Computers, Materials & Continua
DOI:10.32604/cmc.2021.016760
Article

Tech Science Press

DeepFake Videos Detection Based on Texture Features

Bozhi Xu[1], Jiarui Liu[1], Jifan Liang[1], Wei Lu[1,*] and Yue Zhang[2]

[1]School of Computer Science and Engineering, Guangdong Province Key Laboratory of Information Security Technology, Ministry of Education Key Laboratory of Machine Intelligence and Advanced Computing, Sun Yat-sen University, Guangzhou, 510006, China
[2]Department of Computer Science, University of Massachusetts Lowell, Lowell, 01854, MA, USA
[*]Corresponding Author: Wei Lu. Email: luwei3@mail.sysu.edu.cn
Received: 09 January 2021; Accepted: 11 February 2021

Abstract: In recent years, with the rapid development of deep learning technologies, some neural network models have been applied to generate fake media. DeepFakes, a deep learning based forgery technology, can tamper with the face easily and generate fake videos that are difficult to be distinguished by human eyes. The spread of face manipulation videos is very easy to bring fake information. Therefore, it is important to develop effective detection methods to verify the authenticity of the videos. Due to that it is still challenging for current forgery technologies to generate all facial details and the blending operations are used in the forgery process, the texture details of the fake face are insufficient. Therefore, in this paper, a new method is proposed to detect DeepFake videos. Firstly, the texture features are constructed, which are based on the gradient domain, standard deviation, gray level co-occurrence matrix and wavelet transform of the face region. Then, the features are processed by the feature selection method to form a discriminant feature vector, which is finally employed to SVM for classification at the frame level. The experimental results on the mainstream DeepFake datasets demonstrate that the proposed method can achieve ideal performance, proving the effectiveness of the proposed method for DeepFake videos detection.

Keywords: DeepFake; video tampering; tampering detection; texture feature

1 Introduction

In recent years, with the rapid development and widespread application of deep learning [1,2], face manipulation technologies have made great progress. As a representative video manipulation technology, DeepFake technology generates fake facial videos based on deep neural networks such as auto-encoder and Generative Adversarial Networks (GAN). It is easy to replace the target face with the fake face to maliciously tamper the video contents using face manipulation methods [3,4]. Tampering of faces violates personal portrait rights and may cause social disputes. In addition, different from images, a larger amount of information can be spread through videos, which brings more fake information after forgery. Therefore, it is of great important to develop detection methods for DeepFake videos.

In the past two decades, many digital image forensics methods have been developed [5–7]. At present, the forensics of face manipulation videos has been attracted a lot of research interest, and a lot of methods are proposed for DeepFake videos detection. Li et al. [8] review the existing generation technologies of DeepFake videos and several independent open-source implementations. They also introduce the generation process based on deep learning as well as the corresponding detection technologies. In addition, the mainstream DeepFake videos datasets are also introduced. At present, Deep learning is widely used to detect DeepFake videos, which constructs deep neural networks to detect the frame sequence after framing. Based on the observation that some inconsistent choices, such as illuminants, exist between scenes with fake frames, a detection method is proposed in [9] using Recurrent Neural Network (RNN). Firstly, the video is divided into frames and Convolutional Neural Network (CNN) is used to extract features of the face. Then the features are sent into Long Short Term Memory (LSTM) network to detect the relationship of time-series between frames for discrimination. Based on the detection of eye blinking in the videos, a set of features based on eye areas are extracted by CNN which is fed into a LSTM network to identify the fake videos [10]. But when people's eyes are closed, this method can not detect this situation well. In addition, when enough eye blinking images are added deliberately to the training set, the method may lead to misjudgment. Li et al. [11] use two classical deep neural networks, VGG and Residual Network (ResNet), to capture the artifacts caused by the affine transformation, which can efficiently detect fake videos. Different from using existing neural networks, to realize the detection of fake videos, a new network structure, the Meso-4 network, is constructed in [12]. At the same time, the inception module is used to construct the MesoInception-4 network to extract features at the mesoscopic level. In order to simultaneously solve the problems of tampered videos detection and tampered area location, a multi-task learning method that designs a convolutional neural network is proposed in [13]. Different from the method of using deep learning to detect DeepFake videos, many methods [14–16] use handcraft features for classification. Yang et al. [14] estimate 3D head pose based on the inconsistency of the head posture of fake videos and use Support Vector Machine (SVM) for classification. However, the method cannot achieve excellent performance. Agarwal et al. [15] also track facial and head movements, but facial action units are extracted as features to detect DeepFake videos. Jung et al. [16] also analyze the transformation in the pattern of human eye blinking. Based on the period, repeated number and elapsed eye blink time, the features are extracted to determine whether the video is real or fake. In addition, the idea of combining hand-crafted features with deep learning methods is adopted [17,18]. In [17], the deep learning method uses the GoogLeNet network, while some features of steganalysis are extracted as hand-crafted features and sent to the SVM. Finally, the classification probability obtained by CNN and SVM is used to combine as a score for judgment. In [18], based on that some simple visual artifacts, such as the color of the left and right eye, are existed in fake videos, relative features are extracted, then a multilayer feedforward neural network and a logistic regression model are used as the classifier.

In order to improve the interpretability of the model and solve the problem that the training samples may be insufficient, based on the observations that some DeepFake videos are lack of facial texture details, a new method using traditional machine learning technologies is proposed to detect DeepFake videos. Firstly, texture features are extracted using the image gradient, standard deviation, gray-level co-occurrence matrix, and wavelet transform from the face region of every frame, which can represent facial texture details. Secondly, based on the texture features, SVM is employed to realize the detection of DeepFake videos.

The remaining parts of this paper are organized as follows. Section 2 mainly introduces the generation technologies of DeepFake videos and the corresponding defects of fake videos. In Section 3, the proposed method for DeepFake videos detection is discussed, including the extraction method of the texture feature and the feature selection method. Section 4 presents the experimental results and analysis, and finally the conclusion is given in Section 5.

2 DeepFake Videos Generation Technology

The generation technologies of DeepFake videos use neural network to tamper and replace the face in each frame of the video, and then recompress to generate a fake video. A generation technology of DeepFake videos based on the auto-encoder is introduced in [8,12]. The model consists of an encoder network and a decoder network. The encoder network takes a facial frame as input to capture the facial features and convert it into a vector as output. The decoder network reconstructs the vector as a fake face, which is finally fused into the background to construct a fake frame. In the training phase, two sets of different face frames are used to train two pairs of encoder networks and decoder networks, in which two encoder networks share the weights, and two decoder networks are trained separately. After the parallel training is completed, when one person's frame is taken into the auto-encoding model of another person, the encoder captures the facial structure, lighting and other similar features of the faces. And the decoder reconstructs the face details and some unique attributes to generate a fake frame. After performing the same processing on each frame, a fake video is generated.

At present, some generation model of DeepFake videos cannot generate all the textures of the face, causing that some fake faces are relatively rough. For example, the subtle wrinkle of the face cannot be generated perfectly. At the same time, in the final process of generating the fake frame, the generated face is fused into the background. In order to reduce the boundary inconsistency caused by this process, some smoothing operations are usually used [19], which will also cause the loss of facial texture details. For example, Fig. 1a is the frame selected from the real videos dataset VidTIMIT, and Fig. 1b is the frame selected from the face manipulation videos dataset DeepFake-TIMIT [19]. According to the figure, it is difficult for human eyes to distinguish which frame is real. However, the real frame has more detailed texture, such as the double eyelids and wrinkle details around the eyes, while texture details of the fake frame are lacking.

(a) (b)

Figure 1: Example of real and fake frames. (a) is a real frame, (b) is a fake frame

Some current works have focused on the facial texture to distinguish real videos and fake videos [20–23]. Aiming at the insufficient texture details of the DeepFake videos mentioned above,

the relevant features are extracted to capture the characteristics of the texture from the face region of the frame, and then these features are used to identify the authenticity of the frame.

3 DeepFake Videos Detection Method Based on Texture Features

The proposed method for the detection of facial manipulation videos based on texture features is described in this section. The extraction method of the texture features is described in Section 3.1. In Section 3.2, the feature selection method is presented. Finally, the overall pipeline of the proposed method is given in Section 3.3.

3.1 Texture Feature Extraction Method

Texture features are complex visual features that can characterize the roughness and regularity of images. The analysis of image texture is a classical research direction in the fields of image processing and computer vision. A series of theories are constructed for texture feature extraction, which are used to analyze the texture of image. Great development has been achieved in this field [24–28]. Tuceryan et al. [24] divide texture feature extraction methods into four categories: statistical methods, geometrical methods, model methods and signal processing methods. The statistical methods extract the statistical characteristics of the pixel value and its neighborhood as texture features. The geometrical methods analyze image textures by the geometric properties of "texture elements" or primitives. The model methods are implemented by constructing models such as random field models. The signal processing methods extract texture features from the transform domain. In addition, a series of classical texture feature extraction algorithms have been proposed, such as gray-level co-occurrence matrix [25], local binary pattern [26], Markov random field model [27], wavelet transform method [28] and so on.

Based on the defect of insufficient texture details in the DeepFake videos described in the Section 2, four texture feature extraction methods are used based on the gradient domain, standard deviation, gray level co-occurrence matrix, and wavelet transform respectively to extract the corresponding texture features of the face region, which can effectively classify real videos and fake videos.

3.1.1 Texture Feature Based on the Gradient Domain and Standard Deviation

The image gradient characterizes the changes of the gray scale of each pixel in their neighborhood, which can represent the texture level of the image. In the areas with sufficient texture details such as the edge of the image, the gray level changes greatly and the gradient value is large. While the gray level changes smaller and the gradient value is small where the areas are smooth. Images with different texture details have different statistical characteristics of gradients. Therefore, in this paper, the statistical features of the gradient map are used as texture features.

Usually, the difference is used to obtain the vertical and horizontal gradient of the image. Combining the gradient information in the horizontal and vertical directions of the image, the equation for calculating the gradient amplitude M is shown as follows:

$$M = gx^2 + gy^2 \tag{1}$$

where gx and gy represent the gradient of the image in the horizontal and vertical directions, respectively.

Based on the gradient amplitude, the mean, variance, skewness and kurtosis of it are extracted as texture features, which can reflect the statistical characteristics of the data distribution.

At the same time, the standard deviation is calculated for the gray image. The standard deviation reflects the dispersion between the image pixels and the overall level of the image. The larger the standard deviation is, the greater each pixel value changes and the sufficient image texture details are. Therefore, the standard deviation of the grayscale image is also calculated as feature to characterize the texture of the image.

3.1.2 Texture Feature Based on Gray Level Co-Occurrence Matrix

The gray level co-occurrence matrix describes the texture through statistical analysis of the spatial distribution of each pixel in the image. Given the direction and distance, the probability that two adjacent gray level pixels appear in the image with a specific spatial distribution can be calculated. The probability calculated from different gray levels constitutes a gray level co-occurrence matrix. From the gray level co-occurrence matrix, 14 texture features can be calculated [25]. In this paper, five texture features are used, including the contrast, energy, homogeneity, entropy, and correlation. These five features are introduced as follows.

Contrast reflects the richness of the texture details and the depth of the textures. The more pixels that their gray-scale difference is large, the greater the contrast value is [29]. The equation for calculating contrast is shown as follows:

$$\mathrm{f}_{Con} = \sum_{i,j=1}^{N} P_{i,j}\,(i-j)^2 \tag{2}$$

Energy is also called angular second moment, which reflects the uniformity of the gray level distribution of the image [29]. The equation for calculating energy is shown as follows:

$$\mathrm{f}_{\mathrm{Asm}} = \sum_{i,j=1}^{N} \mathrm{P}_{i,j}^2 \tag{3}$$

The homogeneity reflects the intensity of local texture changes. The value of homogeneity is larger where the local texture changes more uniformly. The equation for calculating homogeneity is shown as follows:

$$\mathrm{f}_{\mathrm{Hom}} = \sum_{i,j=1}^{N} \frac{P_{i,j}}{1+|i-j|} \tag{4}$$

Entropy measures the amount of information of the local area. If the image has more texture information, the probability values of the gray-level co-occurrence matrix are uniformly distributed, and the entropy value is large [29]. The equation for calculating entropy is shown as follows:

$$\mathrm{f}_{Ent} = \sum_{i,j=1}^{N} -P_{i,j} \log P_{i,j} \tag{5}$$

Correlation measures the degree of correlation between the elements of the gray level co-occurrence matrix [29]. The equation of calculating correlation is shown as follows:

$$f_{Cor} = \sum_{i,j=1}^{N} \frac{(i-\mu_i)\left(j-\sigma_j\right)P_{i,j}}{\sigma_i \sigma_j} \tag{6}$$

where

$$\mu_i = \sum_{i,j=1}^{N} iP_{i,j} \tag{7}$$

$$\mu_j = \sum_{i,j=1}^{N} jP_{i,j} \tag{8}$$

$$\sigma_i = \sqrt{\sum_{i,j=1}^{N} P_{i,j}(i-\mu_i)^2} \tag{9}$$

$$\sigma_j = \sqrt{\sum_{i,j=1}^{N} P_{i,j}\left(j-\mu_j\right)^2} \tag{10}$$

N is the size of the gray-level co-occurrence matrix, and $P_{i,j}$ is the value of the i-th row and j-th column of the gray-level co-occurrence matrix.

In order to measure the gray level changes in various directions and extract the texture details in each direction as much as possible, we calculate the gray level co-occurrence matrix with distance of 1 in 0°, 45°, 90°, 135° directions. Considering the amount of calculation and the fineness of texture details reflected by the gray-level co-occurrence matrix, the gray level of the gray-level co-occurrence matrix is set to 64. The contrast, correlation, energy, homogeneity and entropy are calculated for the four gray-level co-occurrence matrices. Finally, the features calculated by the four gray-level co-occurrence matrixes are averaged respectively as the extracted texture features.

3.1.3 Texture Feature Based on Wavelet Transform

Wavelet transform is widely used in many fields such as image processing and signal processing [28,30]. Using wavelet transform to decompose the image in both horizontal and vertical directions, low-frequency sub-band, horizontal high-frequency sub-band, vertical high-frequency sub-band and diagonal high frequency sub-band can be given. The high-frequency sub-bands contain most of the texture information of the image. Statistical analysis of these sub-bands can obtain the texture level of the image, which can classify images with different texture richness.

Therefore, wavelet decomposition is performed on the image to obtain three coefficient matrices based on the horizontal high-frequency, vertical high-frequency, and diagonal high-frequency.

The average value, standard deviation and energy of the three coefficient matrices are calculated as texture features. The equation for calculating energy is shown as follows:

$$f_{En} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} x_{i,j}^2 \tag{11}$$

where M and N are the size of the coefficient matrix, and $x_{i,j}$ is the coefficient in the i-th row and j-th column of the coefficient matrix.

3.2 Feature Selection Method

The texture features introduced in Section 3.1 can represent which frame contains more texture details. However, some features are not discriminative enough to distinguish whether the frame is real or fake. On the one hand, the extracted features are used to describe the frame texture, which may cause feature redundancy. On the other hand, the features are extracted based on the block of face, some blocks may be not facial areas. Thus the features extracted from these blocks may be invalid. Therefore, a feature selection method is introduced to perform feature screening for improving the classification performance.

The feature selection method is shown in Algorithm 1. Firstly, the first half of the texture features are selected to initialize feature subset. Then the remaining features are taken into the feature subset one by one. If the performance J can be improved, leave the feature. Otherwise discard it. After the second half features are all selected, the first half features are discarded and then put back into the feature subset one by one to decide if the feature can be left. After all features are selected, the feature subset is the final features set used to distinguish whether the frame is real or fake.

3.3 The Overall Pipeline

The flowchart of the DeepFake videos detection method we proposed is shown in Fig. 2. The steps of the proposed method are described as follows:

(1) The proposed method refines DeepFake videos detection to frame level, so the videos are firstly decoded into frames. To evaluate the texture details of the face area in every frame, the feature points of the face are extracted using DLIB [31], and the face area is located and cropped according to the extracted feature points.

(2) As a classical method, Wiener filter is widely used to remove noise. Many methods are proposed to denoise image based on Wiener filter [32,33]. In order to reduce the influence of noise caused by sensors on the texture features while preserve the texture details of the frame as much as possible, Wiener filtering is used to denoise the face area.

(3) Because the tampered area is unknown, the cropped face area is only a rough area, which may include part of the untampered area. In order to reduce the impact of inaccurate interception on the tampered area, at the same time, to avoid only part of the face area being tampered with, such as only the mouth area, we divide the cropped face area into 9 blocks on average and extract texture features for each sub-block to ensure that the forged area is included in the sub-block, so that the extracted texture features can effectively characterize the richness of texture details in the face area.

(4) Then the extraction method of texture feature introduced in Section 3.1 is used. Through calculating the mean, standard deviation, skewness, and kurtosis of the image gradient, calculating

the standard deviation of the grayscale image, calculating the contrast, inverse moment, correlation, energy, entropy of the gray level co-occurrence matrix, and calculating the mean, energy, and standard deviation of the horizontal, vertical, and diagonal high-frequency approximation matrices obtained by wavelet transform for each face region, the texture features are extracted and composed a 171-dimensional features vector.

(5) Taking the texture feature vector as input, then we use feature selection method introduced in Section 3.2 to remove some redundant features, the retained features are used as the final discriminant features.

(6) After normalization of the extracted texture features, the SVM classifier is used to train and classify each frame of videos.

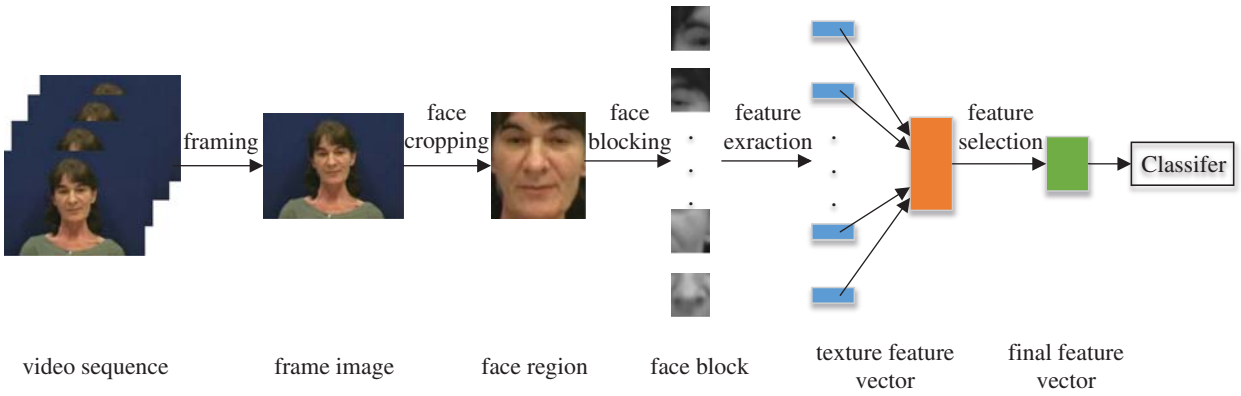

Figure 2: Flow diagram of DeepFake videos detection method based on texture features

4 Experiment

4.1 Dataset and Implementation Details

The datasets used for experiment are DeepFake-TIMIT dataset [19], FaceForensics++ dataset [34], Celeb-df dataset [8] and DeepFake Detection Challenge (DFDC) Preview dataset [35], which are the mainstream datasets for detecting face manipulation videos.

The DeepFake-TIMIT dataset [19], which is based on the VidTIMIT dataset, is generated by the face-swapping algorithm based on GAN. The dataset is divided into two types of videos: one is low quality videos (TIMIT-LQ), whose resolution of the fake face is 64. The other is high quality videos (TIMIT-HQ), whose resolution of the fake face is 128. Each type of video contains 32 different characters, and each character has about 10 videos in which actors speak to the camera, and each video lasts at least 4 seconds. To construct the TIMIT dataset for experiment, the fake videos are selected from the DeepFake-TIMIT dataset, and the real videos are selected from the corresponding VidTIMIT dataset.

The FaceForensics++ dataset [34] is a large scale face manipulation dataset. The real videos of this dataset are collected from the Internet, and most of the videos are downloaded from the YouTube. The real videos consist of 1000 videos, which contain 509914 frames in total. Four face manipulation technologies are used to generate face tampered videos, namely FaceSwap, DeepFake, Face2Face and NeuralTextures. Because the proposed method is to detect DeepFake videos, so only fake videos generated by the DeepFake method in the FaceForensics++ dataset

(FF-DF) are used as fake videos sets. There are three quality videos in the dataset, namely raw videos (C0), light compression videos (C23) and low quality videos (C40). Each type of tampered video contains 1,000 fake videos.

The Celeb-df dataset [8] is a large scale DeepFake videos dataset, which contains 590 real videos and 5639 DeepFake videos. The number of frames is over two million. The real videos are collected from YouTube. The DeepFake videos are generated using an improved DeepFake synthesis algorithm to solve some problems, such as the low resolution of synthesized faces.

The DFDC Preview dataset [35] is a preview of the DFDC dataset, which contains around 5000 videos. The real videos are shot by many actors, which include varied lighting conditions, head poses and visual diversity backgrounds. Two methods are used to generate fake videos, which produces different qualities swaps.

All videos of the four datasets are firstly framed. The number of frames of Deepfake-TIMIT dataset [19], FaceForensics++ dataset [34], Celeb-DF dataset [8] and DFDC Preview dataset [35] are about 70000, 500000, 2000000 and 1000000 respectively. Then the face areas are located and cropped using DLIB [31].

Accuracy (Acc) and the area under the receiver operating characteristic curve (AUC) are used in the experiment for evaluation. We performed detection at the frame level, that is, at the image level. The higher the value of ACC and AUC, the better the performance of the method.

4.2 Experimental Results and Analysis

4.2.1 Impact of Feature Selection Method

To verify the effectiveness of the feature selection method, a comparative experiment that the feature selection method is used or not is taken on different quality videos of DeepFake-TIMIT [19] dataset. The performance is shown in Tab. 1.

Table 1: Accuracy (%) of use feature selection method or not on Deefake-TIMIT dataset

Methods	TIMIT-LQ Acc (%)	TIMIT-HQ Acc (%)
Without feature selection method	92.5	86.1
With feature selection method	**92.6**	**94.4**

From the experimental results, it is obvious that after using the feature selection method, the detection accuracy of low-quality videos has increased by 0.1% approximately, and the accuracy of high-quality videos has increased by 8.3% approximately. The feature selection method can help to improve the detection accuracy, proving the effectiveness of the feature selection method.

4.2.2 Performance on Mainstream DeepFake Datasets

The proposed method is evaluated on four mainstream DeepFake datasets, including DeepFake-TIMIT dataset [19], FaceForensics++ dataset [34], Celeb-df dataset [8] and DFDC Preview dataset [35]. The performance of the proposed method is shown in Tab. 2.

From the experimental results, it can be seen that the proposed method can achieve ideal performance on DeepFake-TIMIT dataset [19] and FaceForensics++ dataset [34]. The accuracy and AUC score of the proposed method on both two datasets are higher than 85% and 94%, respectively. On Celeb-df dataset [8] and DFDC Preview dataset [35], the accuracy and AUC score

of the proposed method are higher than 75% and 79%. Because these two datasets are generated by improved DeepFake synthesis algorithm, the situation that lacking of facial texture details has been improved. Therefore, compared with DeepFake-TIMIT dataset [19] and FaceForensics++ dataset [34], the performance of the proposed method is degraded on the other two datasets. In general, the proposed method can achieve ideal performance, but it is still a challenge for the proposed method to detect on Celeb-df dataset [8] and DFDC Preview dataset [35].

Table 2: Performance of the proposed method on DeepFake-TIMIT, FaceForensics++, Celeb-DF and DFDC preview dataset

Dataset	Acc (%)	AUC (%)
DeepFake-TIMIT [19]	94.4	98.2
FF-DF [34]	87.3	94.3
Celeb-df [8]	75.7	82.3
DFDC Preview [35]	77.7	79.5

4.2.3 Cross-Data Evaluation

In order to evaluate the performance of the proposed method on different quality videos and cross-quality videos, we use three quality videos of FaceForensics++ dataset [34] as the training set and testing set respectively. For example, the C0 quality videos are used to train, then C0, C23 and C40 quality videos are used to test. The experimental results are shown on Tab. 3.

Table 3: Accuracy (%) of training and testing on three different quality videos of FaceForensics++ dataset

The videos' quality of training set	The videos' quality of testing set	Acc (%)
C0	C0	87.3
	C23	86.2
	C40	85.3
C23	C0	86.7
	C23	86.3
	C40	86.6
C40	C0	79.8
	C23	80.2
	C40	91.2

From the experimental results, it can be seen that the accuracy of the proposed method on c0, c23 and c40 quality videos are 87.3%, 86.3% and 91.2%, respectively. When training on one quality videos and testing on other quality videos, the accuracy is basically the same, especially training on c0 and c23 quality videos. In general, the experimental results show that the proposed method can achieve ideal performance on different quality videos. The proposed method is robust to videos with different compression rates.

4.2.4 Comparison with Other Methods

Four mainstream DeepFake videos detection algorithms, including FWA [11], MesoNet [12] and XceptionNet [34], texture method [23] are used for comparison. ResNet is used for detection in FWA [11], MesoNet is the neural network proposed in [12] and XceptionNet is the baseline network used by the authors who build the FaceForensics++ dataset [34]. Two texture features, LBP and HOG are used in [23]. All methods are detected at the frame level. The performance compared with deep learning methods on the DeepFake-TIMIT dataset [19] and FaceForensics++ dataset [34] is shown in Tabs. 4 and 5. The performance compared with texture method [23] on the FaceForensics++ dataset [34] is shown in Tab. 6.

Table 4: Accuracy (%) of the proposed method and deep learning methods on DeeFake-TIMIT and FaceForensics++ dataset

Methods	TIMIT-LQ Acc (%)	TIMIT-HQ Acc (%)	FF-DF Acc (%)
FWA [11]	87.6	71.4	70.9
MesoNet [12]	76.1	77.5	86.5
XceptionNet [34]	88.2	93.6	**99.2**
Proposed	**92.6**	**94.4**	87.3

Table 5: AUC (%) of the proposed method and deep learning methods on DeeFake-TIMIT and FaceForensics++ dataset

Methods	TIMIT-LQ AUC (%)	TIMIT-HQ AUC (%)	FF-DF AUC (%)
FWA [11]	**99.9**	93.2	80.1
MesoNet [12]	87.8	84.3	84.7
XceptionNet [34]	97.5	94.1	**99.7**
Proposed	99.5	**98.2**	94.3

Table 6: Accuracy (%) of the proposed method and texture method on three different quality videos of FaceForensics++ dataset

Methods	C0 Acc (%)	C23 Acc (%)	C40 Acc (%)
LBP [23]	**91.9**	85.4	76.7
HOG [23]	82.1	79.4	73.6
Proposed	87.3	**86.3**	**91.2**

Compared with deep learning methods [11,12,34], the proposed method can achieve the ideal performance. From the result of the DeepFake-TIMIT dataset, it can be seen that the accuracy of the proposed method on high quality videos and low quality videos reaches 94.4% and 92.6%, respectively. The accuracy on both quality videos is better than FWA [11], MesoNet [12] and

XceptionNet [34]. The AUC score of the proposed method on high quality videos and low quality videos reaches 98.2% and 99.5%, respectively. The AUC score on both quality videos is better than MesoNet [12] and XceptionNet [34], and only 0.4 lower than FWA [11]. From the result of the FaceForensics++ dataset, it can be seen that the accuracy and AUC score of the proposed method reach 87.3% and 94.3% respectively, which is also better than FWA [11] and MesoNet [12], but there is a gap compared with XceptionNet [34]. Compared with texture method [23], it is obvious that the performance of the proposed method is better than LBP and HOG [23] on C23 quality videos and C40 videos, and on C0 quality videos, the proposed method is about 4% lower than LBP.

In general, on the DeepFake-TIMIT [19] and FaceForensics++ dataset [34], the proposed method can achieve ideal performance, proving the effectiveness of the proposed method for DeepFake videos detection. However, some works have to be done to further improve the performance of the proposed method, especially on the FaceForensics++ dataset [34].

5 Conclusion

In order to combat the increasingly serious threat of DeepFake videos, a new method is proposed to detect DeepFake videos. Based on the defect that the texture details of some DeepFake videos are insufficient, texture features are extracted using the gradient, standard deviation, gray level co-occurrence matrix and wavelet transform. Then these features are selected and employed to SVM for detecting DeepFake videos. The experimental results show that the proposed method can effectively detect DeepFake videos. In the feature, more researches will be applied to mainstream face manipulation technologies to find out the defects of fake videos. In addition, more effective texture features will be extracted to characterize the texture details of the face region and improve the detection performance of DeepFake videos.

Funding Statement: This work is supported by the National Natural Science Foundation of China (Nos. U2001202, 62072480, U1736118), the National Key R&D Program of China (Nos. 2019QY2202, 2019QY(Y)0207), the Key Areas R&D Program of Guangdong (No. 2019B010136002), the Key Scientific Research Program of Guangzhou (No. 201804020068).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] F. Bi, X. Ma, W. Chen, W. Fang, H. Chen *et al.,* "Review on video object tracking based on deep learning," *Journal of New Media*, vol. 1, no. 2, pp. 63–74, 2019.

[2] B. Hu, H. Zhao, Y. Yang, B. Zhou and A. Noel, "Multiple faces tracking using feature fusion and neural network in video," *Intelligent Automation & Soft Computing*, vol. 26, no. 6, pp. 1549–1560, 2020.

[3] I. Korshunova, W. Shi, J. Dambre and L. Theis, "Fast Face-Swap using convolutional neural networks," in *IEEE Int. Conf. on Computer Vision*, Venice, Italy, pp. 3697–3705, 2017.

[4] Y. Nirkin, I. Masi, A. T. Tuan, T. Hassner and G. Medioni, "On face segmentation, face swapping, and face perception," in *13th IEEE Int. Conf. on Automatic Face & Gesture Recognition*, Xi'an, Shanxi, China, pp. 98–105, 2018.

[5] C. Chen, J. Ni, Z. Shen and Y. Q. Shi, "Blind forensics of successive geometric transformations in digital images using spectral method: Theory and applications," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2811–2824, 2017.

[6] M. Lu, S. Niu and Z. Gao, "An efficient detection approach of content aware image resizing," *Computers, Materials & Continua*, vol. 64, no. 2, pp. 887–907, 2020.

[7] A. Peng, K. Deng, S. Luo and H. Zeng, "Multi-purpose forensics of image manipulations using residual-based feature," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2217–2231, 2020.

[8] Y. Li, X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-DF: A large-scale challenging dataset for DeepFake forensics," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 3204–3213, 2020.

[9] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *15th IEEE Int. Conf. on Advanced Video and Signal Based Surveillance*, Auckland, New Zealand, pp. 1–6, 2018.

[10] Y. Li, M. Chang and S. Lyu, "In ictu oculi: Exposing AI created fake videos by detecting eye blinking," in *IEEE Int. Workshop on Information Forensics and Security*, Hong Kong, China, pp. 1–7, 2018.

[11] Y. Li and S. Lyu, "Exposing DeepFake videos by detecting face warping artifacts," arXiv preprint arXiv: 1811.00656, 2018.

[12] D. Afchar, V. Nozick, J. Yamagishi and I. Echizen, "MesoNet: A compact facial video forgery detection network," in *IEEE Int. Workshop on Information Forensics and Security*, Hong Kong, China, pp. 1–7, 2018.

[13] H. H. Nguyen, F. Fang, J. Yamagishi and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," in *IEEE Int. Conf. on Biometrics Theory, Applications and Systems*, Tampa, FL, USA, pp. 1–8, 2019.

[14] X. Yang, Y. Li and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Brighton, UK, pp. 8261–8265, 2019.

[15] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano *et al.,* "Protecting world leaders against DeepFakes," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, Long Beach, CA, USA, 2019.

[16] T. Jung, S. Kim and K. Kim, "DeepVision: Deepfakes detection using human eye blinking pattern," *IEEE Access*, vol. 8, pp. 83144–83154, 2020.

[17] P. Zhou, X. Han, V. I. Morariu and L. S. Davis, "Two-Stream neural networks for tampered face detection," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, Honolulu, HI, USA, pp. 1831–1839, 2017.

[18] F. Matern, C. Riess and M. Stamminger, "Exploiting visual artifacts to expose Deepfakes and face manipulations," in *IEEE Winter Applications of Computer Vision Workshops*, Waikoloa, HI, USA, pp. 83–92, 2019.

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

[20] Z. Liu, X. Qi and P. H. S. Torr, "Global texture enhancement for fake face detection in the wild," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 8057–8066, 2020.

[21] M. Bonomi, C. Pasquini and G. Boato, "Dynamic texture analysis for detecting fake faces in video sequences," arXiv preprint arXiv: 2007.15271, 2020.

[22] X. Sun, B. Wu and W. Chen, "Identifying invariant texture violation for robust Deepfake detection," arXiv preprint arXiv: 2012.10580, 2020.

[23] Y. Zhang, G. Li, Y. Gao and X. Zhao, "A method for detecting human-face-tampered videos based on interframe difference," *Journal of Cyber Security*, vol. 5, no. 1, pp. 49–72, 2020.

[24] M. Tuceryan and A. K. Jain, "Texture analysis," in *Handbook of Pattern Recognition & Computer Vision*. Singapore: World Scientific Publishing Co., Inc., pp. 235–276, 1993.

[25] R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[26] T. Ojala, M. Pietikäinen and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[27] R. Chellappa and S. Chatterjee, "Classification of textures using gaussian markov random fields," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. 33, no. 4, pp. 959–963, 1985.

[28] S. Arivazhagan and L. Ganesan, "Texture segmentation using wavelet transform," *Pattern Recognition Letters*, vol. 24, no. 16, pp. 3197–3203, 2003.

[29] A. Baraldi and F. Panniggiani, "An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, no. 2, pp. 293–304, 1995.

[30] M. S. Reis and A. Bauer, "Image-based classification of paper surface quality using wavelet texture analysis," *Computers & Chemical Engineering*, vol. 34, no. 12, pp. 2014–2021, 2010.

[31] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[32] L. Petkova and I. Draganov, "Noise adaptive wiener filtering of images," in *55th Int. Scientific Conf. on Information, Communication and Energy Systems and Technologies*, Niš, Serbia, pp. 177–180, 2020.

[33] N. Arazm, A. Sahab and M. F. Kazemi, "Noise reduction of SEM images using adaptive Wiener filter," in *IEEE Int. Conf. on Cybernetics and Computational Intelligence*, Phuket, Thailand, pp. 50–55, 2017.

[34] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies *et al.,* "Faceforensics++: Learning to detect manipulated facial images," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Seoul, Korea (South), pp. 1–11, 2019.

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