Hindawi
Security and Communication Networks
Volume 2021, Article ID 2482942, 11 pages
https://doi.org/10.1155/2021/2482942

Research Article

Deepfake Detection Method Based on Cross-Domain Fusion

Fang Sun,[1] **Niuniu Zhang**,[1] **Pan Xu**,[1] **and Zengren Song**[2]

[1]*School of Computer and Information Technology, LiaoNing Normal University, Dalian 116023, China*
[2]*National Computer Network Emergency Response Technical Team, Coordination Center of China, China*

Correspondence should be addressed to Fang Sun; sunfang@lnnu.edu.cn

Received 29 July 2021; Revised 11 October 2021; Accepted 13 October 2021; Published 24 November 2021

Academic Editor: Xin Liu

In recent years, despite its wide use in various fields, deepfake has been abused to generate hazardous contents such as fake movies, rumors, and fake news by manipulating or replacing facial information of the original sources and, thus, exerts huge security threats to the society. Facing the continuous evolution of deepfake, research on active detection and prevention technology becomes particularly important. In this paper, we propose a new deepfake detection method based on cross-domain fusion, which, on the basis of traditional spatial domain features, realizes the fusion of cross-domain image features by introducing edge geometric features of the frequency domain and, therefore, achieves considerable improvements on classification accuracy. Further evaluations of this method have been performed on publicly deepfake datasets, and the results show that our method is effective particularly on the Meso-4 DeepFake Database.

1. Introduction

With the rapid development of the Internet of Things (IoT), society has entered a new era [1]. The IoT has many distinct advantages, such as security, real time, automation, embeddedness, interoperability, and interconnection. In the context of the Internet of Things and 5G, the speed of the network is constantly improving, which not only brings convenience to people, but also facilitates information fraud [2, 3]. At the same time, deep learning is booming as a new technology. The widespread application of deep learning technology has not only brought unprecedented innovation in various fields, but also created convenient criminal conditions for offenders. For example, the deepfake technique, which has caused bad influence to the society recently, can not only generate realistic fake images, videos, and audio contents, but also forge evidence for electronic crime. The abuse of this technique has seriously threatened the network information and data security of both individuals and the society. Therefore, the research of deepfake detection technology is of great significance to ensure the authenticity of video, image, and audio transmitted in the network.

In the process of generating fake face image, to ensure fidelity, fuzzy functions are usually used. To process the face region to match the background region, however, such operations lead to edge differences between the face and background regions in the forged image, resulting in inconsistent resolution on both sides of the boundary. This paper introduces a new method of deepfake detection based on cross-domain fusion. Our method uses a network to gain the image's spatial domain feature vectors and extracts the edge geometric features in frequency domain, in order to obtain a vector set containing both high-level semantic features and low-level edge features of the image.

The main contributions of this paper are as follows:

(1) We propose a new deepfake detection approach to improve the detection accuracy, which fuses spatial domain features extracted from the image and features extracted from the frequency domain to capture more detailed forgery trails and get more comprehensive face features.

(2) We further propose a better fusion strategy. On the one hand, the spatial domain features of the full connection layer in the meso-net network are extracted to ensure the high-level semantic features of the image; on the other hand, frequency domain features of the face

image are extracted to ensure the low-level texture features. Thus, we obtain cross-domain features through the fusion of two feature vectors, which includes not only frequency features of the edge geometric features, but also spatial features of the face image. Finally, we test this feature set with a classifier.

(3) In order to analyze the effectiveness of the detection method, we evaluate the performance of the model on the deepfake datasets and compare it with other deepfake detection methods. At the same time, we also evaluate the performance of the algorithm on the cross-datasets. Our detection model shows effectiveness and superiority in these experiments.

The structure of the paper is as follows. In Section 1, we briefly overview the development status of deepfake and the contributions of our work; Section 2 describes related works. In Section 3, we introduce the architecture, innovation, and advantages of our deepfake detection method in detail; in Section 4, we present our experimental setup and report our detection performance results; Section 5 gives a summary and introduces future work.

2. Related Work

2.1. Deepfake. The word "deepfake" is the combination of "deep learning" and "fake." It is mainly the product of neural network in machine learning and particularly refers to the forgery of image, video, and audio generated by Generative Adversarial Network (GAN). Deepfakes can promote the development of entertainment, cultural exchange, and education industry, which can improve not only the teaching level in the field of education, but also the quality of life. However most of the time, deepfake is used to generate fake news and forge electronic evidence, which misleads the public and disturbs social order. This technology has become the most advanced means of network attack. Deepfake can produce fake images/videos that are difficult to distinguish with human eyes and results in social chaos. For example, Trump's forged video, together with the release of former President Barack Obama's fake speech video, has caused public panic about deepfake [4].

The creation of deepfakes mainly falls into the following categories, i.e., face synthesis, facial reenactment, face replacement, and face editing (such as hair color and skin color). According to the different focus of forgery technology, four types of forged images are employed as shown in Figure 1. Figure 1 (a) shows a computer-generated face synthesis, Figure 1 (b) is generated by editing the real face image attributes with a beauty camera, Figure 1 (c) is from Celeb-DF dataset, and Figure 1(d) is from the FaceForensics++ dataset [5]. Obviously, human eyes can hardly distinguish between the real and fake images. If this technology is abused to spread false information to the society, the public may be misled, and the related consequences might be unknown.

2.2. Deepfake Detection Technology. According to the human five senses, deepfake can be divided into visual detection technology and auditory detection technology. Visual detection includes image detection and video detection. Image detection is based on either traditional digital image forensics technology, convolution neural network model, or authenticity difference of generation principle. Video detection, however, is more complex. One type of detection is based on (a) temporal characteristics of cross video frame group, (b) visual artifacts within video frame, or (c) emerging technologies such as block chain. Another type of video detection is to extract each frame from the video and convert it into a static image and, therefore, convert dynamic deepfake video detection to static deepfake image detection. Auditory detection mainly detects the difference between the real and fake biological signals such as speech speed, voiceprint, and spectrum distribution in audio [6].

To deal with the potential threat of deepfake, researchers are exploring the classification method of real and fake images. As deepfake method is a special branch of traditional image tampering, early detection methods learn from the traditional forensics methods. Recently, people began to study the generation process of deepfake and reverse the use of deep neural network to detect forged images.

According to the principle of traditional detection methods, Zhou et al. [7] proposed a two-stream CNN network for face forgery detection. Tan et al. [8] proposed a feature set to capture the statistical information of color images and then identify forged images. Cozzolino et al. [9] extracted the camera fingerprint of the deep forgery image for the detection task and proposed a camera model fingerprint method called "noise print." Hu Yongjian proposed a network using image segmentation to detect deep false face video [10].

McCloskey et al. [11] show that GAN has obvious differences in color and saturation with the actual camera in color processing to distinguish real and fake images. Nataraj et al. [12] proposed a method using cooccurrence matrix and deep learning to detect forged images generated by GAN. Zhang et al. [13] proposed a GAN simulator, AutoGAN, which can simulate artifacts generated by a common channel shared by multiple GAN models. According to Mohammed Akram Younnus, if a blur function is added in the generation process of deep forgery video to reach the level of real video, the detection technology should be able to distinguish the authenticity by detecting the blur degree between the face and background boundary [14].

Christian Szegedy et al. [15] extended the network by using proper convolution kernel and regularization and proposed the InceptionV3 network. Darius Afchar et al. proposed two networks. One of them is Meso-4, which is composed of a series of continuous convolution and pooling layers. At the same time, in order to improve the generalization ability of the model, the convolution layer uses ReLu excitation function to normalize the input to avoid gradient disappearance. The other network is MesoInception4 network, which replaces the first two layers of Meso-4 network by introducing a series of induction modules in [16]. It uses $3 * 3$ dilated convolutions [17] rather than $5 * 5$ convolutions of the original module to avoid high semantics. Moreover, multiple convolution layers with different core sizes are stacked to increase the function space of the optimization model and realize parallel detection [18].

Figure 1: Deepfakes classification. (a) Face synthesis. (b) Face editing. (c) Face replacement. (d) Facial reenactment.

Run Wang et al. [19] proposed using MNC average neurons to monitor the behavior of neurons in each layer and extract the low-level features of each layer to make the detection more accurate. Nguyen et al. [20] proposed an architecture based on capsule network to detect deep forgery images. Ekraam Sabir et al. [21] designed a best strategy for combining variations in some networks along with domain specific face preprocessing techniques for forgery video detection. Mattern et al. [22] used the missing facial texture features and facial markers in eyes and teeth to generate feature vectors to judge the authenticity of the image. Li and Lyu only generate limited resolution images based on deepfake algorithm, and the fake image needs to be further deformed to match the real face.

In the real video, this deformation leaves a unique artifact in the forged video. Therefore, they propose to detect the unique artifact in the forged video to judge the authenticity of the video [23].

Stehouwer et al. [24] proposed using attention mechanism to process and improve the feature map of classification task instead of multitask learning method of detecting deep fake image and predicting forgery region at the same time. Sohail Ahmed Khan et al. [25] proposed a fusion network, using VGG16, InceptionV3, and XceptionNet parallel tests and average prediction results to obtain the final prediction result.

3. Method

In this paper, we propose a deepfake detection method based on cross-domain fusion, which utilizes frequency domain features to extract edge geometric features of the image (blue box in Figure 2); at the same time, spatial domain features of the image are integrated to extract the high-level features of the image under neural network (green box in Figure 2). To get more comprehensive facial features, low-level features and high-level features are fused to form feature vectors of the image (red box in Figure 2). The detailed model structure is illustrated in Figure 2.

3.1. Cross-Domain Features Extraction. Through the analysis of the image in frequency domain, we can see that the edge of the face is fuzzy, and the background is clear in the transformed forged face image. It clearly shows that the geometric features of the face edge in frequency domain provide a significant contribution to deepfake detection (Figure 4). The bottom network layer of neural network extracts low-level features (edge features, geometric features), the middle network layer extracts middle-level features (texture features), and the top network layer extracts high-level features of the image, the essence of which is the recombination of middle-level features, which can highly summarize the image attributes. It can be seen that, with the increase of neural network layers, low-level features gradually disappear, and their proportion in the final feature set is extremely low. On the other hand, neural network with less layers produces incomplete generalization of image attributes. In this paper, Haar transform is used to extract low-level features of face image such as edge features to ensure the contribution of low-level features. At the same time, a four-layer neural network is used to extract high-level features of image to ensure the influence strength of high-level features. The fusion of the two can achieve a comprehensive face image feature set with both low-level frequency domain features and high-level spatial domain features.

A sized $m \times n$ image I is resized to $r \times c$ size. To ensure generalization of the detection model, according to the convex optimization theory and data probability distribution theory, we know that the data centralization conforms to data distribution law and is easier in obtaining the generalization effect after training. In this paper, we define the image standardization operation, so that data can be

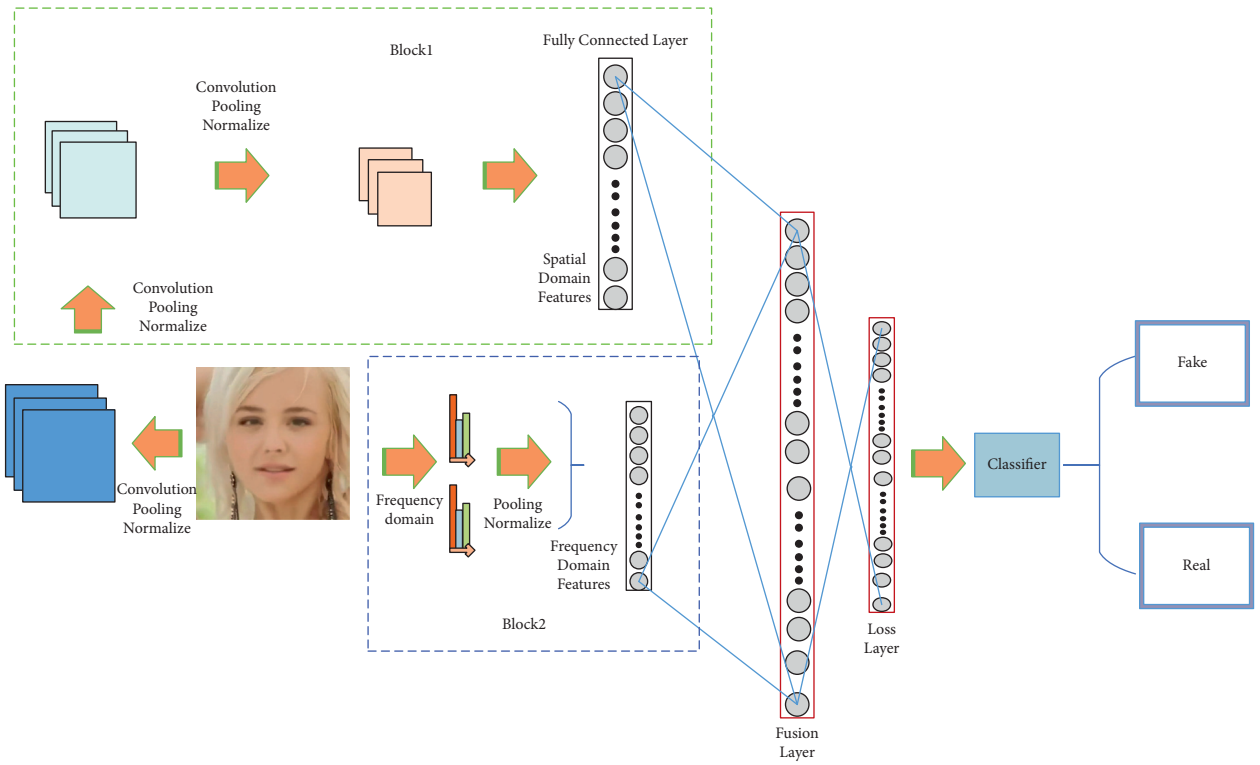

FIGURE 2: Cross-domain detection model. The model consists of three parts: extracting spatial domain features (Block1), extracting frequency domain features (Block2), and fusion layer. The detailed diagram of cross-domain network structure is illustrated in Figure 3.

centralized by means of deaveraging. X_{rc} is defined as the image matrix, and the image I data standardization formula is as follows:

$$\begin{aligned} \hat{X_{rc}} &= \frac{X_{rc} - \text{mean}(I_{rc})}{\text{std}[I_{rc}]}, \\ \text{std}[I_{rc}] &= \max\left(\sigma, \frac{1.0}{\sqrt{N}}\right), \end{aligned} \tag{1}$$

where $\text{mean}(I_{rc})$ is the mean value of the image, $\hat{X_{rc}}$ is the normalized image matrix, σ is the standard deviation, and N is the number of pixels of the image I_{rc}.

As the closer the distribution of data learned from the model and that of the training data are, the narrower the gap between the prediction label and the actual label would be, this paper uses cross entropy loss function to adjust the model parameters and, at the same time, avoids overfitting of the model and effectively improves the training efficiency. The loss function is defined as follows:

$$L = -\left[y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})\right], \tag{2}$$

where y refers to the real label and $\hat{y}$ means the model prediction label.

In this paper, Adam optimization algorithm based on adaptive moment estimation [26] is used to adjust the learning rate of each epoch of the model due to its nature of lower memory requirement and higher computational efficiency. Meanwhile, the algorithm is able to adaptively adjust learning rate parameters according to training situation and, therefore, is more efficient than the traditional random gradient descent algorithm.

We define X_{CA}, X_{CH}, X_{CV}, and X_{CD} as the approximation of the original image, the edge texture information along horizontal direction, the edge texture information along vertical direction, and the edge texture information along diagonal direction, respectively. $X_{\alpha\in\{CA,CH,CV,CD\}}(i, j)$ is the pixel value of i row and j column in each direction.

The definition is as follows:

$$\begin{aligned} X_{CA}(i, j) &= X_{rc}(2i - 1, 2j - 1) + X_{rc}(2i - 1, 2j) + X_{rc}(2i, 2j - 1) + X_{rc}(2i, 2j), \\ X_{CH}(i, j) &= -X_{rc}(2i - 1, 2j - 1) + X_{rc}(2i - 1, 2j) + X_{rc}(2i, 2j - 1) + X_{rc}(2i, 2j), \\ X_{CV}(i, j) &= -X_{rc}(2i - 1, 2j - 1) - X_{rc}(2i - 1, 2j) + X_{rc}(2i, 2j - 1) + X_{rc}(2i, 2j), \\ X_{CD}(i, j) &= X_{rc}(2i - 1, 2j - 1) - X_{rc}(2i - 1, 2j) - X_{rc}(2i, 2j - 1) + X_{rc}(2i, 2j). \end{aligned} \tag{3}$$

In order to ensure that the extracted edge texture information vectors in frequency domain can be fused with the traditional spatial features, we map the three edge texture information vectors of image into a feature matrix by two norms. The mapping function is defined as follows:

$$X_{rc}^{\wedge} = \left\| X_{CH} + X_{CV} + X_{CD} \right\|_2 = \sqrt{X_{CH}{}^2 + X_{CV}{}^2 + X_{CD}{}^2}. \tag{4}$$

At the same time, $X_{rc}^{\wedge}$ is normalized [27] and pooled, and the range of matrix pixel value is limited within a fixed range to guarantee the correct fusion with spatial characteristics for subsequent network processing. We define f as the pixel value of $X_{rc}^{\wedge}$, and $f_{O\in\{R,G,B\}}(X_{rci\times j}^{\wedge})$ as the pixel value of column j in row i of the matrix, and the normalization formula for $X_{rc}^{\wedge}$ is as follows:

$$f_{O\in\{R,G,B\}}^{\wedge}\left(XX_{rci\times j}^{\wedge}\right) = \frac{f_{O\in\{R,G,B\}}^{\wedge}\left(X_{rci\times j}^{\wedge}\right) - \min\left(f_{O\in\{R,G,B\}}^{\wedge}\left(XX_{rci\times j}^{\wedge}\right)\right)}{\max\left(f_{O\in\{R,G,B\}}^{\wedge}\left(XX_{rci\times j}^{\wedge}\right)\right) - \min\left(f_{O\in\{R,G,B\}}^{\wedge}\left(XX_{rci\times j}^{\wedge}\right)\right)}, \tag{5}$$

where $f^{\wedge}$ is to standardize the image of each channel, $\text{mean}(f_{O\in\{R,G,B\}}^{\wedge}(X_{rci\times j}^{\wedge}))$ is the mean value of image I_{rc} in each channel, and $\text{std}(f_{O\in\{R,G,B\}}^{\wedge}(X_{rci\times j}^{\wedge}))$ is the standard deviation of image I in each channel.

3.2. Features Fusion. In order to obtain more comprehensive facial features, we retain the high-level spatial domain features of neural network to capture more small traces of forgery. $f_s^{\wedge}(I_{rc})$ is defined as the output of the last layer of convolution neural network, and the spatial domain feature extraction results are normalized by formula (5).

In this paper, we design the edge feature vector in the frequency domain and the advanced feature vector in the spatial domain to be merged into an image feature set in a splicing mode. Define $X_{rc}^{(1)}$ to mean the frequency domain edge feature matrix that has undergone tiling vectors, and $X_{rc}^{(2)}$ to mean the finally extracted spatial domain high-level feature matrix, define the ϕ function as matrix splicing, and then define the following fusion formula:

$$\varphi = \phi\left(\left[X_{rc}^{(1)}, X_{rc}^{(2)}\right]\right) = \begin{Bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{Bmatrix}, \tag{6}$$

where ϕ is the final image feature vector. Then, ϕ participates in the classification through formula (7) to obtain a predicted value, which is defined as follows:

$$y^{\wedge} = \text{soft}\max(Z) = \text{soft}\max(W \cdot \varphi + b) = \frac{e_k^z}{\sum_{k=1}^{K} e_k^z}, \tag{7}$$

where W means the neuron weight, b is the bias term, k refers to the classification value, and K is defined as the number of classification categories, and finally, it obtains a prediction result $y^{\wedge}$.

We extract features in the spatial domain based on Meso-4 network and retain the first four layers of neural networks. The first layer uses a $3 * 3$ convolution kernel, and the second, third, and fourth layers use a $5 * 5$ convolution kernel, while ensuring that the first two layers have 8 channels and the last two layers have 16 channels. Excitation function, normalization, and pooling operations are added to each layer to prevent overfitting of the network (great performance on the training set and unable to generalize well to the verification set and test set). The pooling core ensures that the first three layers have $2 * 2$ parameters, and the last layer has $4 * 4$ parameters. Finally, $16 * 8 * 8$ features are obtained for tiling operations, so that the three-dimensional features become $1024 * 1$ vector and are ready for fusion with frequency domain features. As deepfake usually blurs the fake image in the last step of production to achieve a realistic result, the blur extent at the edge of the face and the background are different. Therefore, extraction of edge frequency domain features contributes more to the detection.

We choose the Haar transform, which is sensitive in the horizontal, vertical, and diagonal directions, to extract edge features in the three directions. The features in the three directions are mapped to the same image through formula (4) to obtain a boundary map, and this feature is also stimulated, normalized, and pooled, so it can be connected and matched with spatial domain features. A $2 * 2$ pooling kernel is used in the process, and finally a $3 * 32 * 32$ frequency domain feature is obtained, also presented as a $3072 * 1$ vector feature. Next, we fuse spatial domain and frequency domain features to obtain more comprehensive facial features to improve detection accuracy. The key algorithm of Cross-domain model is described in Algorithm 1. Through vector splicing, a high-level feature vector of 4096 $(64 * 8 * 8)$ is obtained and sent to the fully connected layer for classification. We use a $64 * 1$ feature vector for final classification to ensure the comprehensiveness of features. In the fully connected layer, the Dropout [28] specification is used to improve the robustness of the model, to prevent the occurrence of model over fitting, and to stop the training process in time to achieve better results. The specific parameter details are illustrated in Figure 3.

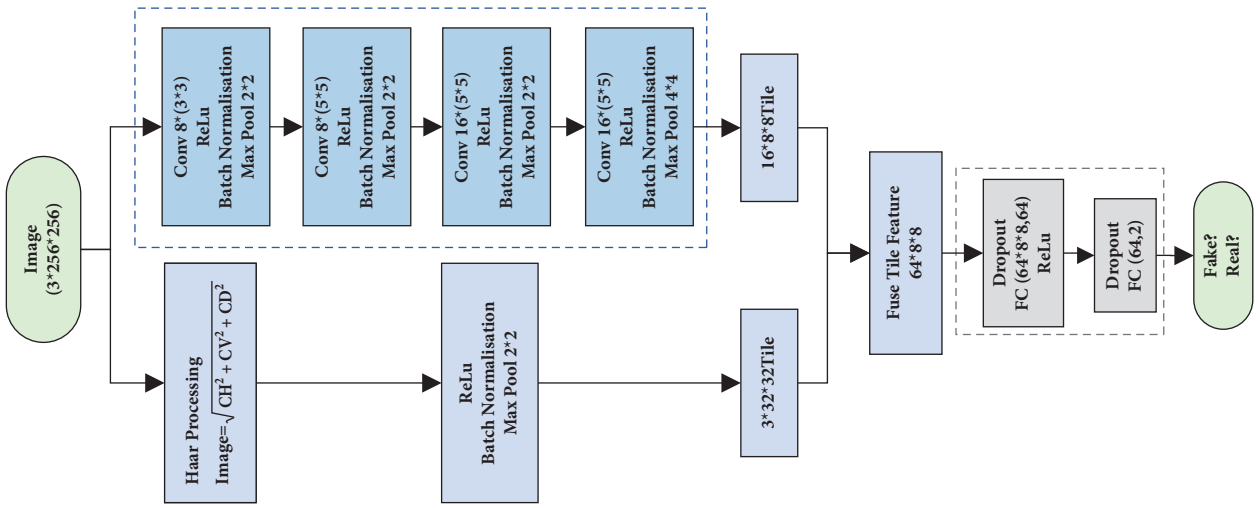

Figure 3: Detailed configuration of cross-domain network structure.

/* Frequency domain feature extraction process */
Input: preprocessed image X_{rc}.
Output: frequency domain feature $X_{rc}^{(1)}$.
(1) Use formula (3) to calculate X_{CH}, X_{CV}, X_{CD}.
(2) X_{CH}, X_{CV}, X_{CD} is mapped into an edge matrix $X_{rc}^{\wedge}$ by formula (4).
(3) Q = operate $X_{rc}^{\wedge}$ on ReLu layer, batch normalization layer, and maximum pooling layer.
(4) $X_{rc}^{(1)}$ = the tile of Q.
/*Obtain high-level semantic spatial domain feature*/
Input: preprocessed image X_{rc}.
Output: high-level semantic spatial domain feature $X_{rc}^{(2)}$.
(5) For $i \leq 4$ do
(6) P = operate $X_{rc}^{\wedge}$ on convolution layer, ReLu layer, batch normalization layer, and maximum pooling layer.
(7) End for
(8) $X_{rc}^{(2)}$ = the tile of P.
/*Fuse features */
Input: frequency domain feature $X_{rc}^{(1)}$ and spatial domain feature $X_{rc}^{(1)}$.
Output: the image facial feature vector ϕ.
(9) Obtain image feature vector by formula (6).
(13) Return

Algorithm 1: The key algorithm of Cross-domain model.

4. Results and Discussion

In this section, we introduce the setup of the experiment, show the processing of the datasets, and discuss the experimental results on the commonly used datasets. In addition, we also compare our method with the two networks proposed by Darius Afchar et al. and with the traditional InceptionV3 network, as well as the FAW method of Li et al.

4.1. Dataset. We use two public datasets to evaluate our method. We optimize our model on the training set and the validation set and obtain the detected AUC value on the test set.

4.1.1. Deepfake Database. The Deepfake Database dataset is proposed by Afchar et al. [18], including training set and validation set. All face images in this dataset are collected from a wide range of Internet sources, including videos that are manually processed to eliminate misalignment and false face detection. In this database, the training set has a total of 12355 images, and the verification set has 7106 images, which always contain 7950 deep forged face images and 11511 real face images. The specific details are illustrated in Table 1. During the experiment, we extracted a part of data from the training set at a ratio of 5 : 1 as the experimental verification set, and the remaining part of training set was used as the experimental training set. The verification set of the original database was used as the experimental test set to ensure the extensiveness of experimental data.

4.1.2. FaceForensics++. The FaceForensics++ dataset was proposed by A. Ro¨ssler and others at the ICCV conference

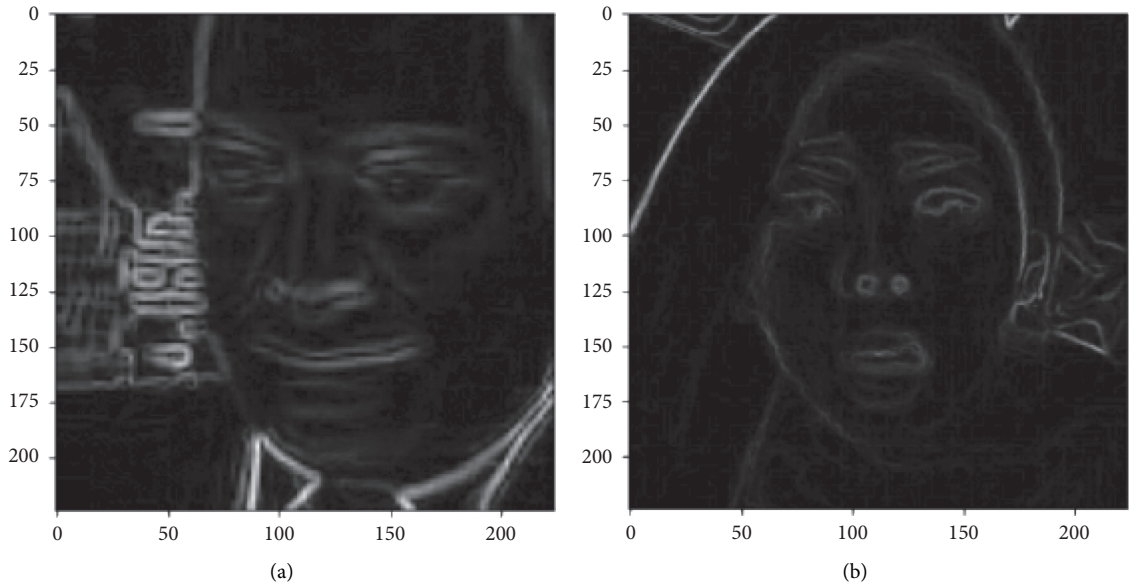

Figure 4: Face image in frequency domain. (a) Real image and (b) fake image (the edge of the face is fuzzy, and the background is clear).

Table 1: Deepfake Database dataset details.

Deepfake database	Train	Val
df	5104	2846
real	7251	4260
total	12355	7106

in 2019 [5]. The dataset contains four different types of forged datasets (high and low resolution). FF++ consists of 1000 original video sequences. The four automatic face fraud methods used are DeepFake, FaceSwap, Face2Face, and Neural Textures. The data comes from 977 YouTube videos, all of which contain a trackable frontal face, enabling automatic tampering methods to generate realistic fake faces. All videos have three resolutions, namely, uncompressed original quality, high-quality using 23 quantized light compression, and low-quality using 40 quantized heavy compression. We conduct experiments on two resolution datasets separately. During the experiment, the ratio of training set and test set is 7 : 3. We use the training set in the dataset for model training and test the model on the validation set in the dataset.

4.2. Experiment Environment. We used the above public datasets for experiments and implemented the tests on a PC with Intel(R) Core(TM) i5-7500 CPU @ 3.40 GHz 3.41 GHz and 4Gram.

On the Deepfake Database dataset, we resize the image to $256 * 256 * 3$. The initial learning rate of 0.1 is divided by 10 every 1000 iterations down to 10−6. We set the weight attenuation to 0.1, the epoch to 20, and the batch size to 64. Finally, the detection result is optimal when the learning rate is 0.001, and the weight attenuation is 0.1.

On the FF++ Database dataset, the initial experimental setup is the same as the Deepfake Database dataset, and the best effect is achieved through fine tuning. The parameters of fine tuning for different datasets are as follows:

(1) For the DeepFake 40 dataset experiment, we set the parameter of weight attenuation 0.1 and set the base learning rate as 0.001 divided by 70 every 5 epochs

(2) For the DeepFake 23 dataset experiment, we set the parameter of weight attenuation 0.3 and set the base learning rate as 0.001 divided by 50 every 5 epochs

(3) For the Face2Face 40 dataset experiment, we set the parameter of weight attenuation 0.03 and set the base learning rate as 0.001 divided by 30 every 5 epochs

(4) For the Face2Face 23 dataset experiment, we set the parameter of weight attenuation 0.1 and set the base learning rate as 0.001 divided by 10 every 5 epochs

(5) For the FaceSwap 40 dataset experiment, we set the parameter of weight attenuation 0.05 and set the base learning rate as 0.001 divided by 50 every 5 epochs

(6) For the FaceSwap 23 dataset experiment, we set the parameter of weight attenuation 0.05 and set the base learning rate as 0.001 divided by 60 every 5 epochs

In order to prove the advantages of the deepfake detection model in this paper, we analyze the detection AUC value of our method on different datasets and show the ROC curve graph. Moreover, we compare and analyze it with other deepfake detection methods.

4.3. Comparison Method. In order to prove the effectiveness of the model in this paper, we implemented a number of comparative experiments on the Deepfake Database dataset, and the experimental results are illustrated in Figure 5(a).

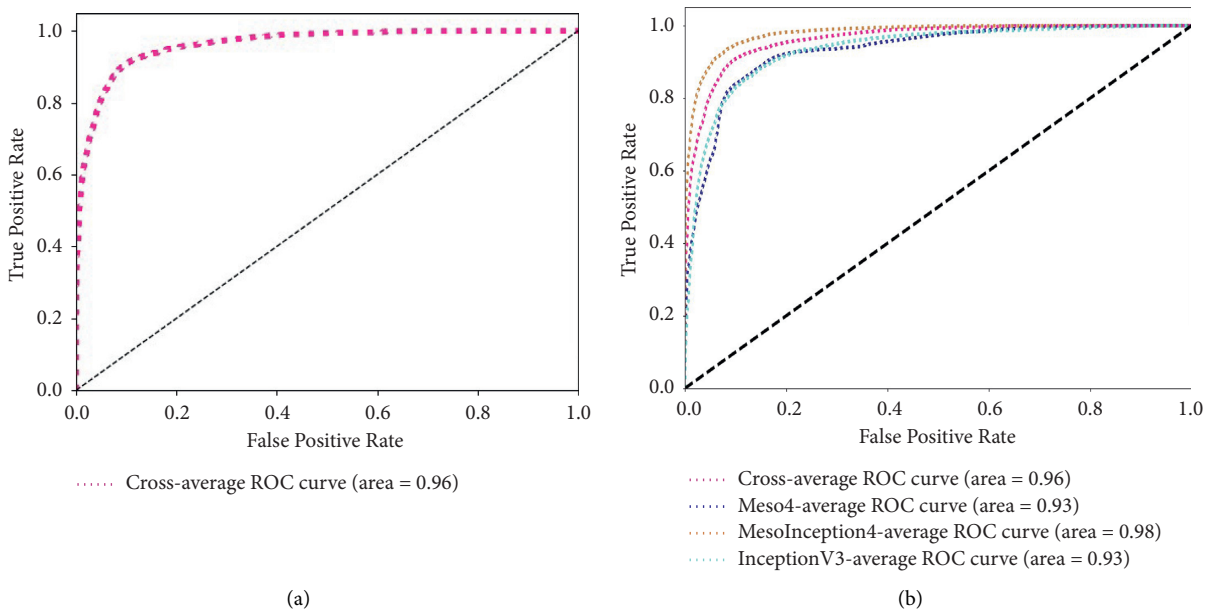

Figure 5: Experiment result of detection on the Deepfake Database dataset. (a) Cross-domain model performance and (b) ROC curves of Meso4, MesoInception4, InceptionV3, and Cross-domain model.

Table 2: The accuracy on the testing set on Deepfake Database dataset.

Method	Deepfake database (%)
Meso4 [15]	87.05
MesoInception4 [15]	92.50
InceptionV3 [12]	88.05
FAW [20]	64.30
Our method	**90.50**

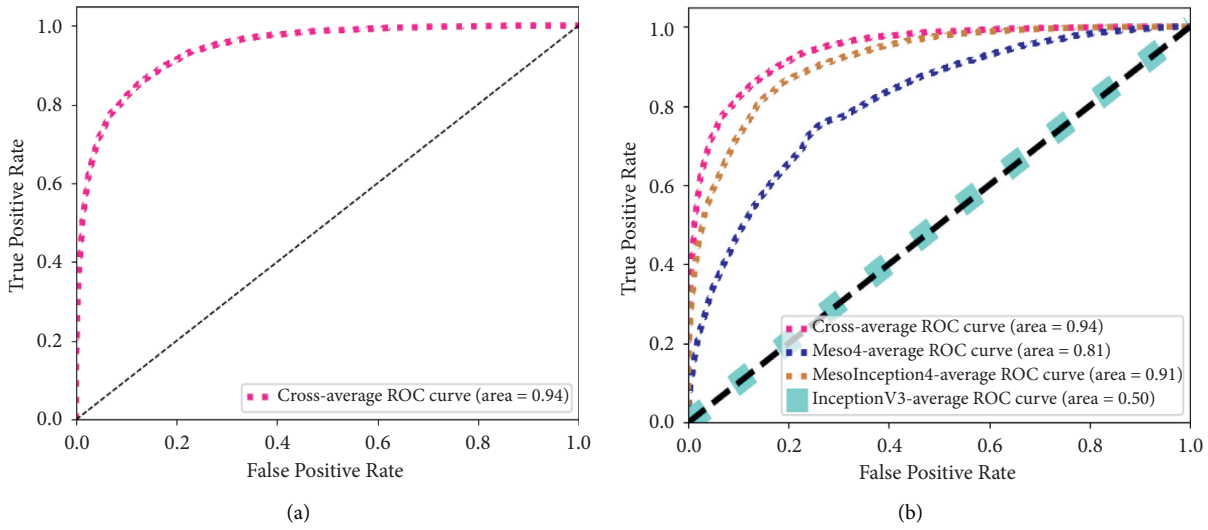

Figure 6: Continued.

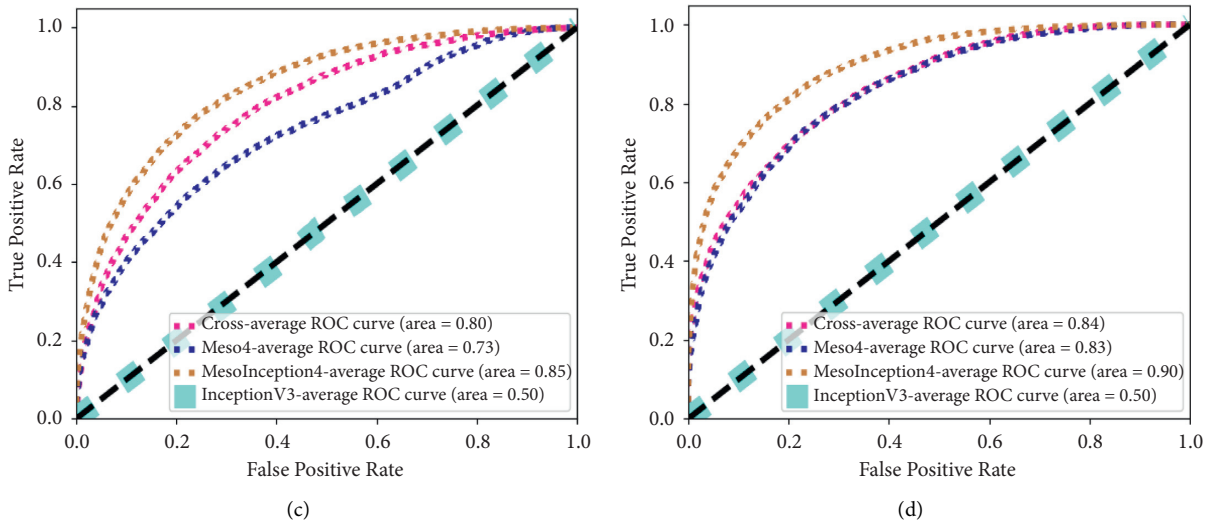

FIGURE 6: Experiment result of detection on the FF++ low-quality dataset. (a) Cross-domain model performance on the DP40 dataset, (b) ROC curves of comparison methods on the DP40, (c) ROC curves of comparison methods on the FS40, and (d) ROC curves of comparison methods on the F2F40.

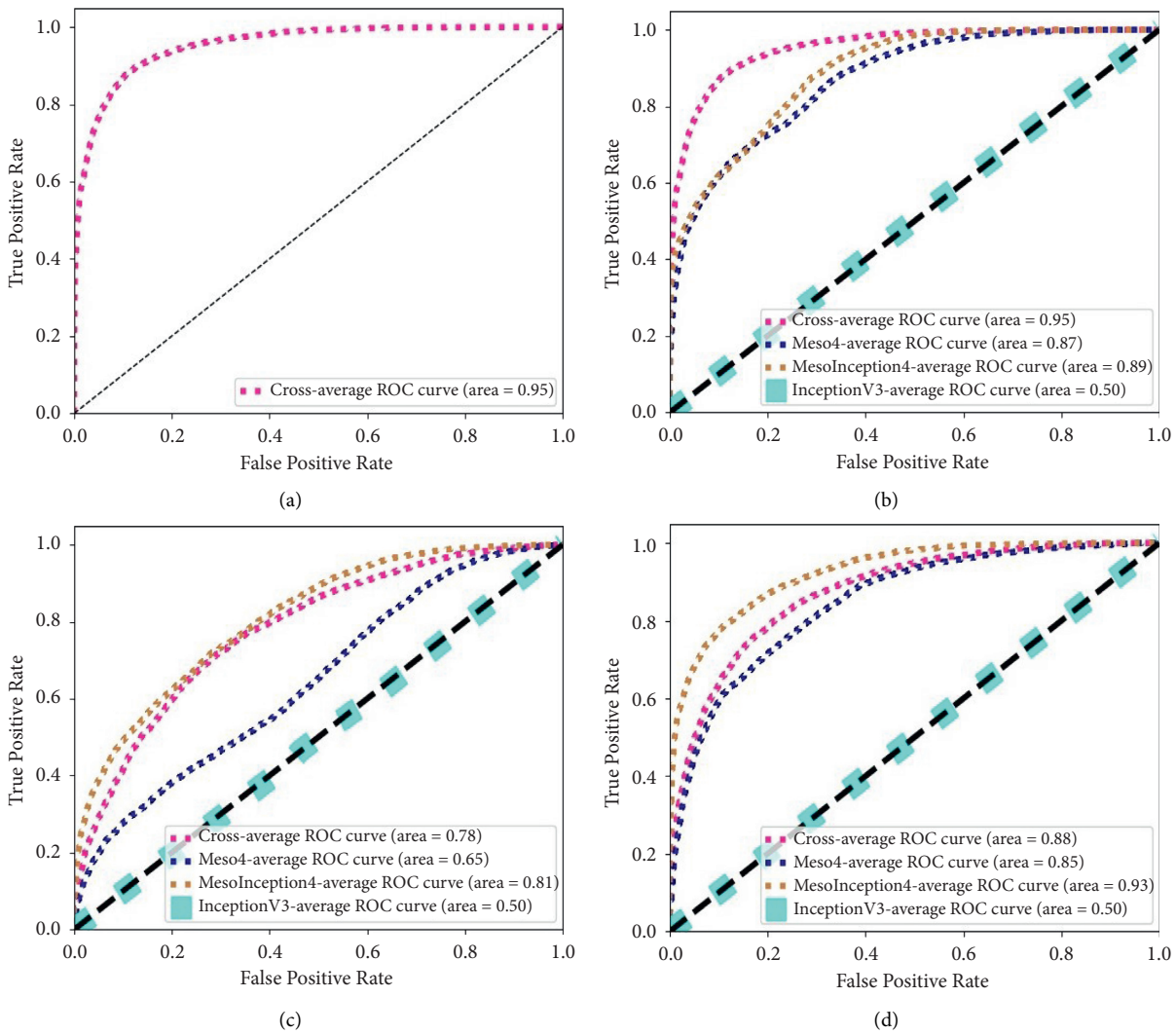

FIGURE 7: Experiment result of detection on the FF++ high-quality dataset. (a) Cross-domain model performance on the DP23 dataset, (b) ROC curves of comparison methods on the DP23, (c) ROC curves of comparison methods on the FS23, and (d) ROC curves of comparison methods on the F2F23.

Table 3: The AUC value of testing on FF++ high-quality dataset with training model on FF++ low-quality dataset.

Method	DP40-model test DP23	F2F40-model test F2F23	FS40-model test FS23
Meso4 [15]	0.87	0.75	0.75
MesoInception4 [15]	0.94	0.90	0.89
InceptionV3 [12]	0.50	0.50	0.50
Our method	**0.97**	**0.84**	**0.83**

The bold is the detection result of our method.

The results indicated that the AUC value of our detection method reached 0.96. At the same time, we also compare our approach with the InceptionV3 network in [15]and the Meso4 and MesoInception4 networks in [18], and the results are illustrated in Figure 5(b). Figure 5(b) signifies that the Meso4 network and the traditional InceptionV3 network achieve an AUC value of 0.93, while the MesoInception4 network achieves an AUC value of 0.98. Our cross-detection model achieves an AUC value of 0.96, right after Meso-Inception4 network. Meanwhile, we use the method presented in [23] to perform the test, and the accuracy results are illustrated in Table 2.

In order to better demonstrate the effectiveness of our method, we test the model in different databases. Detailed parameter settings are introduced in section 4.2. We firstly selected low-quality and high-quality images in FF++ for training and testing. The DP40 data showed that our model reached an AUC value of 0.94, which is relatively a good result as shown in Figure 6(a). Comparison results with other methods are illustrated in Figure 6(b). The experimental results for the FS40 and F2F40 datasets are illustrated in Figures 6(c) and 6(d). Our model reached an AUC value of 0.80 on the FS40 dataset and had an AUC value of 0.84 on the F2F40 dataset. It can be seen from the AUC value in the figures that the detection method in this paper has a better performance. Figure 7(a) indicates that our model has reached an AUC of 0.95 in the DP23 dataset, which is an increase of 0.01 better than the result in the DP40 dataset. Also compared with the results of the other three detection methods, our result is the best, illustrated in Figure 7(b). Our method reached an AUC value of 0.78 on the FS23 dataset, and 0.88 on the F2F23 dataset, illustrated in Figures 7(c) and 7(d). For all of the above, it is shown that our method has better performance no matter in high quality or low quality. Our method has the best on deepfake dataset, and second on other datasets. This is because our method on DeepFake dataset has fully taken face edge features into account, and the forgery traces of this dataset are clearly exposed on the edge features. However, on other forgery methods datasets, it has less consideration on subtle features such as lips, eye, and nose. This is the direction to be diligent in the future.

In addition, we use models trained on low-quality datasets to test high-quality datasets in order to verify the cross-database performance. Experimental results are illustrated in Table 3. The performance of our method will not degrade with the change of datasets. On the contrary, our method achieved an AUC of 0.97 on DP23 dataset and had an AUC value of 0.84 and 0.83 on F2F23 dataset and FS23 dataset, respectively.

5. Conclusion

In this paper, we propose a new deepfake detection method based on cross-domain fusion, where edge geometric features in frequency domain are introduced to image feature set and fused with high-level semantic features in spatial domain for better detection accuracy. Experimental analysis shows that our detection method is effective and has better cross-database performance when the target datasets are forged using deepfake technology. This is due to the fact that our method has fully taken face edge features into account. However, our method is less effective on Neural Textures dataset as it has less consideration on subtle features such as lips, eye, and nose. In the future, we will further investigate the extraction of subtle facial features and continue to enhance detection robustness and cross-dataset performance.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the industry university cooperation collaborative education project of Higher Education Department of the Ministry of Education of China (project no. 201801002012). This work was also supported by the National Natural Science Foundation of China (nos. U1936117 and 62076052), the Science and Technology Innovation Foundation of Dalian (no. 2021JJ12GX018), and the Fundamental Research Funds for the Central Universities (DUT21GF303, DUT20TD110, and DUT20RC(3)088).

References

[1] X. Liu, X. Zhai, W. Lu, and C. Wu, "QoS-guarantee resource allocation for multibeam satellite industrial Internet of Things with NOMA," *IEEE Transactions on Industrial Informatics*, no. 99, p. 1, 2019.

[2] X. Liu and X. Zhang, "Rate and energy efficiency improvements for 5G-based IoT with simultaneous transfer," *IEEE Internet of Things Journal*, vol. 6, no. 99, p. 1, 2018.

[3] X. Liu, X. Zhang, W. Lu, and M. Xiaong, "Energy efficiency maximization for green cognitive internet of things with energy harvesting," *Machine Learning and Intelligent Communications*, Springer, New York, Ny, USA, 2019.

[4] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: a survey," 2020, https://arxiv.org/abs/2004.11138.

[5] A. Ro¨ssler, D. Cozzolino, L. Verdoliva, C. Riess, and J. Thies, "M. Nie?ner, Faceforensics++: learning to detect manipulated facial images," in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, Seoul, Korea, November 2019.

[6] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: a survey of face manipulation and fake detection," *Information Fusion*, vol. 64, pp. 131–148, 2020.

[7] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Two-stream neural networks for tampered face detection," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1831–1839, IEEE, Honolulu, HI, USA, July 2017.

[8] H. Li, B. Li, S. Tan, and J. Huang, "Detection of deep network generated images using disparities in color components," 2018, https://arxiv.org/abs/1808.07276.

[9] D. Cozzolino and L. Verdoliva, "Noiseprint: a CNN-based camera model fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 144–159, 2019.

[10] Y. Hu, Y. Gao, B. Liu, and G. Liao, "Deepfake videos detection based on image segmentation with deep neural networks," *Journal of Electronics and Information Technology*, vol. 43, no. 1, pp. 162–170, 2021.

[11] S. McCloskey and M. Albright, "Detecting GAN-generated imagery using color cues," 2018, https://arxiv.org/abs/1812.08247.

[12] L. Nataraj, T. Mohammed, S. Chandrasekaran, A. Flenner, J. H. Bappy, and B. S. Anjunath, "Detecting GAN generated fake images using co-occurrence matrices," 2019, https://arxiv.org/abs/1903.06836.

[13] X. Zhang, S. Karaman, and S.-F. Chang, "Detecting and simulating artifacts in gan fake images," 2019, https://arxiv.org/pdf/1907.06515.

[14] M. A. Younus and T. M. Hasan, "Effective and fast DeepFake detection method based on haar wavelet transform," in *Proceedings of the 2020 International Conference on Computer Science and Software Engineering (CSASE)*, Jeju, Korea, December 2020.

[15] C. Szegedy, V. Vincent, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Las Vegas, NV, USA, July 2016.

[16] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the 2015 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 1–9, Boston, MA, USA, July 2015.

[17] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2016, https://arxiv.org/abs/1511.07122.

[18] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: a compact facial video forgery detection network," 2018, https://arxiv.org/abs/1809.00888.

[19] R. Wang, F. Juefei-Xu, L. Ma et al., "Fakespotter: a simple yet robust baseline for spotting AI-synthesized fake faces," 2019, https://arxiv.org/abs/1909.06122.

[20] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: using capsule networks to detect forged images and videos," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Brighton, United Kingdom, May 2019.

[21] S. Ekraam, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 80–87, IEEE, Long Beach, CA, USA, June 2019.

[22] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *Proceedings of the 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pp. 83–92, IEEE, Waikoloa Village, HI, USA, January, 2019.

[23] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," 2019, https://arxiv.org/abs/1811.00656.

[24] J. Stehouwer, H. Dang, F. Liu, X. Liu, and A. Jain, "On the detection of digital face manipulation," 2020, https://arxiv.org/abs/1910.01717.

[25] S. A. Khan, A. Artusi, and H. Dai, "Adversarially robust deepfake media detection using fused convolutional neural network predictions," 2021, https://arxiv.org/abs/2102.05950.

[26] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2015, https://arxiv.org/abs/1412.6980.

[27] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," 2015, https://arxiv.org/abs/1502.03167.

[28] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfifitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.