A Heterogeneous Feature Ensemble Learning based Deepfake Detection Method

Jixin Zhang*, Ke Cheng*, Giuliano Sovernigo†, Xiaodong Lin†
* School of Computer Science, Hubei University of Technology, China
{zhangjx, 1910301023}@hbut.edu.cn
† School of Computer Science, University of Guelph, Canada
{gsoverni, xlin08}@uoguelph.ca

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9838630

***Abstract*—The Deepfake technique can swap the face of a person with the face of another person in an image or a video which may cause a public security problem. Recently, researchers have focused on detecting deepfake images by deep learning. However some recent works have observed that detectors trained on images produced by one deepfake model perform poorly when tested on others. In this paper we propose to detect deepfake images through heterogeneous feature ensemble learning. We first extract gray gradient features, spectrum features and texture features from real and fake face images, then integrate them into an ensemble feature vector through a flatten process, and finally adopt a back-propagation neural network to train a deepfake detector with the feature vector. Experimental results show that our approach achieves better detection accuracy compared with several state-of-the-art deepfake detectors.**

***Index Terms*—Deepfake detection, Ensemble Learning, Heterogeneous Feature, Neural Network**

I. Introduction

Deepfake is a very popular technique which uses deep learning methods such as generative adversarial convolutional neural networks to create fake images which swap the face of a person with the face of another person [1]. Deepfake has found applications in settings such as movie filming, where the face of the stuntman can be replaced with that of the actor. However, the deepfake technique may also pose problems for public security. For example, a person may use deepfake to generate falsified news reports about public figures, which can have significant impact or backlash depending on the subject.

In this paper, we aim to detect fake face images generated by deepfake. Some researchers have proposed several approaches for deepfake detection. Durall et al. [2] proposed using power spectrum to detect fake images synthesized by generative deep convolutional neural networks. Up-convolution, which is always used by convolutional neural networks, will change the power spectrum. Wang et al. [3] proposed a generative adversarial network named proGAN to train a classification model for deepfake detection.

Although the aforementioned approaches can accurately detect fake faces generated by specific deepfake models, the detectors struggle to detect fake faces generated by various unknown deepfake models. Some recent works have observed that detectors trained on images produced by one GAN architecture perform poorly when tested on others and these detectors might not generalize when tested on new data due to the data bias [3].

To address these problems, we propose deepfake detection through the integration of different features to improve detection accuracy, since we found that no single feature can achieve optimal fake detection performance.

However, challenges still remain even when using feature integration. On one hand, some features may cause over-fitting problems (detection accuracy will be reduced when detecting the fake faces generated by other unknown models), such as auto-encoding features. The over-fitting problem has significant impact on the performance of the ensemble model. To overcome it, we need to extract well used features that can collaboratively improve the performance of deepfake detection. On the other hand, the features for integration may be heterogeneous, the representations of the features need to be unified.

In this paper we first extract three different features: the facial landmark points, the facial spectrum and texture. Since the representations of these features are quite different. Next, we take advantage of the histogram of gray gradients to represent facial landmark points, extract the high-level features of co-occurrence matrix to represent facial texture, and use a flatten process to integrate them for ensemble learning before finally sending them into a back-propagation neural network classifier to distinguish the deepfake images from real images. The experimental results show that our approach can improve detection accuracy compared with the state-of-the-art methods when detecting the fake face images produced by other unknown models.

The contributions of our work can be summarized as follows:

- We propose a heterogeneous feature ensemble learning based deepfake detection method to detect fake face images generated by various deepfake models.
- Our approach proposes the extraction of three heterogeneous features and integrates them to improve the accuracy and generalization of deepfake detection.
- To evaluate our approach, we first train our detector on the samples generated by only one of the deepfake models, and then test the samples generated by various deepfake models. The experimental results show that our

approach can achieve better detection accuracy (97.04%) compared with several state-of-the-art methods.

The remainder of the paper is organized in as follows: Section II introduces the related works about deepfake detection. Section III analyzes the selection of heterogeneous features and Section IV introduces our heterogeneous feature ensemble learning based deepfake detection method. In Section V, we show the performance evaluation and comparison of our method. Finally, we conclude our work and examine future research directions in Section VI.

II. Related Works

In this section, we introduce other related research topics in facial manipulation by deepfake and deepfake detection.

A. Facial manipulation

In recent years, researches have begun preferring the use of generative deep (convolutional) neural networks for facial manipulation. Karras et al. [4] proposed a generative adversarial network named styleGAN to create images of entirely non-existent faces. Zhu et al. [5] proposed a GAN-based face-swapping method based on a generative adversarial network named CycleGAN. Other tools such as FaceSwap [6] and DeepFakes [7] can replace the face of one person in a video with the face of another person. Choi et al. [8] proposed a technique named starGAN which can be used to modify some attributes of a face, such as the colour of the hair or the skin, the gender, the age, adding glasses, etc. Thies et al. [9] proposed a generative adversarial network based technique named Face2Face to modify the facial expressions of people in the images. Additionally, there are some other famous deepfake techniques such as proGAN [10] and SAGAN [11]. The deepfaked faces have reached the point where they look like the real faces.

B. Fake Detection

Since attackers can use the deepfake technique to generate fake news which may pose a public security risk, researchers have been working to propose solutions to detect deepfake images. Some of them preferred to analyze the internal generative adversarial network pipeline to find the different artifacts between fake and real images, such as [2], [3]. Most of the researchers preferred to train a classification model by machine (deep) learning to distinguish fake and real images. Nataraj et al. [12] first computed the co-occurrence matrix of fake and real images as the inputs, and then used a multi-layer deep convolutional neural network to classify fake and real images. Guarnera et al. [13] proposed a fake detection system based on the analysis of the convolutional traces. Marra et al. [14] proposed a multi-task incremental learning method to detect fake images generated by new types of generative adversarial network. Sabir et al. [15] proposed a recurrent convolutional neural network for fake video detection.

Although above mentioned approaches can detect fake faces generated by a specific deepfake model, some recent works observe that the detectors perform poorly when detecting fake faces generated by various unknown deepfake models [1].

III. Feature analysis

Because different features may reflect different properties of images, so we aggregated several features usually used in digital image processing in order to find which of them perform well in detecting deepfake images. In this section, we analyze the performance of different features and select three of them which will be used for the later ensemble learning.

Since some recent works propose the use of convolutional neural networks (CNN) to train a classification model with prepared real and fake images, we first consider the neural network features such as the convolutional auto-encoder feature, CNN hidden layer feature, etc. to detect deepfake facial images. However, we observe that while these neural network features may achieve 100% of accuracy when testing images generated by a specific deepfake model, they perform poorly (even as low as 50%) when training on images produced by one deepfake model but testing on others.

Facial landmark point features, which represent the geometric characteristics of faces is widely used in face detection and recognition. It is believe that the noise in deepfake images may change the landmark point features in natural images. So in this paper we try to extract the gray gradients of facial landmark points to detect deepfake images. We observe that the facial landmark point feature performs well in some fake images, but performs poorly when training on images produced by proGAN and testing on other models.

The spectrum feature and texture feature mentioned in [2], [12] both perform well from our observation. It is worth mentioning that in this paper we extract several high-level features from various co-occurrence texture matrices to optimize the feature representation.

Because of their performance, we selected facial landmark point feature, spectrum feature, and texture feature as features used in the ensemble learning.

IV. Methodology

A. The overview of our method

In this paper, we propose a method to detect deepfake face images by first extracting heterogeneous features included gray gradient features, spectrum feature and texture feature from real and fake face images, then integrating them into a ensemble feature vector by a flatten process, and finally sending the feature vector to a back-propagation neural network to train a classification model as the deepfake detector. The architecture of our approach is shown in Fig. 1.

B. Heterogeneous feature extraction

Since some features such as auto-encoder feature, neural network feature may not be generalized in detecting various deepfake models due to the over-fitting problem. Here we propose to extract three types of features which perform well together. All of the features are shown in Table I.

- **Gray Gradient Feature (GGF):** The histogram of gray gradients for each facial landmark point. For a given face image, we first extract its 68 facial landmark points by

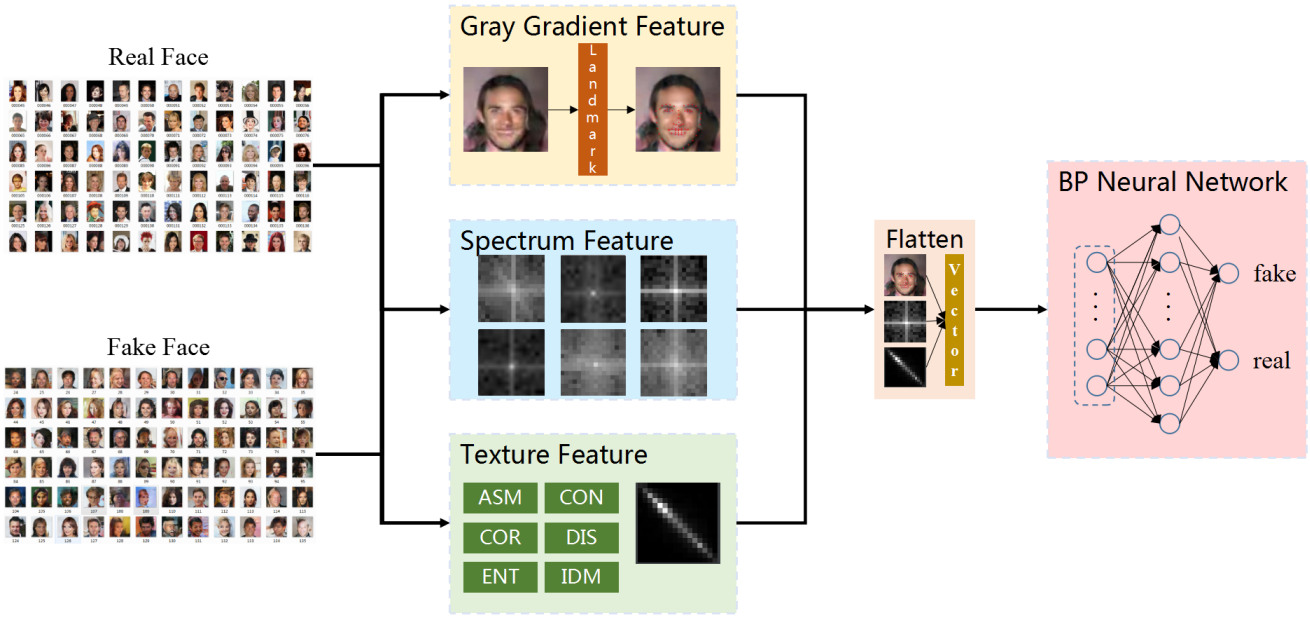

Fig. 1. The architecture of our approach

[16], then compute the gray gradients in four directions for each facial landmark point according to Eqs. (1), (2), (3), (4). Finally we integrate the gray gradients into a vector called histogram of gray gradients according to Eq. (5), where GGF is the gray gradient feature, (x,y) is the x, y axis of the facial landmark point, G(x,y) is the gray value of (x,y), HGGGF is the histogram of gray gradients of all of the facial landmark points.

$$GGF(x,y)_{x+} = |G(x+1,y) - G(x,y)| \quad (1)$$

$$GGF(x,y)_{x-} = |G(x,y) - G(x-1,y)| \quad (2)$$

$$GGF(x,y)_{y+} = |G(x,y+1) - G(x,y)| \quad (3)$$

$$GGF(x,y)_{y-} = |G(x,y) - G(x,y-1)| \quad (4)$$

$$HGGGF(x,y)_{(x,y)\in landmark} = [GGF(x,y)_{x+,x-,y+,y-}] \quad (5)$$

- **Spectrum Feature (SF):** The 2-D spectrum matrix of face images by using Fast Fourier Transform (FFT) [17]. Since most of deepfake models contain up-convolutional process which leads to the difference of spectrum between real and fake images [2], we select the spectrum feature by FFT as one of our features. For a given face image, we extract its 2-D spectrum matrix according to Eq. (6), where SF(k,l) is the spectrum matrix, and k and l are the sampling frequency.

$$SF(k,l) = \sum_{x=0}^{M}\sum_{y=0}^{N} G(x,y)\cdot e^{-2\pi i\cdot\frac{x\cdot k}{M}}\cdot e^{-2\pi i\cdot\frac{y\cdot l}{N}} \quad (6)$$

- **Texture Feature (TF):** We first compute the gray co-occurrence matrices with different gaps (1, 2, 8, 16) and angles (0° , 90°, 180°, 270°) [18] of a given face image, then use some feature operators included angular second moment (ASM), contrast (CON), correlation (COR), dissimilarity (DIS), entropy (ENT), and inverse different moment (IDM) to extract the high-level features of the gray co-occurrence matrix as the texture features according to Eqs. (7), (8), (9), (10), (11), (12), (13), (14), (15), (16), where comat(i,j) is the gray co-occurrence matrix of the face image.

$$ASM = \sum_{i=1}^{N}\sum_{j=1}^{N} comat(i,j)^2 \quad (7)$$

$$CON = \sum_{i=1}^{N}\sum_{j=1}^{N} (i-j)^2\cdot comat(i,j) \quad (8)$$

$$COR = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{i\cdot j\cdot comat(i,j) - u_i\cdot u_j}{s_i\cdot s_j} \quad (9)$$

$$u_i = \sum_{i=1}^{N}\sum_{j=1}^{N} i\cdot comat(i,j) \quad (10)$$

$$u_j = \sum_{i=1}^{N}\sum_{j=1}^{N} j\cdot comat(i,j) \quad (11)$$

$$s_i^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} (i-u_i)^2\cdot comat(i,j) \quad (12)$$

$$s_j^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} (j-u_j)^2\cdot comat(i,j) \quad (13)$$

$$DIS = \sum_{i=1}^{N}\sum_{j=1}^{N} |i-j|\cdot comat(i,j) \quad (14)$$

$$ENT = \sum_{i=1}^{N}\sum_{j=1}^{N} comat(i,j) \cdot log(comat(i,j)) \quad (15)$$

$$IDM = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{comat(i,j)}{1+(i-j)^2} \quad (16)$$

TABLE I
THE DESCRIPTION OF HETEROGENEOUS FEATURES

Feature	Description
GGF	The histogram of gray gradients for each facial landmark point
SF	The 2-D spectrum matrix of face images by using fast Fourier transform
ASM	The angular second moment of the gray co-occurrence matrix
CON	The contrast of the gray co-occurrence matrix
COR	The correlation of the gray co-occurrence matrix
DIS	The dissimilarity of the gray co-occurrence matrix
ENT	The entropy of the gray co-occurrence matrix
IDM	The inverse different moment of the gray co-occurrence matrix

C. Ensemble learning for deepfake detection

After we extract above proposed features, We then integrate the features into an ensemble vector by a flatten process. The process converts multi-dimension structured features into a 1-dimension vector. Next, we send the ensemble vector to a back-propagation neural network to train a model to classify real or fake face images.

The back-propagation neural network consists of three layers: an input layer, a hidden layer and an output layer. The input layer fully connects the hidden layer and the hidden layer fully connects the output layer using the ReLU function as the activation function according to Eq. (17), where w is weight of the connection between two layers and x is the element in the ensemble vector. We use cross entropy error (according to Eq. (18)) as the loss function of the neural network, where y is the label value of the input sample and h(w · x) is the confidence score of the output layer. Here we use ReLU and cross entropy error in order to prevent over-fitting and improve the speed of convergence.

$$ReLu = max(0, \sum w \cdot x) \quad (17)$$

$$Loss = -ylog(h(w \cdot x)) - (1-y)log(h(w \cdot x)) \quad (18)$$

For training our deepfake detection model, we adopt gradient decent method according to Eq. (19) to update the weights of the connections between adjacent layers. For detecting a deepfake facial image, we extract the ensemble vector of the heterogeneous features and input it into the pre-trained deepfake detection model to get the decision result.

$$\triangle w = \frac{\delta Loss}{\delta w} \quad (19)$$

Fig. 2. The examples of deepfake images generated by various models

V. EXPERIMENTS

In this section, we first show the experimental setup, the data set, the deepfake models and the validation method for evaluations. Then we show the detection accuracy improvement of our approach compared with several state-of-the-art methods. And finally we show the ensemble features can indeed improve the accuracy compared with different feature combinations.

A. Experimental setup

We implemented all of the methods in the same environment and configuration. The CPU used in the environment is an Intel 17-9750H@2.60GHz, with 16.0GB of RAM, and the operating system is Windows 10. Our heterogeneous feature ensemble learning model is developed in the Python programming language and trained on a 4.0GB GTX1650 GPU.

B. Data set

We evaluate our approach on a data set of facial images from celebA [19] which contains 5,000 real faces. We randomly select 80% of the facial images for training and the remaining 20% of them will be used for testing. The training data and the testing data are different facial images in CelebA. The dimensions of the facial images is 178 × 218 × 3.

C. Deepfake model and Validation

We test our approach with various well known deepfake models including proGAN [10], SAGAN [11], DCGAN [20], BEGAN [21] and EBGAN [22]. The examples of deepfake images (the size is 64 × 64) generated by various models are shown in Fig. 2. The deepfake images are 64 × 64 pixels.

During training, only one set of fake images produced by a single deepfake model are used. During testing, we use the falsified images from the other four models. We repeatedly select the training faces and the testing faces for 10 times and compute the average detection accuracy for each evaluation.

D. Performance evaluation

To evaluate the performance of our Ensemble Learning (EL) approach, we compare our work with several state-of-the-art deepfake detection methods such as Spectrum [2], [23] and Co-occurrence [12]. Durall et al. [2] proposed a power spectrum classifier based deepfake detection method through

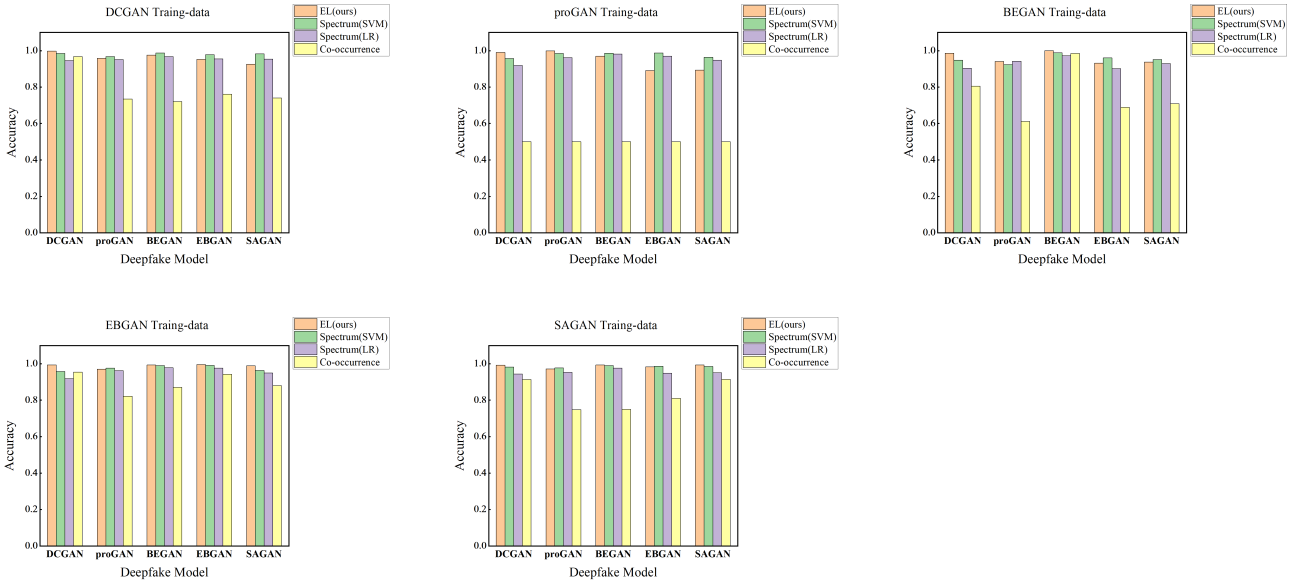

Fig. 3. The accuracy comparison between our approach and the state-of-the-art methods

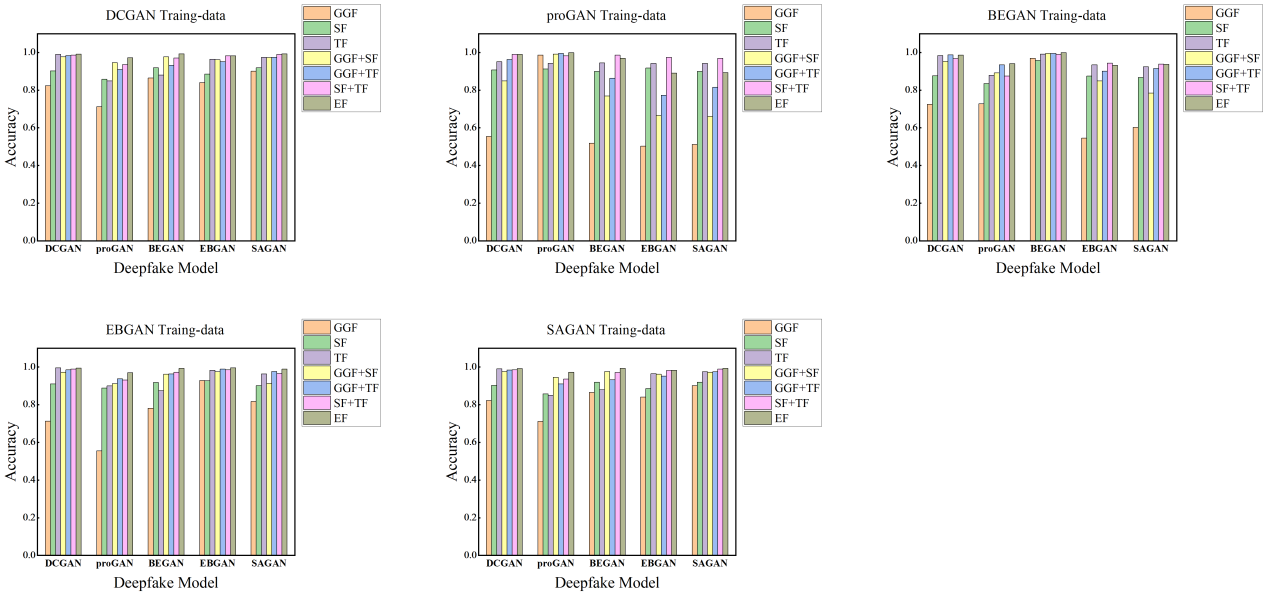

Fig. 4. The accuracy comparison between different feature combinations

observation of the fingerprint of generative convolutional neural networks. The code of [2] is shown in [23]. Nataraj et al. [12] proposed the use of a deep convolutional neural network to detect the co-occurrence matrix of fake images.

Fig. 3 shows the accuracy comparison between our Ensemble Learning (EL) approach and the other state-of-the-art methods (Spectrum (SVM), Spectrum (LR) and Co-occurrence). The experimental results show that:

- Our ensemble learning based approach achieves 97.04% detection accuracy when training with fake images generated by only one deepfake model and testing with fake images generated by various deepfake models. In almost 40% testing cases, our approach achieves more than 99% detection accuracy.
- Both Spectrum (SVM) and Spectrum (LR) perform well and Co-occurrence usually performs worse in most cases. That means the selected features have a significant effect on detection accuracy.
- When detecting the fake images generated by various deepfake models, in most cases our approach performs best compared with the state-of-the-art methods.

E. Feature analysis

To show the features we integrated can improve the accuracy of detecting deepfake images, we compared the accuracy of different feature combinations, as shown in Fig. 4. The feature combinations include single GGF, single SF, single TF, GGF-SF (the features integrated by GGF and SF), GGF-TF, SF-TF and EF (ours). The experimental results show that:

- Compared with different feature combinations, the integration of GGF, SF and TF achieves the highest accuracy in most cases.
- Single TF performs better than the other two single features but always performs worse than our ensemble features. The combinations of two features perform better than single features.
- When training with proGAN and testing with other deepfake models, our ensemble learning model cannot always achieve the best accuracy due to the poor performance of GGF feature, but can still improve the accuracy when combining with the other features.

VI. Conclusion

In this paper we propose an ensemble learning based deepfake detection method to improve the detection accuracy and model generalization, which integrates heterogeneous features that includes Gray Gradient Feature, Spectrum Feature and Texture Feature. The experimental results show that our approach can achieve better detection accuracy compared with several state-of-the-art methods. In future work, we will not only apply our approach to deepfake detection but will also apply it to many other image manipulation detection and forensic techniques..

Acknowledgment

This work is partially supported by the National Natural Science foundation of China under Grant No. 62002106, the Research Foundation of Education Commission of Hubei Province No. Q20201408 and the Research Foundation of Hubei University of Technology No. BSQD2020066.

References

[1] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez and Javier Ortega-Garcia. DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection. Information Fusion, 2020, vol.64, pp. 131-148.

[2] Ricard Durall, Margret Keuper and Janis Keuper. Watch your Up-Convolution: CNN Based Generative Deep Neural Networks are Failing to Reproduce Spectral Distributions. IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[3] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens and Alexei A. Efros. CNN-generated images are surprisingly easy to spot ... for now. IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[4] Tero Karras, Samuli Laine and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[5] Jun-Yan Zhu, Taesung Park, Phillip Isola and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. IEEE/CVF International Conference on Computer Vision, 2017.

[6] FaceSwap. https://github.com/MarekKowalski/FaceSwap. 2021.

[7] Deepfake. https://github.com/deepfakes/faceswap. 2021.

[8] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim and Jaegul Choo. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Imageto-Image Translation. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[9] Justus Thies and Michael Zollhöfer and Marc Stamminger and Christian Theobalt and Matthias Nießner. Face2face: Real-Time Face Capture and Reenactment of RGB Videos. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[10] Tero Karras, Timo Aila, Samuli Laine and Jaakko Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. International Conference on Learning Representations, 2018.

[11] Han Zhang, Ian Goodfellow, Dimitris Metaxas and Augustus Odena. Self-Attention Generative Adversarial Networks. International Conference on Machine Learning, 2019.

[12] Lakshmanan Nataraj, Tajuddin Manhar Mohammed, B. S. Manjunath, Shivkumar Chandrasekaran, Arjuna Flenner, Md Jawadul Hasan Bappy and Amit Roy-Chowdhury. Detecting GAN Generated Fake Images Using Co-Occurrence Matrices. Electronic Imaging, 2019, vol. 5, pp. 1–7.

[13] Luca Guarnera, Oliver Giudice, Sebastiano Battiato. DeepFake Detection by Analyzing Convolutional Traces. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020.

[14] Francesco Marra, Cristiano Saltori, Giulia Boato, Luisa Verdoliva. Incremental Learning for the Detection and Classification of GAN-Generated Images. IEEE International Workshop on Information Forensics and Security, 2019.

[15] Ekraam Sabir, Jiaxin Cheng, Ayush Jaiswal. Recurrent Convolutional Strategies for Face Manipulation Detection in Videos. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019.

[16] Facial landmark. http://dlib.net, 2021.

[17] Numpy.FFT. https://numpy.org/doc/stable/reference/ generated/numpy.fft.fft2.html, 2021.

[18] Scikit-mage. https://scikit-image.org/docs/dev/api/skimage.feature.html, 2021.

[19] CelebA. http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html, 2021.

[20] Alec Radford, Luke Metz and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv:1511.06434, 2016.

[21] David Berthelot, Thomas Schumm and Luke Metz. BEGAN: Boundary Equilibrium Generative Adversarial Networks. arXiv:1703.10717, 2017.

[22] Junbo Zhao, Michael Mathieu and Yann LeCun. Energy-based Generative Adversarial Network. arXiv:1609.03126, 2017.

[23] DeepFake Detection. https://github.com/cc-hpc-itwm/DeepFakeDetection, 2021.