Create a portrait image of the face with a rig-like command over pretrained style GAN.

Table 2. Pros and Cons of deepfake technology

Pros	Cons
Artificial intelligence advances are making it more difficult to spot phony faces with the naked eye.	Deep fake technology is used to put celebrity faces onto the bodies of porn performers.
The technique is used to control the expressions of people's faces. For instance, the Malaria Must Die campaign incorporates a video developed by Malaria. David Beckham, the legendary soccer player, speaks nine languages.	It can also create fictitious satellite photos of the world in order to deceive the military by containing an item that does not exist.
On the search engine, the website gets popular. As more individuals look for such erotic themes as a result of attracting unusual attention from the online public.	This technique manipulates idols by transforming them into something they are not, resulting in a negative impact on their reputation.

The key contributions of this manuscript are:

1. A deep learning approach based on CNN is proposed for recognizing real and fake faces.
2. The proposed approach is compared with numerous deep learning-based detection methods on three well-known benchmark datasets, namely 140k Real and Fake Faces [14], Real and Fake Face Detection [15], and Fake faces [16].
3. A deep voting ensemble of various deep CNN model is proposed to achieve better accuracy compared to any of the single models.

The remainder of the paper is organised as follows. Section 2 is devoted to the related studies. Section 3 describes the fundamental notions of how deepfakes may be detected using different CNN architectures. Section 4 outlines a potential methodology for recognizing fake faces. The results of the experiments are presented in Section 5, along with a commentary. Section 6 concludes the research work and defines the future scope.

2. RELATED WORK

Recently, researchers are developing models for classification of deepfake faces.

They are utilizing deep learning techniques such as CNNs, GANs, and transfer learning techniques.

Korshunov and Marcel [17] introduced a collection of deepfake videos as the first publicly available dataset created with videos from the Vid-TIMIT database [18]. The main goal of this dataset is to use GANs to produce the swapped faces of two people from videos. This study was done with two-dimensional (2D) facial videos. They emphasized that the setting of training and blending put a major influence on video quality. VGG-Net and FaceNet have seen to be in jeopardy due to deepfake videos. FaceNet revealed an error rate of 8.97%. GAN provided a challenge to face detection and recognition systems. Using GAN, face-swapping approach makes 2D face recognition is a more challenging task. Dolhansky et al. [19] created a huge collection of face swap videos to train the detection methods. The introduced DeepFake Detection Challenge (DFDC) dataset is the largest publicly available face swap video collection with over 100,000 total clips collected from 3,426 hired actors and multiple deep fakes. Deepfake detection is a difficult task that has yet to be addressed. DFDC trained a deepfake detection model that can be applied to genuine deepfake videos. This model may be useful for analyzing potentially deep phony videos. Li et al. [2] proposed Celeb-DF datasets with enough videos. This dataset consists of 590 genuine videos and 5639 deep fake videos. Celeb-DF dataset is used to address the color mismatch and improper face masks problems. A new deep fake synthesis algorithm was used to generate the fake videos and improve visual quality. Several techniques such as CNN [20], GoogleNet [21], Inception v3 [22], ResNet [23] were used for creation of this dataset.

Bitouk et al. [24] described a technique for creating 3D textured model to detect the face from a single shot by adding a new facial texture. During the conversion from 3D to picture, the algorithm served as morphable model that optimizes the control parameters of the model. Korshunova et al. [25] approached face-swapping as a style transfer challenge with facial recognition. Both the contents and styling losses are handled by a multiscale texture network using VGG-19 feature spaces. Güera and Delp [26] demonstrated the deepfake videos have distinct properties that distinguish them from unaltered ones. They employed CNN information from video frames to anticipate sequences by using a Long Short Term Memory (LSTM) network. Li et al. [27] recommended a face X-ray to reveal the trace of alteration around the artificial face's boundary region. Nguyen et al. [28] introduced a capsule network for image and video detection. Korshunov and Marcel

[17] proposed a capsule network to overcome the limitations of CNN in the inverse graphics task, which tries to find physical processes. Liu et al. [29] suggested Gram-Net, a novel architecture based on verified research on real and fake faces, and made two key discoveries. Fake faces have a different texture than actual faces. The statistics of global texture are more resistant to image modification and are transferrable between fake faces from other GANs. So, for powerful fake image identification, Gram-Net grasps global image texture representation. Gram-Net was more powerful for image-altering tasks such as JPEG, noise, and blur.

A new dataset was released with approximately 53,000 images and 150 videos obtained from a range of fractionally generated fakes such as computer graphic image generation and various fiddling-based ways [30]. Photos from popular face-swapping applications often seen on phones were also included. Large-scale tests using a deep learning-based detection method were done to verify the adequacy of the detection methodology. Tariq et al. [31] presented an image forensic tool that uses the neural network to detect fake face photos. The main goal was to distinguish between GAN-generated phony photos and human-created standards. The method was utilized for recognizing fraudulent face photos made by humans and GANs with higher accuracy. Guo et al. [32] proposed an Adaptive Manipulation Traces Extraction Network (AMTEN), which performs as a starting point for restricting picture size and emphasizing direction traces. AMTEN employed an adaptive convolution layer for predicting image modification traces that are subsequently repeated in the resultant layer. It is done by modifying weight during the backpropagation step to increase manipulation artifacts. AMTEN and CNN were combined to create the fake detector.

Hsu et al. [33] proposed a contrastive loss-based deep learning-based appearance for identifying fake photos. A variety of cutting-edge fake–real photo pairings were created using GANs. Then, the proposed common fake feature network was trained to distinguish between the fake and real photos using paired learning. Ismail et al. [34] proposed a novel deepfake detection method by using Extreme gradient boosting. You Only Look Once (YOLO) face detector was used to extract the face region from video frames. InceptionResNetV2 was used to extract features from these faces. These characteristics were supplied to XGBoost, which acts as a recognition system on the top of CNN network. Zhang et al. [35] focused on addressing the relevance of this problem and discussed the viability using machine learning. They performed an automated face-swapping detection. Tariq et al. [36] introduced CLRNet as a Convolutional LSTM-based Residual Network (CLRNet) that uses a novel model training technique to examine spatial and temporal information in deepfakes. Guarnera et al. [37] used an Expectation-Maximization (EM) method. The suggested technique extracted a collection of local features that were precisely targeted to describe the underlying convolutional generating process. Shad et al. [38] distinguished between bogus and real photos properly. Eight distinct CNN models were trained. Table 3 summarizes the research work done in direction of deepfake face detection.

Table 3. A summary of deepfake face detection techniques

Ref.	Model	Remark	Dataset
[38]	CNN	In identifying and classifying GAN-generated pictures, convolutional neural networks are quite successful.	140k Real and Fake Faces [14]
[32]	AMTENnet	AMTEN combines an adaptive convolution layer with a resultant layer for predicting modification traces of images, which are subsequently repeated in the resultant layer by modifying weight during the backpropagation pass to increase manipulation artifact.	Hybrid Fake Face (HFF) [39]
[36]	CLRNet	CLRNet is more encyclopedic; they can break spatio-temporal instruction, extract features from input images, and eliminate the demand for fleeting by combining two networks.	DeepFake-in-the-Wild (DFW) video [40]
[34]	XGBOOST	Face ranges are obtained from video frames using YOLO face detectors, features have been extracted from the face using InceptionResNetv2 CNN, and the CNN network's scale level is recognized using xgboost.	CelebDF-FaceForensics (Celeb-DF [2], FaceForensics++ [41])
[2]	Xception-c40	Xception-c40 is a fraudulent video detector trained on H.26 video with an intermediate degree of compacting (23) and a large degree of compacting (40).	Celeb-Df [2]
[29]	GramNet	Gram-Net uses global picture texture representations to identify fraudulent images with high accuracy. Gram-Net is more resistant to image processing techniques such as quantizing, JPEG compression, blur, or noise.	CelebA-HQ [42], Flickr-Faces-HQ [43]
[27]	CNN	Rather than recording a single artifact of a certain operation, assist in detecting the blending of the target and original faces.	Face Forensics++ [41], DFDC [44], Celeb-DF [2]
[33]	CNN concatenated to common fake feature network (CFFN)	To give prominence, the CFFN is used to extract features using the Siamese network architecture [45] and a CNN to classify them.	CelebA [46]
[47]	ResNet [23] pre-trained with ImageNet [48]	The classifier uses many fake photos generated by a fast unconditional GAN model, such as ProGAN [49]. Examine how well the classifier applies to additional CNN-generated pictures.	Face Swapping [24], StyleGAN [50]
[37]	KNN, linear discriminant analysis, and SVM	Extracting local features related to the convolutional generation process of a GAN-based image deep-fake generator using the expectation-maximization technique.	CelebA [46]
[28]	Capsule Network	capsule network can identify many types of parodies, ranging from replay assaults via printed pictures or recorded films to computer-generated movies utilizing deep convolutional neural networks	FaceForensics [51] Computer generated images (CGIs) and photographic images (PIs) [52]

[26]	Recurrent Neural Network (RNN)	The frame-level features from the video are extracted using a convolutional neural network, and the extracted features are then used to train a recurrent neural network that enrolls to identify whether the video has been ruled to the direction or not. A convolutional neural network is used to extract the frame-level features from the video and then extracted features are used to train a recurrent neural network that enrolls to classify if video has been ruled to the direction or not.	HOHA [53]
[20]	Inception v3	In a face classification stream, train GoogLeNet inception v3 to identify tampering artifacts, and in a second stream, train a patch-based triplet network to exploit features collecting local noise residuals and camera attributes.	FaceSwap [54] Swapme [55]
[35]	SVM, RF,MLP	Extraction of discriminant features using a bag of words method. The extracted features are fed into RF, SVM, and MLP for classification	LFW face database [56]
[21]	GoogLNet	In the ImageNet Large-Scale Visual Recognition Challenge 2014, a deep convolutional neural network architecture called Inception reaches the new state-of-the-art classification and detection. The enhanced usage of computer resources inside the network is the fundamental feature of this design.	ImageNet Large-Scale Visual Recognition Challenge [48]

The major goal of this research is to identify deepfake pictures from normal photographs with ease. Many studies have been conducted on the tricky problem of "deepfake." To detect deepfake photos, several researchers employed a CNN-based method, whereas others utilized feature-based approaches. Machine learning classifiers were utilized by a handful of them to recognize deepfake photos. The originality of this study is that it uses the CNN model to recognize deepfake pictures from regular photographs with 95.85% accuracy. In our study, we used more CNN architectures than many other academics, which has set us apart. Moreover, we provided a thorough analysis, and the result exceeded earlier efforts.

3. BACKGROUND

This section discusses the different CNN architectures considered for the deepfake detection.

3.1 ResNet50

A residual neural network (ResNet) is a type of artificial neural network (ANN) that stacks residual blocks on top of each other to form a deep network [23]. ResNet50 whole design is seen in Figure 1. ResNet is divided into five phases, each phase has identity and convolution block. Because of the pooling layer at each level, the size of ResNet50 is reduced. There are three convolution layers in each convolution block and three layers in each identity block. There are around 23 million trainable parameters in the ResNet50. Due to its superior performance, ResNet50 has been utilized in various image-based detection and classification applications [57, 58].

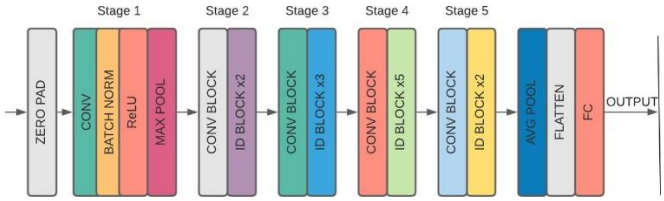

Figure 1. Architecture of ResNet50

3.2 Convolution neural network

Multiple building blocks make up a CNN, including convolution layers, pooling layers, and fully connected layers. It is built so that it automatically learns spatial hierarchies of features through backpropagation. After passing through multiple construction blocks, CNN architecture gets an input image and distinguishes between real and fake faces. The number of epochs, batch size, activation layer, regularization approach, and optimizer are the hyper-parameters that must be optimized during CNN training.

3.3 VGG 16

Visual Geometry Group (VGG) models represent a CNN architecture [59]. The most distinctive feature of VGG16 is a large number of hyper-parameters. The 16 in VGG16 alludes to the fact that it contains 16 layers with different weights. VGG16 have 3×3 filter convolution layers with a stride of 1 and always used the same padding and max pool layer of 2×2 filter stride of 2. The convolution and max pool layers are arranged throughout the architecture in the same way. It has two FC (fully connected layers) followed by a SoftMax for output. This network is quite huge, with approximately 138 million (estimated) parameters [60] (see Figure 2).

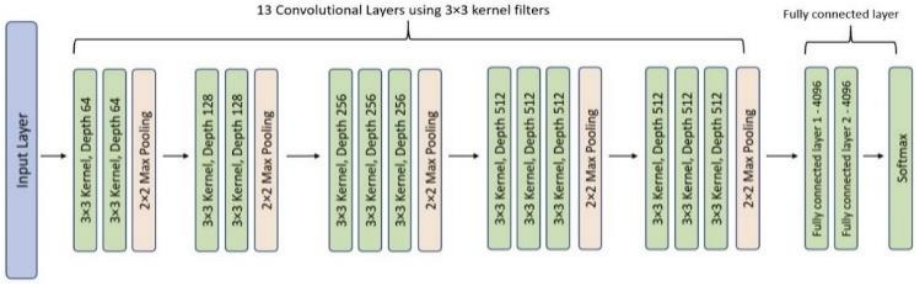

Figure 2. Architecture of the VGG16 model [60]

4. PROPOSED DEEPFAKE DETECTION

This section describes the motivation, proposed framework for the deepfake identification, and characteristics of 2D ConvNet.

4.1 Pre-processing

Two main steps are usually considered in the pre-processing phase: Data augmentation and handling class imbalance. Data augmentation is a technique for synthetically increasing the size of a training dataset by altering the images in the dataset using different data augmentation techniques such as flip, rotation, and scale. The data augmentation helps to minimize the overfitting problem. In our study, there is no such thing as a class imbalance. The utilized dataset, which includes 140K real and fake faces, was already balanced. Figure 3 represents the histogram plot of classes.

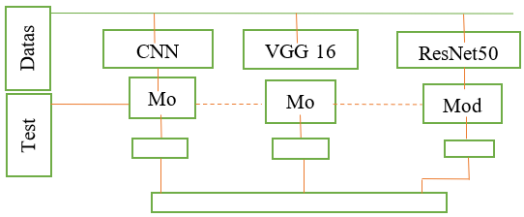

Figure 3. Real vs. Fake images for checking the class imbalance issue

4.2 Proposed deepfake recognition framework

The deep fake detection framework is based on the concepts of deep learning. The multiple models including 2D-CNN, ResNet50, and VGG16 to create the framework that can recognize whether a face image is real or fake. The sigmoid activation function is used for image categorization. Ensemble voting is also used to enhance the model performance than any of the single models in the ensemble approach. Figure 4 illustrates the architecture of the ensemble voting approach. Two approaches are proposed for ensemble voting, which are

1. Hard voting ensemble, which involves summing the votes from all models and the class label is predicted using the majority voting scheme.
2. Soft voting ensemble, which involves summing the predicted probabilities from all models and predicting the class label with the largest sum probability.

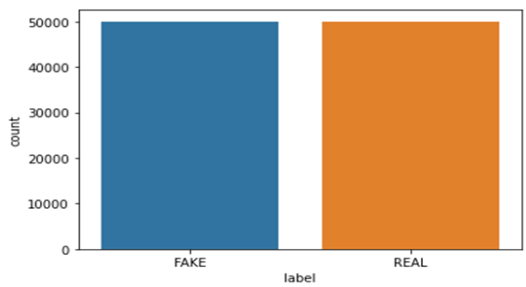

Figure 4. Architecture of the ensemble voting approach

4.3 Description of the model

The proposed CNN model is enhanced with a pooling layer after each convolution neural network for the experimentation. The pooling layer helps to reduce the image size and maintenance of a sustainable form model. The dropout layer and batch normalization follow each convolution layer. The convergence of learned representations is caused by batch normalization. Finally, a dense layer is employed as the model's output layer.

4.4 ConvNet model details

The model utilizes 2D convolutional layers, which convolve the input images using various convolutional filters. The layers distort the input by moving the filter and input vertically and horizontally, computing a dot product of the input and the weight followed by adding the chosen term. 2D ConvNet model structure with different layers, output shape, and a number of parameters is shown in Table 4.

Table 4. Model architecture of 2D ConvNet

Layers	Output Shape	No. of parameters
Conv2D	(None,222,222,32)	896
Max pooling2D	(None,111,111,32)	896
Conv2D 1	(None,109,109,32)	9248
Max_pooling2D_1	(None,54,54,32)	0
Flatten	(None,93312)	0
Dense	(None,128)	11944064
Dense_1	(None,128)	16512
Dense_2	(None,128)	16512
Dense_3	(None,1)	129

5. EXPERIMENTS AND RESULTS

This section discusses the experimental results obtained from the proposed approach and other models. The datasets, model settings, and evaluation metrics are mentioned in the succeeding subsections.

5.1 Methodology

In Figure 5, the information is first obtained from a dataset gathered from Kaggle and afterward sent through the convolution layer. Convolution is the layer that extricates various qualities from the info photographs. Convolution is a numerical cycle that is directed between the input picture and a channel of indicated size (m × m). The speck item between the channel and the info picture segment is determined by sliding the channel across the picture (m × m).

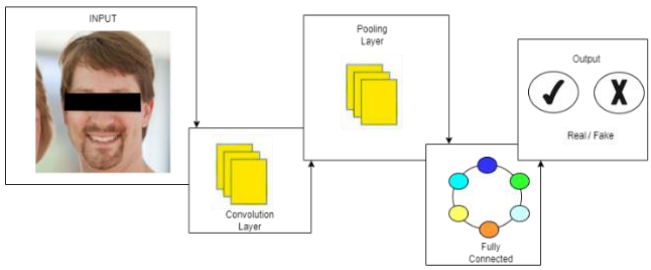

Figure 5. Work low of the proposed methodology

Subsequently, it goes through the pooling layer. The primary objective of this layer is to limit the size of the convolved highlight map. This is achieved by decreasing the associations among layers and working autonomously on each

element map. Various strategies for pooling give particular outcomes. Max-pooling chooses the greatest component from the element map.

Feed-forward neural networks are what the Fully Connected Layer is all about. Weights, biases, and neurons are all present. The preceding layers' input is flattened before being transmitted to the FC layer. On the flattened vector, more FC layers are used to perform mathematical functional operations. The categorizing process begins at this level.

5.2 Datasets

Three datasets are used in our experimentation to evaluate the proposed approaches. The first dataset is the 140k real and fake faces dataset [14], which includes 70k real faces from Nvidia's Flickr dataset and 70k fake faces taken from a million fake faces generated by a style GAN. The second dataset is the Real and Fake Face Detection Dataset [15], which contains 1081 and 960 real and fake images, respectively, with a total of 2041 images in the entire dataset. The third dataset is Fakefaces, which includes 6400 fake faces generated by the StyleGan2 model [16]. The description of all the mentioned datasets is listed in Table 5.

5.3 Parameter configuration

Table 6 summarizes the different setup configuration for the proposed CNN, ResNet50, VGG16 parameter settings. The CNN model is trained for 20 epochs with a batch size of 64. ReLU and the adaptive moment optimiser (Adam) are used as an activation and optimizer, respectively. For ResNet50 model, 20 epochs are used for training, and the batch size is set to 64. Stochastic Gradient Descent (SGD) is used as an optimiser. For VGG16, the training epochs are set to 20. The batch size is set as 64 and the SGD is considered an optimization algorithm. Even though various optimizers, namely RMSprop, SGD, and Adam are used for performance comparison on the considered datasets, the Adam optimizer provided better accuracy than the other optimizers.

5.4 Performance evaluation

Table 7 shows how the three different models perform across the three different datasets. Several evaluation metrics were used to verify the results. The table presents the results of employing CNN, ResNet50, and VGG16 on the benchmark datasets.

In Table 8, we observe that is enhanced by employing the ensemble voting of CNN+ResNet50+VGG16 on 140k Real and Fake Faces dataset. The accuracy on Real and Fake Face Detection dataset was comparatively low without ensemble voting scheme. However, it is increased to 78.56% with the ensemble. On the third dataset, Fakefaces, the ensemble voting was employed to improve the accuracy to 97.80%.

Table 5. Description of the used datasets

Dataset	Images	Resolution	Annotated
140k Real and Fake Faces [14]	1,40,000	256 × 256	No
Real and Fake Face Detection [15]	2041	600 × 600	No
Fakefaces [16]	6400	1024 × 1024	No

Table 6. Parameters setting for the proposed approach

Proposed Technique	140k Real and Fake Faces	Real and Fake Face detection	Fakefaces
2D CNN			
Number of epochs	20	20	20
Batch Size	64	64	64
Activation	ReLU	ReLU	ReLU
Optimizer	Adam	Adam	Adam
ResNet50			
Number of epochs	20	20	20
Batch Size	64	64	64
Activation	ReLU	ReLU	ReLU
Optimiser	Adam	Adam	Adam
VGG16			
Number of epochs	20	20	20
Batch Size	64	64	64
Activation	ReLU	ReLU	ReLU
Optimiser	Adam	Adam	Adam

Table 7. Performance evaluation of the models on a variety of datasets

Model	Dataset	Train Accuracy	Test Accuracy	Validation Accuracy	Sensitivity	Specificity	F1-Score	Precision	Recall
CNN		99.41	95.85	95.97	93.62	98.09	95.94	93.89	98.09
ResNet50	140k Real and Fake Faces	95.89	93.98	94.21	93.40	94.56	94.53	93.63	95.49
VGG16		89.58	86.63	87.40	81.49	91.15	87.21	84.77	89.81
CNN		55.63	53.25	54.18	53.65	53.95	53.65	52.34	52.95
ResNet50	Real and Fake Face Detection	63.95	58.24	60.75	61.63	62.05	61.13	62.11	62.37
VGG16		58.45	52.37	55.13	57.90	57.85	56.68	55.73	54.35
CNN		89.23	88.63	87.90	88.45	88.78	88.52	87.12	88.34
ResNet50	Fakefaces	88.72	85.90	83.43	85.79	86.17	87.87	85.64	87.32
VGG16		85.67	80.45	82.86	80.34	83.41	81.57	81.23	80.92

Table 8. Results obtained from ensemble voting on different datasets

Model	Dataset	Train Accuracy	Test Accuracy	Validation Accuracy	Sensitivity	Specificity	F1-Score	Precision	Recall
CNN+ResNet50+VGG16	140k Real and Fake Faces	100	98.79	98.90	97.21	99.57	99.45	99.87	99.63
CNN+ResNet50+VGG16	Real and Fake Face Detection	78.56	75.79	75.13	73.22	77.91	75.68	73.58	77.43
CNN+ResNet50+VGG16	Fakefaces	97.80	95.52	95.23	91.68	96.23	93.82	91.89	96.88

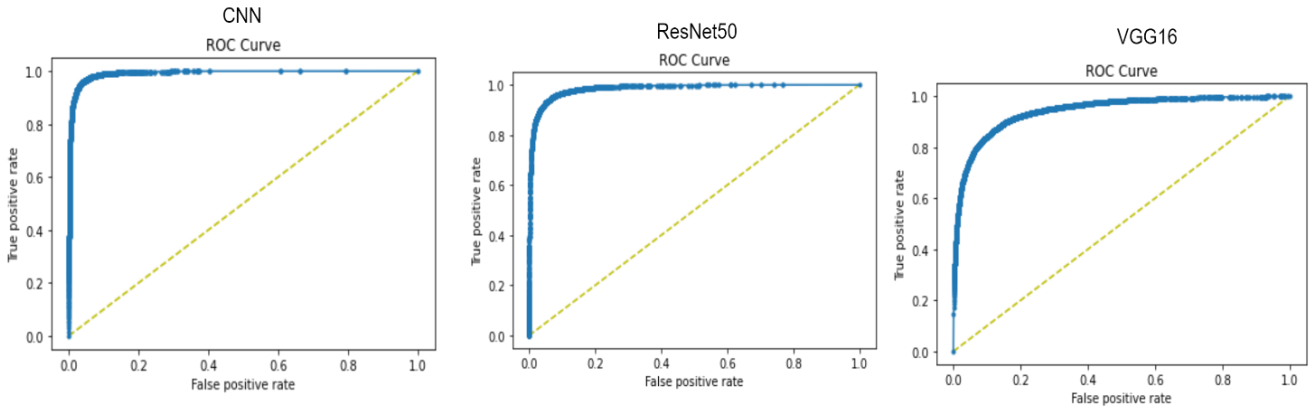

Figure 6. ROC curves for binary classification over 140k real and fake faces dataset

ROC curve is a graph that depicts the model's classification performance across all classification levels. Figure 6 depicts the ROC curve for the deep fake detection using different models in the binary classification task over 140k Real and Fake Faces dataset.

Table 9 shows the performance comparison between the proposed model and other models in terms of accuracy. It is observed from table that the proposed CNN model is better than the other models.

Table 9. Performance comparison of proposed approach and other approaches on 140k real and fake faces dataset

Reference	Model Name	Test Accuracy (%)
[61]	ResNet50	53.43
[62]	FaceNet	94.51
Proposed Approach	**CNN**	95.85

Table 10. Comparison of this work with previous studies

References	Model Name	Accuracy (%)	Accuracy in this paper (%)
[1]	VGG 16	81.6	86.63
[1]	ResNet 50	81.6	93.98
[63]	CNN	90.76	95.85

A comparison graph of numerous works investigated by deepfake is shown in Table 10. The table compares this work to a number of studies done by previous researchers who used the same models as we used in our study. The authors [1, 63] utilized VGG16 and ResNet50, individually, and the comparing correctness were 81.6% and 81.6%, separately. Wen and Xu [63] utilized a CNN model to lead their exploration and accuracy upto 90.76%.

Experimentation is conducted to clarify the accuracy of the proposed approach. Fake and actual photographs of each model are used in the experiment. As demonstrated in Figure 7, almost all photos were accurately identified or classed as "Real" or "fake," and original and false photos were chosen randomly from the validation folders.

Figures 8-10 show different models and their performance when using different optimisers. These figures show the different train and validation accuracy curves for experiments carried out on 140k Real and Fake Faces. Figure 8 shows how the Resnet50 model achieved a good performance after 20 epochs. Figure 9 shows that the CNN model after 20 epochs performed well when using RMSProp and Adam optimisers, but not so well when using the SGD optimiser. Figure 10 shows how the VGG16 model performed after 20 epochs, indicating a good accuracy.

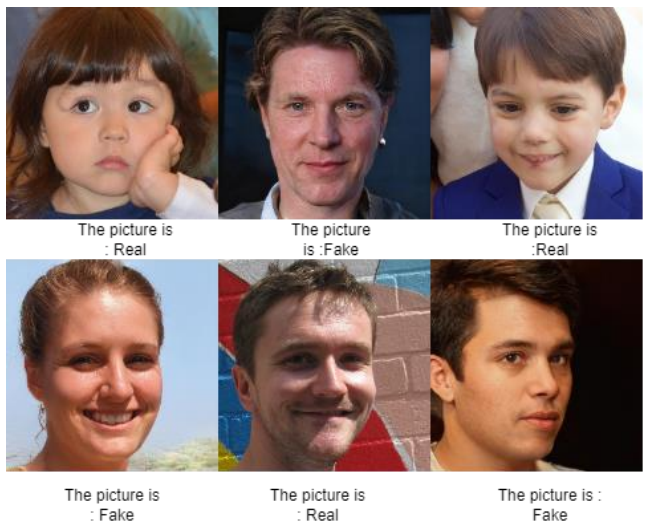

Figure 7. Classification of "Real" and "Fake" images

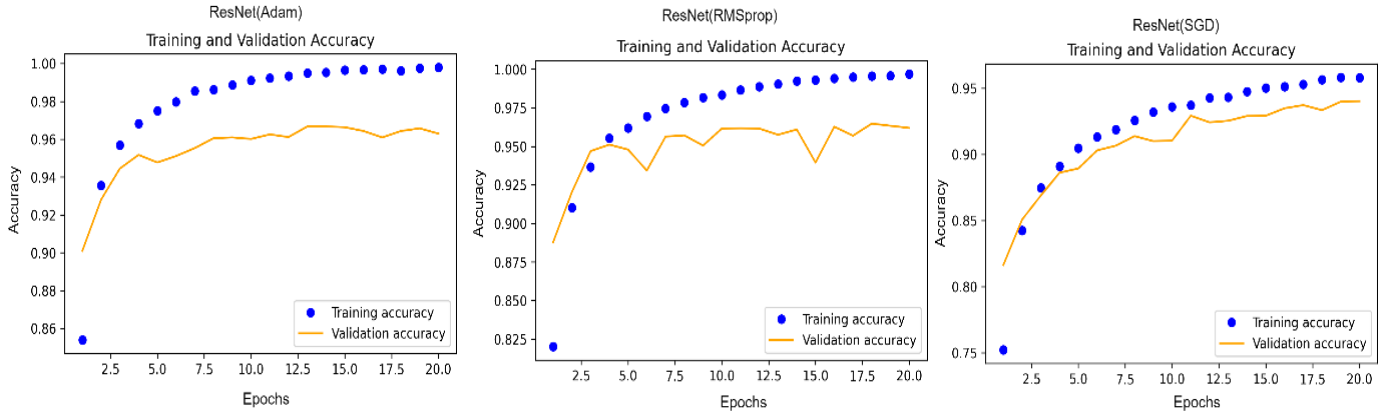

Figure 8. Training and validation curves for various optimizers using the ResNet50 model

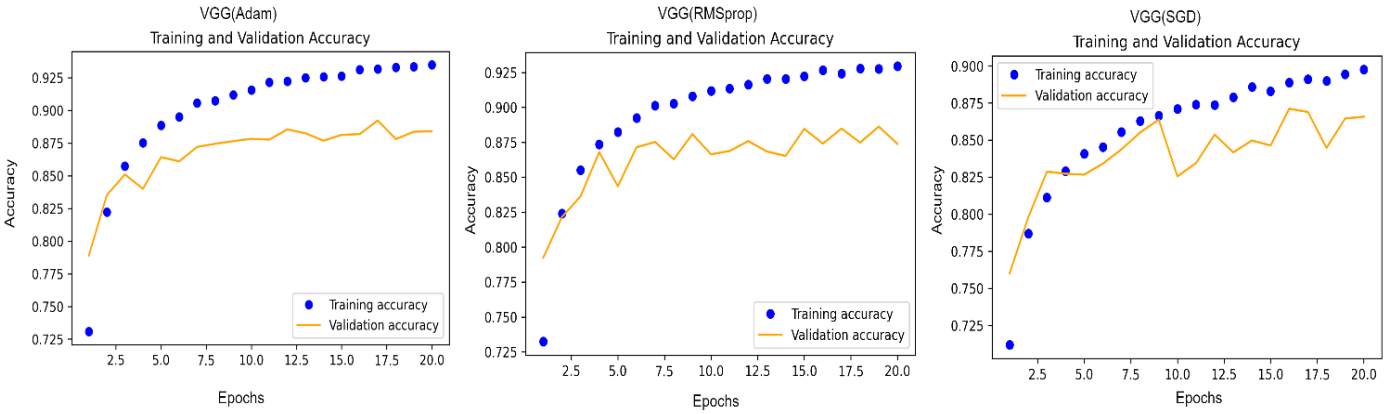

Figure 9. Training and validation curves for various optimizers using the CNN model

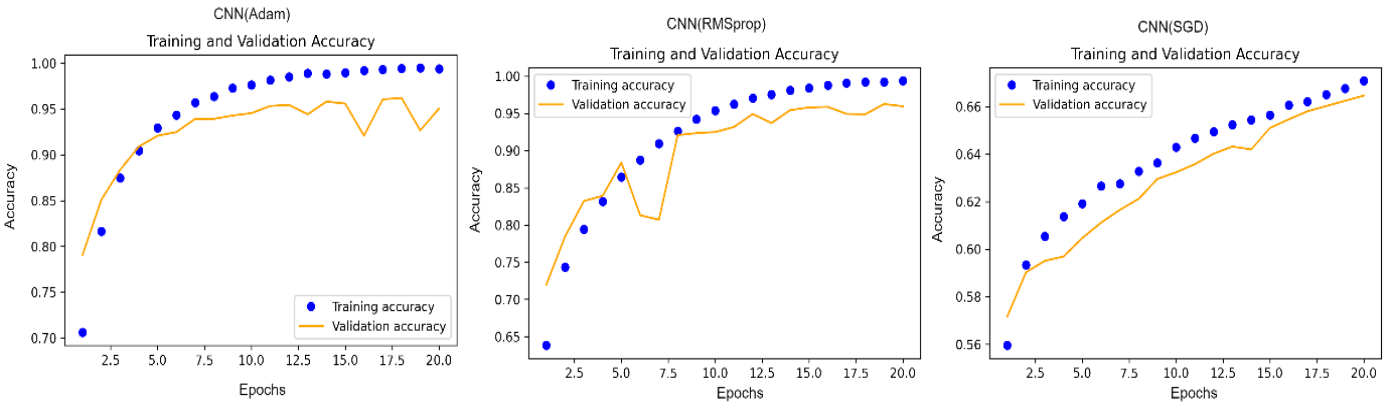

Figure 10. Training and validation curves for various optimizers using the VGG16 model

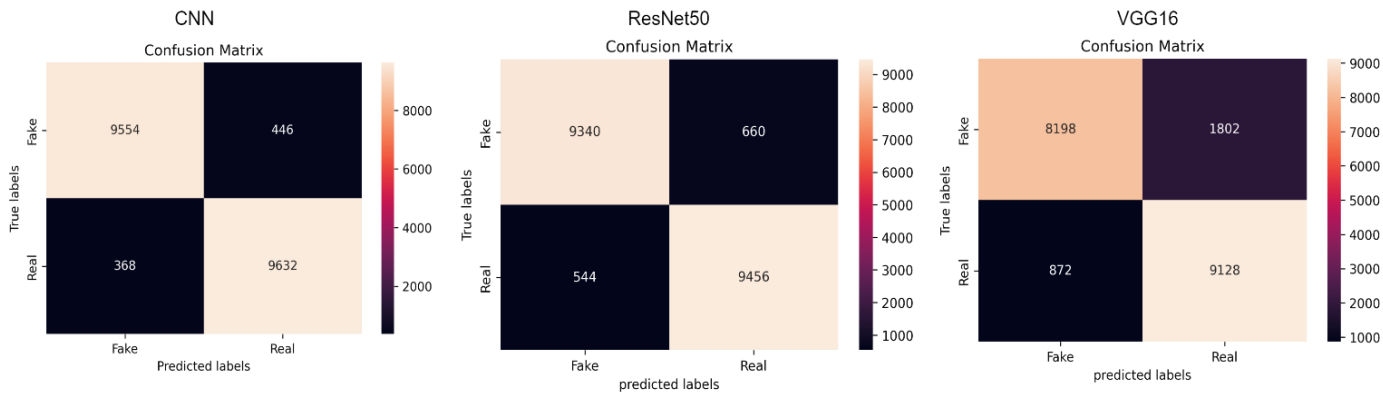

Figure 11. Confusion matrix for the different models

Figure 11 depicts the confusion matrix of the binary-class classification such as Fake and Real. The anticipated and actual classification is shown in a confusion matrix of size n × n (n number of rows and columns) associated with a classifier, and n is the number of distinct classes.

The confusion matrix on the 140k Real and Fake Faces dataset represents the true positives and true negatives in the diagonal elements for all three models namely, CNN, ResNet50, and VGG16.

6. CONCLUSION AND FUTURE WORK

In this paper, a deep CNN model was proposed for real and fake face image classification. The transfer learning approach using different pretrained models namely, VGG16 and ResNet50 were also employed. Three well-known datasets, namely 140k Real and Fake Faces, Real and Fake Face Detection, and FakeFaces, were used for validating the proposed approach. The proposed model achieved test accuracies over the three benchmark datasets as 95.85%, 53.25%, and 88.63%, respectively. The performance of the proposed model was significantly improved when we combined it with other pretrained models of VGG16 and ResNet50 to construct deep ensembles. The overall performance is greatly improved with the ensemble model achieving accuracies on the three datasets as 98.79%, 75.79%, and 95.52%, respectively. The experimental results revealed the superiority of the proposed model over the existing models. In future, the developed ensemble approach can be improved by incorporating the concept of occlusion invariant. We also plan to extend this work to real and fake video detection, as well as evaluating it on low resolution and low light images.

DATA AVAILABILITY

The datasets used to conduct the experiments are publicly available for download from these websites: https://www.kaggle.com/xhlulu/140k-real-and-fake-faces, https://www.kaggle.com/ciplab/real-and-fake-face-detection, and https://www.kaggle.com/hyperclaw79/fakefaces?select=1016.jpg.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for supporting this research through a Research Groups Program under Grant (RGP.2/16/43).

REFERENCES

[1] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. Information Fusion, 64: 131-148. https://doi.org/10.1016/j.inffus.2020.06.014

[2] Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S. (2020). Celeb-DF: A large-scale challenging dataset for deepfake forensics. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3204-3213. https://doi.org/10.1109/CVPR42600.2020.00327

[3] Khalil, H.A., Maged, S.A. (2021). Deepfakes creation and detection using deep learning. In 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), pp. 1-4. https://doi.org/10.1109/MIUCC52538.2021.9447642

[4] Badale, A., Castelino, L., Gomes, J. (2021). Deep fake detection using neural networks. International Journal of Engineering Research & Technology (IJERT), 9(3): 349-354. https://doi.org/10.17577/IJERTCONV9IS03075

[5] Nguyen, T., Nguyen, Q.V.H., Nguyen, C.M., Nguyen, D., Nguyen, D.T., Nahavandi, S. (2019). Deep learning for deepfakes creation and detection: A survey. arXiv preprint arXiv:1909.11573. https://doi.org/10.48550/arXiv.1909.11573

[6] Li, X., Du, Z., Huang, Y., Tan, Z. (2021). A deep translation (GAN) based change detection network for optical and SAR remote sensing images. ISPRS Journal of Photogrammetry and Remote Sensing, 179: 14-34. https://doi.org/10.1016/j.isprsjprs.2021.07.007

[7] Sharma, S., Kumar, V. (2020). Voxel-based 3D occlusion-invariant face recognition using game theory and simulated annealing. Multimedia Tools and Applications, 79(35): 26517-26547. https://doi.org/10.1007/s11042-020-09331-5

[8] GitHub - deepfakes/faceswap: Deepfakes Software for All. Available: https://github.com/deepfakes/faceswap, accessed on June 3, 2021.

[9] GitHub - YuvalNirkin/fsgan: FSGAN - Official PyTorch Implementation. Available: https://github.com/YuvalNirkin/fsgan, accessed on June 3, 2021.

[10] GitHub - kyleolsz/TB-Networks. Available: https://github.com/kyleolsz/TB-Networks, accessed on June 3, 2021.

[11] Olszewski, K., Tulyakov, S., Woodford, O., Li, H., Luo, L. (2019). Transformable bottleneck networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7648-7657.

[12] MarioNETte: Few-shot Face Reenactment Preserving Identity of Unseen Targets. Available: https://hyperconnect.github.io/MarioNETte/, accessed on June 3, 2021.

[13] StyleRig: Rigging StyleGAN for 3D Control over Portrait Images, CVPR 2020. Available: https://gvv.mpi-inf.mpg.de/projects/StyleRig, accessed on June 3, 2021.

[14] 140k Real and Fake Faces | Kaggle. Available: https://www.kaggle.com/xhlulu/140k-real-and-fake-faces, accessed on May 30, 2021.

[15] Real and Fake Face Detection | Kaggle. Available: https://www.kaggle.com/ciplab/real-and-fake-face-detection, accessed on June 3, 2021.

[16] Fakefaces | Kaggle. Available: https://www.kaggle.com/hyperclaw79/fakefaces?select=1016.jpg, accessed on Oct. 16, 2021.

[17] Korshunov, P., Marcel, S. (2018). Deepfakes: A new threat to face recognition? Assessment and detection. arXiv preprint arXiv:1812.08685.

[18] Sanderson, C. (2002). The Vidtimit database (No. REP_WORK). IDIAP.

[19] Dolhansky, B., Howes, R., Pflaum, B., Baram, N., Ferrer, C.C. (2019). The deepfake detection challenge (DFDC) preview dataset. arXiv preprint arXiv:1910.08854.

[20] Zhou, P., Han, X., Morariu, V.I., Davis, L.S. (2017). Two-stream neural networks for tampered face detection. In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1831-1839. https://doi.org/10.1109/CVPRW.2017.229

[21] Szegedy, C., Liu, W., Jia, Y., et al. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp. 1-9. https://doi.org/10.1109/CVPR.2015.7298594

[22] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818-2826. https://doi.org/10.1109/CVPR.2016.308

[23] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

[24] Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P., Nayar, S.K. (2008). Face swapping: automatically replacing faces in photographs. In ACM SIGGRAPH 2008 Papers, pp. 1-8. https://doi.org/10.1145/1399504.1360638

[25] Korshunova, I., Shi, W., Dambre, J., Theis, L. (2017). Fast face-swap using convolutional neural networks. In Proceedings of the IEEE International Conference on Computer Vision, pp. 3677-3685. https://doi.org/10.1109/ICCV.2017.397

[26] Güera, D., Delp, E.J. (2018). Deepfake video detection using recurrent neural networks. In 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1-6. https://doi.org/10.1109/AVSS.2018.8639163

[27] Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., Guo, B. (2020). Face x-ray for more general face forgery detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5001-5010. https://doi.org/10.1109/CVPR42600.2020.00505

[28] Nguyen, H.H., Yamagishi, J., Echizen, I. (2019). Capsule-forensics: Using capsule networks to detect forged images and videos. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2307-2311. https://doi.org/10.1109/ICASSP.2019.8682602

[29] Liu, Z., Qi, X., Torr, P.H. (2020). Global texture enhancement for fake face detection in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8060-8069. https://doi.org/10.1109/CVPR42600.2020.00808

[30] Khodabakhsh, A., Ramachandra, R., Raja, K., Wasnik, P., Busch, C. (2018). Fake face detection methods: Can they be generalized? In 2018 International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1-6. https://doi.org/10.23919/BIOSIG.2018.8553251

[31] Tariq, S., Lee, S., Kim, H., Shin, Y., Woo, S.S. (2019). Gan is a friend or foe? A framework to detect various fake face images. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, pp. 1296-1303. https://doi.org/10.1145/3297280.3297410

[32] Guo, Z., Yang, G., Chen, J., Sun, X. (2021). Fake face detection via adaptive manipulation traces extraction network. Computer Vision and Image Understanding, 204: 103170. https://doi.org/10.1016/j.cviu.2021.103170

[33] Hsu, C.C., Zhuang, Y.X., Lee, C.Y. (2020). Deep fake image detection based on pairwise learning. Applied Sciences, 10(1): 370. https://doi.org/10.3390/app10010370

[34] Ismail, A., Elpeltagy, M., Zaki, M.S., Eldahshan, K. (2021). A new deep learning-based methodology for video deepfake detection using XGBoost. Sensors, 21(16): 5413. https://doi.org/10.3390/s21165413

[35] Zhang, Y., Zheng, L., Thing, V.L. (2017). Automated face swapping and its detection. In 2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP), pp. 15-19. https://doi.org/10.1109/SIPROCESS.2017.8124497

[36] Tariq, S., Lee, S., Woo, S. (2021). One detector to rule them all: Towards a general deepfake attack detection framework. In Proceedings of the Web Conference 2021, pp. 3625-3637. https://doi.org/10.1145/3442381.3449809

[37] Guarnera, L., Giudice, O., Battiato, S. (2020). Deepfake detection by analyzing convolutional traces. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 2841-2850. https://doi.org/10.1109/CVPRW50498.2020.00341

[38] Shad, H.S., Rizvee, M., Roza, N.T., et al. (2021). Comparative analysis of deepfake image detection method using convolutional neural network. Computational Intelligence and Neuroscience. https://doi.org/10.1155/2021/3111676

[39] Guo, Z., Yang, G., Chen, J., Sun, X. (2021). Fake face detection via adaptive manipulation traces extraction network. Computer Vision and Image Understanding, 204: 103170.

[40] GitHub - deepfakeinthewild/deepfake-in-the-wild: deepfake dataset collected on the web for deepfake detection. Available: https://github.com/deepfakeinthewild/deepfake-in-the-wild, accessed on February 16, 2022.

[41] Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1-11. https://doi.org/10.1109/ICCV.2019.00009

[42] CelebAMask-HQ Dataset. Available: http://mmlab.ie.cuhk.edu.hk/projects/CelebA/CelebAMask_HQ.html, accessed on February 16, 2022.

[43] GitHub - NVlabs/ffhq-dataset: Flickr-Faces-HQ Dataset (FFHQ). Available: https://github.com/NVlabs/ffhq-dataset, accessed on February 16, 2022.

[44] Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., Ferrer, C.C. (2020). The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397.

[45] Chen, X., He, K. (2021). Exploring simple Siamese representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15750-15758.

[46] Liu, Z., Luo, P., Wang, X., Tang, X. (2015). Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision, pp. 3730-3738. https://doi.org/10.1109/ICCV.2015.425

[47] Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A. A. (2020). CNN-generated images are surprisingly easy to spot. for now. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8695-8704.

https://doi.org/10.1109/CVPR42600.2020.00872
[48] Russakovsky, O., Deng, J., Su, H., et al. (2015). ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 115(3): 211-252. https://doi.org/10.1007/s11263-015-0816-y
[49] Karras, T., Aila, T., Laine, S., Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196.
[50] Karras, T., Laine, S., Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4401-4410. https://doi.org/10.1109/CVPR.2019.00453
[51] Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M. (2018). Faceforensics: A large-scale video dataset for forgery detection in human faces. arXiv preprint arXiv:1803.09179.
[52] Ng, T.T., Chang, S.F., Hsu, J., Pepeljugoski, M. (2005). Columbia photographic images and photorealistic computer graphics dataset. Columbia University, ADVENT Technical Report, 205-2004. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.8266&rep=rep1&type=pdf, accessed on March 15, 2022.
[53] Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B. (2008). Learning realistic human actions from movies. In 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8. https://doi.org/10.1109/CVPR.2008.4587756
[54] GitHub - MarekKowalski/FaceSwap: 3D face swapping implemented in Python. Available: https://github.com/MarekKowalski/FaceSwap, accessed on February 17, 2022.
[55] Connecting to Apple Music. Available: https://apps.apple.com/us/app/swapme-by-faciometrics.
[56] Zhang, N., Deng, W. (2016). Fine-grained LFW database. In 2016 International Conference on Biometrics (ICB), pp. 1-6. https://doi.org/10.1109/ICB.2016.7550057
[57] Yildirim, M., Cinar, A. (2020). A deep learning based hybrid approach for COVID-19 disease detections. Traitement du Signal, 37(3): 461-468. https://doi.org/10.18280/ts.370313
[58] Gedik, O., Demirhan, A. (2021). Comparison of the effectiveness of deep learning methods for face mask detection. Traitement du Signal, 38(4): 947-953. https://doi.org/10.18280/ts.380404
[59] Simonyan, K., Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/1409.1556
[60] Tammina, S. (2019). Transfer learning using vgg-16 with deep convolutional neural network for classifying images. International Journal of Scientific and Research Publications (IJSRP), 9(10): 143-150. http://dx.doi.org/10.29322/IJSRP.9.10.2019.p9420
[61] Dhar, A., Acharjee, P., Biswas, L., Ahmed, S., Sultana, A. (2021). Detecting deepfake images using deep convolutional neural network. Doctoral dissertation, Brac University.
[62] Hettiarachchi, S. (2021). Analysis of different face detection and recognition models for Android. Digitala Vetenskapliga Arkivet.
[63] Wen, L., Xu, D. (2019). Face image manipulation detection. In IOP Conference Series: Materials Science and Engineering, 533(1): 012054. https://doi.org/10.1088/1757-899X/533/1/012054