Al-Nahrain Journal of Science

ANJS, Vol.26(3), September, 2023, pp. 44-50

Efficiency Evaluation of Popular Deepfake Methods Using Convolution Neural Network

Noor K. Alzurfi *, Mohammed S. Altaei

Department of Computer Science, College of Science, Al-Nahrain University, Baghdad, Iraq

Article's Information

Received: 05.07.2023
Accepted: 21.08.2023
Published: 15.09.2023

Keywords:
Convolution neural network (CNN)
Deepfake detection
Deep learning
Deepfake methods
Video deepfake

Abstract

Many deepfake techniques in the early years are spread to create successful deepfake videos (i.e., FaceSwap, DeepFake, etc.). These methods enable anyone to manipulate faces in videos, which can negatively impact society. One way to reduce this problem is the deepfake detection. It has become such a hot topic and the most crucial task in recent years. This paper proposes a deep learning model to detect and evaluate deepfake video methods using convolutional neural networks. The model is evaluated on the FaceForensics++ video dataset that contains four different deepfake ways (deepfake, face2face, faceswap, and neuraltexture), and it achieved 0.96 accuracy on the deepfake method, 0.95 accuracy on face2face approach, 0.94 precision on faceswap method and 0.76 accuracy on neuraltexture method.

DOI: 10.22401/ANJS.26.3.07

*Corresponding author: noor.kadem21@ced.nahrainuniv.edu.iq

1. Introduction

The development of technology and society leads to making deep learning to become easier to use. Many bad unprincipled people are creating fake pictures and videos by using various deep learning technology that industriously endangers the stability of the society and country, including faking politicians to make inappropriate statements, using face-swapping tools to publicize false information, and blackmailing [1]. Deepfake is one of the deep learning-powered applications that has recently emerged. It helps to create new videos for people who appear to say or do something they never did[2].

Applications and tools in deepfake are so progressing that help users to use them without any experience in digital arts and photo retouching [3]. This technological advancement led to new artistic possibilities as their applications in visual effects, Snapchat filters, and digital avatars, generate voices for those who have lost theirs, and help in updating episodes of movies without needing to reshoot them. Deepfake has creative and productive effects in photography, movie productions, video games, and entertainment [2].

There are several deepfake creation methods. Autoencoders and Generative Adversarial Networks (GANs) are popular methods [2-4]. Autoencoder consists of two components: an encoder and a decoder. Two autoencoders are trained to pass latent faces between the source and the target video frames in the deepfake algorithm. The encoder extracted latent features from the image and reconstructed faces by inputting these extracted features into two decoders. So, the face generated from face A will be passed to decoder B. The decoder B would try to reconstruct face B from a feature relative to face A. This process is repeated for every frame in the video [5]. GAN includes two neural networks: a generator and a discriminator [2-6]. The generator produced images closer to the real images while the discriminator trained to improve the capability of classifying the real and fake images [2]. Both networks were trained with backpropagation to enhance their efficiency.

It is worth mentioning some of the commonly deepfake tools:

- FaceSwap: It works on pairing the target and source frames; both will detect the facial landmarks and then transform these landmarks to match the target facial landmarks and blend into the target frame [7].

- DeepFake: it depends on the auto-encoder method, while the encoder works on specific features like facial expressions and then restores them by the decoder [7].
- Face2Face: It works by creating a 3D model of the source and target faces. The target face model is deformed to match the expressions of the source face, and the new facial expression will be retrieved after the best matches of the mouth shape and blended into the target face [7]. Recently, the video dataset has been increased for experiments [8].

The most commonly used video datasets are:

- FaceForensics++: the first large-scale video dataset used for deepfake detection. It is introduced by Rossler t al. [5]. It consists of 5000 videos, 1000 real videos from YouTube, and 4000 fake videos by four different fake methods (DeepFake, FaceSwap, Face2Face, NeuralTexture). The DeepFake and FaceSwap methods depend on swapping the face, while the Face2Face and NeuralTexture methods depend on the manipulation of the expression of that face [5].
- Celeb-DF: The Celeb-DF dataset consists of 590 real videos from YouTube and 5639 deepfake videos generated by swapping faces [9].
- UADFV: The UADFV consists of 49 real videos from YouTube and 49 deepfake videos generated by FakeApp [5-9].
- DFDC: The DeepFake Detection Challenge dataset consists of 1,131 real videos and 4,113 deepfake videos [8-9].

The deepfake detection process has become a challenging task, and accordingly, researchers have proposed various DL methods for detection. Mousa et al. in 2020, designed and implemented a deepfake detection model with mouth features (DFT-MF) by using CNN to classify fake and real videos. Experiments were done on Celeb-DF and Vid-TIMIT datasets. The proposed model has achieved a 71.25% accuracy rate on the Celeb-DF dataset, 98.7% on the Vid-TIMITLQ dataset, and 73.1 on the Vid-TIMIT- HQ dataset [10]. David Guera et al. 2018 proposed a temporal-aware pipeline to detect deepfake videos using CNN to extract features and recurrent neural network (RNN) to classify a video as fake or real. They evaluated their method on a large set of deepfake videos from the HOHA dataset with an accuracy greater than 97% [11]. Momina Masood, et al. 2021 have developed a pipeline for recognizing and detecting faces and used several deep learning (DL) based techniques to compute features from extracted faces using the DFDC dataset. SVM classifier is used to classify the data as real or fake. The highest accuracy was 98% for DenseNet-169, and the lowest accuracy was 89% for VGG-16 [12]. Liwei Deng, et al. in 2022, proposed using a new EfficientNet-V2 network to determine the authenticity of images and videos. They handled the large-scale fake face datasets and compared the EfficientNet-V2 performance with the existing detection networks. The used datasets are FF++ and FFIW10k++; they can reach about 97.9% and 93.0% [1]. Vurimi Vamsi, et al. 2022 provided ResNext, a CNN and Long Short-Term Memory (LSTM), used to detect Deepfake videos. This model used the celeb-df dataset and they can obtain a high accuracy of about 91% [13]. Alakananda Mitra, et al. in 2020, presented a neural network (NN) consisting of CNN for extracting frame features and a proposed classifier network for detecting deepfake videos. This network obtained the best accuracy of 96% for compression level c=23 and 93% for c=40 on the FaceForensics++ dataset [14]. Atharva Shende in 2021, used CNN for extracted features and RNN to classify the videos as real or fake. The model can predict 94.21% correctly on Celeb- DF dataset [15].

Deressa Wodajo, et al. in 2021, proposed a Convolutional Vision Transformer (ViT) to detect the Deepfakes, consisting of CNN and ViT. They trained the model on the DFDC dataset and achieved 91.5% accuracy, 0.91 AUC, and 0.32 loss [16]. Nicolo Bonettini, et al. in 2021, studied the ensemble of different trained CNN models to tackle the problem of detecting the face manipulation in videos. The proposed system has obtained 0.94 accuracy on FaceForensics ++ dataset and 0.87 accuracy DFDC dataset [3]. Md Shohel Rana, et al. in 2021, used the traditional procedure of feature construction, extraction, and training and testing; an ML classifier was used to propose a classical ML-based technique to detect Deepfakes. They presented results on some datasets: 99.84% accuracy on FaceForensics++, 99.66% accuracy on VDFD ,99.38% accuracy on DFDC, and 99.43% on Celeb-DF datasets [17].

In this paper, we propose a simple method based on deep learning to detect facial deepfake videos. The CNN is used to extract features and to detect the real and fake videos of the FaceForensics++ dataset [18], which is one of the large video datasets and is widely used in the deepfake detection field.

2. Work and Methods

2.1 The Proposed Deepfake Detection Method

The concept of multistage processing and video preprocessing has been used to design the proposed deepfake detection method. It is claimed that these stages can beneficially be combined to establish an efficient deepfake video detection model. The generic structure of the proposed method is shown in Table 1. The proposed methods consist of the following sequential stages:

2.2 The Pre-processing stage

The preprocessing of the video dataset is established with different available datasets; the widely used FaceForensics++ video dataset is chosen, which consists of 1000 real and 4000 fake videos, and it is produced with different deepfake tools (Deepfake [7], Face2face [19], Faceswap [7], NeuralTexture [7]) [18]. To make the video dataset easy to use, we need to convert the video dataset to an image dataset. First, the dataset was rearranged by dividing it into four sub-datasets:

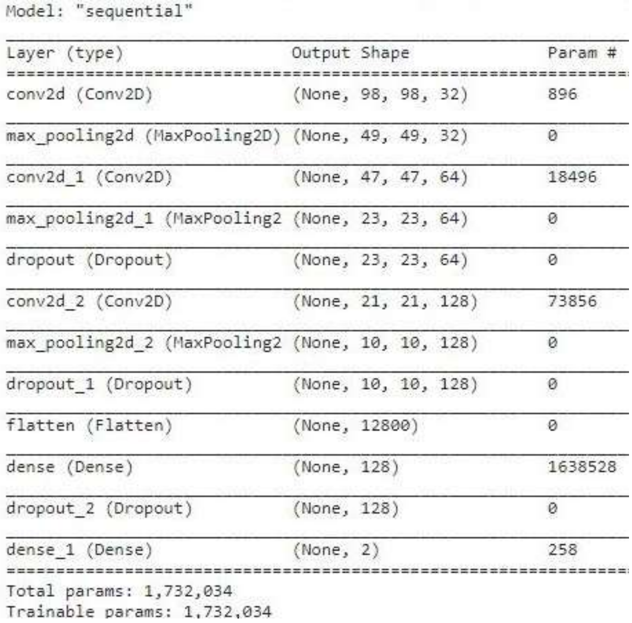
i. FaceForensics++_Deepfake
ii. FaceForensics++_Face2face
iii. FaceForensics++_Faceswap
iv. FaceForensics++_NeuralTexture

Each one of these consists of two folders, one for fake videos which contain 1000 deepfake videos and the second containing the 1000 real videos. This step evaluates the different deepfake methods and determines how the proposed method can detect the different deepfake. To convert the video dataset to an image dataset, the frames were extracted from the videos using the OpenCV library, and then one frame was chosen from each 25 frames to reduce the computation cost and time. While the work depends on the face area, the face region at each extracted frame was cropped using the face recognition library. This operation may result in some errors in face extraction, so it is necessary to delete incomplete face images, blurred images, non-face images, and different faces appearing in clothes or backgrounds. These steps converted each 1000 video to around (16-20*k*) images depending on video seconds. Then, the images were resized into 100×100 to make them easier to use before normalizing them. The dataset is then split as 85% for training (including 15% of them for validation) and the remaining 15% for testing.

2.3 CNN-based fake detection model

CNN is used to establish an effective deepfake video detection model for evaluating different popular deepfake videos. The CNN model generally consists of two stages: the feature extraction stage and the classification stage. The size of the input image data was (100×100×3). The feature extraction stage consisted of three convolution layers with a 3×3 filter size followed by three max-pooling layers stacked together. Dropout layers were added after the last two max-pooling layers to decrease the overfitting in the training process. The classification stage consisted of a flattened layer and two dense layers. The output of feature extraction layers was converted to one vector by a flattened layer. The first denes layer has 128 nodes which are fully connected layers that are followed by a dropout layer. Finally, the results will feed to the last layer in the model, which is a dense layer. This dense layer has two nodes and uses the sigmoid classifier to give only two output classes. Table 1 lists the layers of the proposed method.

Table 1. The layers of the proposed model



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 98, 98, 32)        896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 49, 49, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 47, 47, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 23, 23, 64)        0
_________________________________________________________________
dropout (Dropout)            (None, 23, 23, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 21, 21, 128)       73856
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 10, 10, 128)       0
_________________________________________________________________
dropout_1 (Dropout)          (None, 10, 10, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 12800)             0
_________________________________________________________________
dense (Dense)                (None, 128)               1638528
_________________________________________________________________
dropout_2 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 258
=================================================================
Total params: 1,732,034
Trainable params: 1,732,034
```

3. Results and Discussion

The proposed method is applied to the FaceForensics++ dataset, which is one of the most famous and widely used datasets in the detection field. FaceForensics++ dataset consists of 5000 real and fake videos taken from YouTube, found on the GitHub website [18] and the Kaggle website. Such videos have a compression rate factor of 23. Figure 2 shows different deepfake tools' frames belonging to the same real video. The preprocessing was implemented on a video dataset by Google Colab with GPU, while the creation model was done on the Kaggle notebook. Keras and Tensorflow libraries were used for implementing the CNN model. The train-test-split function found in the sklearn library was used to split the dataset. In the training phase, the epoch value was 50, the batch size was 32, the Adam optimizer was used to compile the proposed model with a learning rate of 0.001, and the loss function was categorical cross entropy which is the most commonly used for the classification.

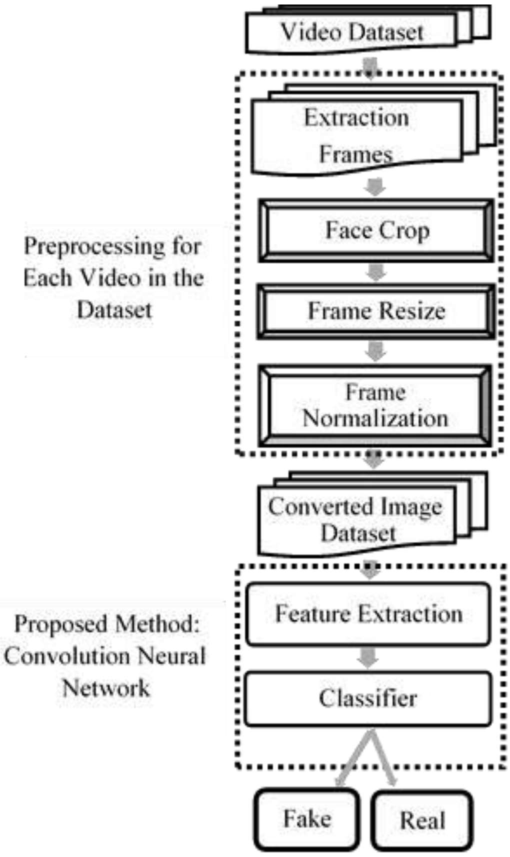


Figure 1. Proposed CNN based fake detection model

The resulting accuracy achieved by the proposed method are as follows:

i. 0.96 using FaceForensics++_Deepfake.
ii. 0.95 using FaceForensics++_Face2Face.
iii. 0.94 using FaceForensics++_Faceswap.
iv. 0.76 using FaceForensics++_NeuralTexture.

The training, validation, and testing accuracies and the training and validation loss for the dataset are shown in Table 2. It is noticeable that the FaceForensics++_NeuralTexture has the lower accuracy compared with the remaining sub-dataset. The deepfake methods differ in creating deepfake techniques; the NeuralTexture depends on the mouth area to create the deepfake, making it difficult to detect. While the other methods work on the whole face to create the deepfake that makes them easier to detect.

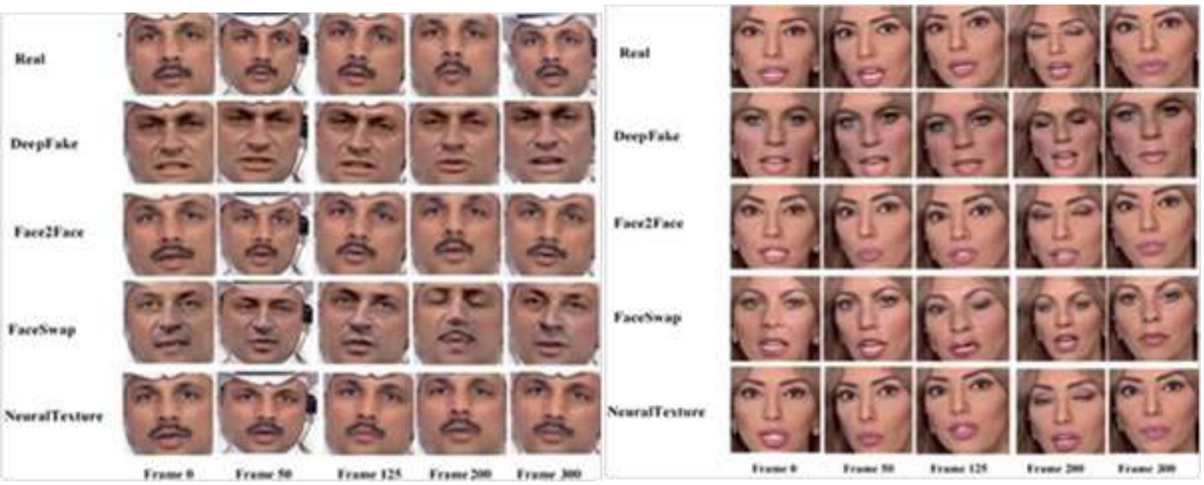
Table 2. Training and testing results of deepfake video detection of different datasets.

Dataset	Training Accuracy	Training loss	Validation accuracy	Validation loss	Testing Accuracy
FaceForensics++ _DeepFake	0.98	0.03	0.96	0.09	0.96
FaceForensics++ _Face2Face	0.98	0.03	0.95	0.09	0.95
FaceForensics++ _FaceSwap	0.98	0.04	0.95	0.1	0.94
FaceForensics++_ NeuralTexture	0.91	0.2	0.76	0.7	0.76

Figure 3 shows the training and validation accuracy behaviors, where the training curve explains how the model is trained, and the validation curve evaluates the model's training. Also, the training and validation accuracy curves on the first three deepfake methods, shown in Figure 3, increase rapidly in the first eight epochs, indicating that the network is learning fast. Then the curves increase quietly until they are flattened, which means that not required more epochs to train the model. While the training and validation accuracy curves are parallel, the validation accuracy values were close to the training accuracy, which ensures the well training of the proposed model. But, with the NeuralTexture method, the diagram showed a training accuracy curve had quietly learned, and the validation accuracy curve shows that the proposed model does not learn well with this type of data.

The acceptable results from the proposed deepfake video model were compared with a few other papers that used the same dataset. Table 3 shows a comparison between our results and other papers. In [14], they used only one deepfake method to evaluate their works. At [1-3-17], they used the dataset in general without splitting the fake videos into their deepfake methods.

We notice at [17] that they achieved too high accuracy, but they chose only 400 videos that maybe have a clear fake. The proposed model achieved a good accuracy when evaluated over thethree different deepfake methods, which means these methods are less efficient than the Neural Texture deepfake method. The Neural Texture method is hard to detect by our model and also hard to recognize by humans since their images are close to the real ones.



(a) (b)

Figure 2. (a) Random frames for the same video by different deepfake methods from FaceForensics++ dataset; (b) Random frames for the same video by different deepfake methods from FaceForensics++ dataset.

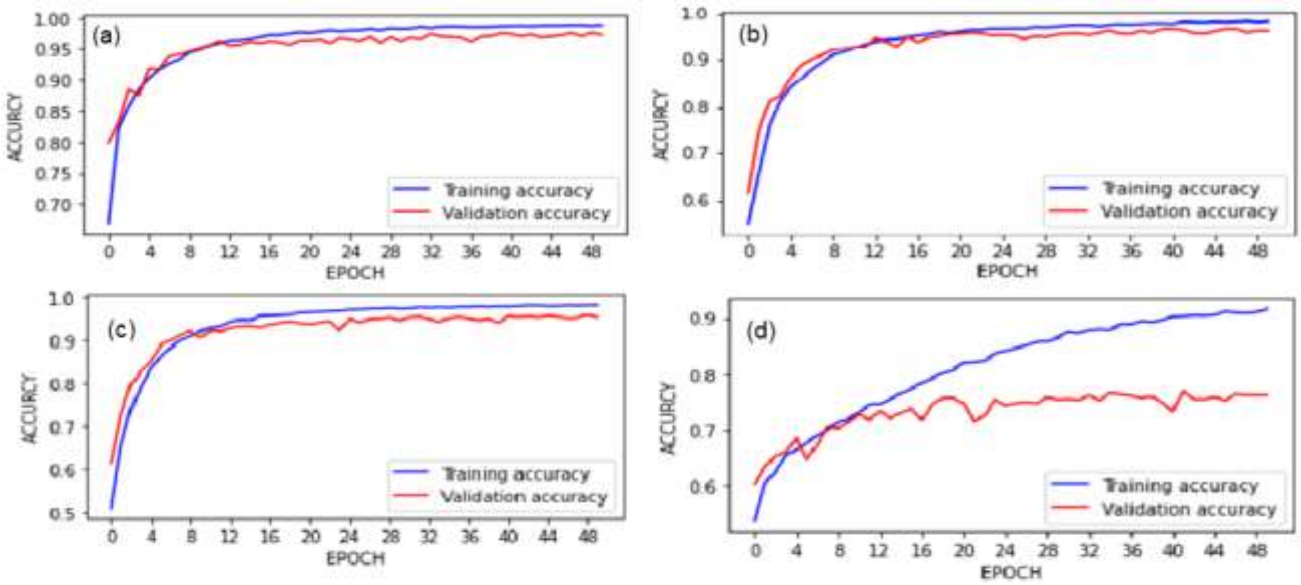


Figure 3. The training and validation accuracies for the different methods:
(a) Deepfake method; (b) Face2Face method; (c) Faceswap method; (d) Neuraltexture method.

Table 3. Comparison of present work results with other works.

Reference	Method	DeepFake Accuracy	Face2Face Accuracy	FaceSwap Accuracy	NeuralTexture Accuracy
[14]	CNN (Xception model) + classifier Network	96 %	–	–	–
	CNN (Inception model) + classifier Network	86 %	–	–	–
	CNN (ResNet50 model) + classifier network	88 %	–	–	–
[3]	Ensemble of CNNs	-	0.94	0.94	-
[1]	EfficientNet-V2	97.90 % (20K real image + 20K fake image)			
[17]	Machine learning	99.84% (using 400 videos: 200 real ones and 200 fake ones)			
Present work	CNN	0.96	0.95	0.94	0.76

4. Conclusions

This paper has presented a simple CNN model to detect deepfake video and evaluate the different deepfake methods. The experiment results were evaluated over FaceForensics++, which is the publicly available dataset. The proposed network achieved high accuracy on three deepfake methods (Deepfake, Face2face, and Faceswap). Thus, this trained network is efficient in detecting fake and real videos, and the dangers of the deepfake technique can be limited. Also, it was found that the NeuralTexture deepfake method was the best and most efficient deepfake method so that it can be the best choice in video games, movie production, and other fields. For future work, it is intended to improve the accuracy of the NeuralTexture deepfake method by using the RNN model.

Conflict of Interest: The authors declare no conflict of interest

References

[1] Deng, L.; Suo, H.; Li, D.; "Deepfake video detection based on EfficientNet-V2 network,". Comput. Intell. Neurosci, 2022, 2022.

[2] Nguyen, T.T.; Nguyen, Q. V. H.; Nguyen, D.T.; Nguyen, D. T.; Huynh-The, T. ; Nahavandi, S. et al.; "Deep learning for deepfakes creation and detection: A survey,". Comput. Vis. Image Underst., 223: 103525, 2022.

[3] Bonettini, N. ; Cannas, E.D. ; Mandelli, S.; Bondi, L.; Bestagini, P.; Tubaro, S.; "Video face manipulation detection through ensemble of cnns," in 2020 25th of ICPR,5012-5019,2021.

[4] Mahmud, B.U.; Sharmin, A.; "Deep insights of deepfake technology: A review". ArXiv.org:2105.00192, 2021.

[5] Yu, P.; Xia, Z.; Fei, J.; Lu, Y.; "A survey on deepfake video detection,". IET Biom., 10: 607-624, 2021.

[6] Abu-Ein, A.A.; Al-Hazaimeh, O.M. ; Dawood, A.M.; Swidan, A.I.; "Analysis of the current state of deepfake techniques-creation and

detection methods". IJEECS, 28: 1659-1667, 2022.

[7] Johansson, E.; "Detecting deepfakes and forged videos using deep learning". Master's Theses in Mathematical Sciences, 2020.

[8] Verdoliva, L.; "Media forensics and deepfakes: an overview". IEEE J. Sel. Top. Signal Process., 14: 910-932, 2020.

[9] Li, Y. ; Yang, X. ; Sun, P. ; Qi, H.; Lyu, S.; "Celeb-df: A large-scale challenging dataset for deepfake forensics," in Proceedings of IEEE/CVF: 3207-3216, 2020.

[10] Jafar, M. T.; Ababneh, M.; Al-Zoube, M.; Elhassan, A.; "Forensics and analysis of deepfake videos". 11th of ICICS, 053-058, 2020.

[11] Güera, D.; Delp, E.J.; "Deepfake video detection using recurrent neural networks". in 2018 15th IEEE international conference on AVSS,1-6, 2018.

[12] Masood, M.; Nawaz, M.; Javed, A.; Nazir, T.; Mehmood, A.; Mahum, R.; "Classification of Deepfake videos using pre-trained convolutional neural networks," ICoDT2, 1-6, 2021.

[13] Vamsi, V.V.V.N.S.; Shet, S.S.; Reddy, S.S.M.; Rose, S.S.; Shetty, S.R.; Sathvika, S., et al.; "Deepfake detection in digital media forensics". Glob. Transit., vol. 3,74-79, 2022.

[14] Mitra, A.; Mohanty, S.P.; Corcoran, P.; Kougianos, E.; "A novel machine learning based method for deepfake video detection in social media". IEEE International Symposium on iSES (Formerly iNiS), 91-96, 2020.

[15] Shende, A.; "Using deep learning to detect deepfake videos". TURCOMAT, 12: 5012-5017, 2021.

[16] Wodajo, D.; Atnafu, S.; "Deepfake video detection using convolutional vision transformer". ArXiv.org:2102.11126, 2021.

[17] Rana, M.S. ; Murali, B.; Sung, A.H.; "Deepfake Detection Using Machine Learning Algorithms". 10th International Congress on IIAI-AAI, 458-463, 2021.

[18] Nie{\ss}ner, A. R. o. a. D. C. a. L. V. a. C. R. a. J. T. a. M. (2019). Face{F}orensics++: Learning to Detect Manipulated Facial Images. Available: https://github.com/ondyari/FaceForensics

[19] Thies, J.; Zollhofer, M.; Stamminger, M.; Theobalt, C.; Nießner, M.; "Face2face: Real-time face capture and reenactment of rgb videos". Proceedings of the IEEE/ CVPR,2387-2395, 2016.