Deep forgery discriminator via image degradation analysis

Miaomiao Yu | Jun Zhang | Shuohao Li | Jun Lei | Fenglei Wang | Hao Zhou

Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China

Abstract

Generative adversarial network-based deep generative model is widely applied in creating hyper-realistic face-swapping images and videos. However, its malicious use has posed a great threat to online contents, thus making detecting the authenticity of images and videos a tricky task. Most of the existing detection methods are only suitable for one type of forgery and only work for low-quality tampered images, restricting their applications. This paper concerns the construction of a novel discriminator with better comprehensive capabilities. Through analysis of the visual characteristics of manipulated images from the perspective of image quality, it is revealed that the synthesized face does have different degrees of quality degradation compared to the source content. Therefore, several kinds of image quality-related handicraft features are extracted, including texture, sharpness, frequency domain features, and deep features, to unveil the inconsistent information and modification traces in the fake faces. In this way, a 1065-dimensional vector of each image is obtained through multi-feature fusion, and it is then fed into RF to train a targeted binary classification detector. Extensive experiments have shown that the proposed scheme is superior to the previous methods in recognition accuracy on multiple manipulation databases including the Celeb-DF database with better visual quality.

1 | INTRODUCTION

It is known that DeepFake technology is developing rapidly, in which deep learning models are used to generate and manipulate images, videos, or audio contents. The technology has attracted widespread public attention [1–3]. A non-professional user without any deep learning knowledge background can easily swap the face of one person for the one of another person in a video just by utilizing easily available tools, such as ZAO, FaceApp and FakeApp.

DeepFake has many application scenarios, such as video games, film and television program production. In addition to these entertainment-related scenarios, there are huge risks behind such technology. Its malicious used may bring serious consequences [4]. Particularly in business and politics, a tampered image, video or audio may damage a company’s long-established brand image, destroy the reputation of celebrities [5], or even cause political conflicts. In April 2018, a video about former US President Barack Obama mocking Trump in a critical tone received more than two million views on Twitter, but the video was actually tampered by a public statement made by a director in the United States, as shown in Figure 1. Katie Jones, a network character created through AI, is active in the political and business circles of the United States with a very convincing fictional image. Actually, it is just one of a large number of “spies” lurking on the social networking site LinkedIn. In the US election in November 2020, fake videos and news manipulated by DeepFake were used to confuse voters in various ways, which had a serious impact on the credibility of the news media.

The widespread dissemination of such forged contents truly breaks the conventional perception of “Seeing is believing.” For face forgery, in order to reduce the various risks and challenges that DeepFake may bring about to individuals, societies and the country, it is vital to develop a deep forgery discriminator that can effectively distinguish real images from fake images.

According to our findings, most of the existing detection methods [5–8] are mainly targeted at masking forgery, and a small part of researches is adapted to unmasking DeepFake [9–11], while few methods are suitable for both types at the same time. Thus, the application range of these single-type forgery detection models is greatly restricted. What is more, most models [4, 12–14] achieve high accuracy on rough and
low-quality (LQ) forged databases, while the detection of high-quality (HQ) forged contents becomes particularly difficult. The target of this research is to build a deep forgery discriminator with better comprehensive performance.

With the continuous optimization of computer performance and the development of some fields such as computer vision and deep learning, deep generative models based on deep neural networks (DNN) are widely used to produce fake images and videos consistent with the distribution of real data. However, since the encoder used to generate the fake face is not familiar with the skin texture and scene area other than the face-changing region in the source image, and the size and resolution of the generated synthetic face are very limited, visually visible boundary artefacts would appear when the face-changing area is merged with the rest of the area. In order to match the fake face with the surrounding background, further blur and smoothing is necessary. The positions of the bounding boxes over frames are smoothed by blurring the colour intensity of the pixels near the border to eliminate jitter generated by swapped face, resulting in the lack of texture details of the skin in the swapped region, as shown in Figure 2. In addition to changes made to the skin, blurring and smoothing can also change the presentation of facial organs, as shown in Figure 3. Additionally, in the process of masking DeepFake, the source face is covered with a mask. If the position and shape of the mask are not accurately estimated, artefacts visible to the naked eye can be exposed along the border of the mask, such as eyebrows and face contours, as shown in Figure 4.

From the above observations, it is clear that the synthesized faces do have different degrees of quality degradation compared to the source contents. The detection of synthetic faces can be analyzed by the following two factors: (1) the appearances of visual artefacts; (2) the changes of texture structure. Here, a deep forgery discriminator via image degradation analysis is presented, capable of effectively capturing the inconsistent information generated by forgery process by extracting multiple features related to image quality, thus realizing the distinction between the original face and the manipulated face.

In summary, the major contributions of this paper are as follows:

- Combining the concepts of DeepFake detection with image quality analysis, the visual characteristics of the synthetic face are analyzed from the aspect of image quality degradation. The well-built detection model can not only effectively identify low-quality forged images, but also achieve higher detection accuracy for images with high resolution and better visual quality. Meanwhile, an alternative is proposed that can effectively reduce the computational complexity.
A novel texture features descriptor is proposed and improved on the basis of the classic local binary pattern (LBP) operator. The local structure map is constructed and adopted as a weighting factor to represent the local texture information and highlight the structural loss in the forged image when counting histogram features, so as to expose the changes in texture distribution that may occur during the forgery process.

The strategy of multi-feature fusion is adopted to combine the four handpicked and representative features extracted from spatial and frequency domains. The experimental results indicate that the combined feature vector is valid enough to maximize the representation of the facial image. The established model is not only suitable for detecting the entire face synthesis forgery, but also for the identity swap manipulation. To the best of our knowledge, this is the first method designed for both unmasking and masking DeepFake detection tasks at the same time.

The rest of this paper is organized as follows: Section 2 briefly summarizes some typical works most relevant to the research of this article, including the generation and detection methods of forged images or videos; Section 3 describes in detail the establishment process of our proposed deep forgery discriminator; Section 4 demonstrates the validity and reliability of the proposed scheme through extensive experiments. The last Section is the conclusion and outlook.

2 | RELATED WORKS

2.1 | Face forgery methods

Recently, the excellent ability of deep generative models, such as generative adversarial networks (GAN) [17] and variational autoencoders (VAE) [18], are widely used to produce forged faces with realistic appearance.

GAN and its variants are one of the most important achievements in the field of machine learning in the past 10 years. They consist of two neural networks: a generator and a discriminator. The generator is to fool the discriminator by learning the data distribution and producing seemingly convincing fake samples, while the discriminator attempts to distinguish between the real samples and the fake samples generated by the generator. The learning processes of these two networks are adversarial. Similar to the GAN generator, the VAE is composed of encoder and decoder. The encoder is mainly used to reduce the dimensionality of the input layer to obtain the compressed representation of the image, whereas the decoder serves to create a new output closely resembling the original input. Figure 5 depicts DeepFake generation pipeline based on the combination of VAE and GAN. It is worth mentioning that the reason why GAN generator is superior to VAE network is that GAN discriminator rejects some bad samples.

Face forgery mainly embraces four types: entire face synthesis, identity swap, attribute manipulation, and expression swap. The first one applies GAN-based architecture [19] to create high-definition faces that do not exist in the real world. Identity swap is the most popular face tampering technology, but unlike the entire face synthesis, it replaces the face of the subject in the source video with the face of the target subject. Typically, the Deepfake-AE [20] is good at generating facial images with more realistic visual quality on the basis of autoencoder architecture with a shared encoder and two decoders of the source and target faces. Methods AttGAN [21], GDWCT [22], StarGAN [23], PGGAN [24] and CycleGAN [25] are all designed for the last two types of forgery to modify facial attributes or reshape facial expressions.
2.2 | Face manipulation detection

2.2.1 | Traditional methods

Previous studies on face-swapped image/video detection approaches were conducted respectively on three bases: the consistency of imaging equipment [26, 27], the remnants of the forgery process [5, 10, 11], and the statistical properties of an image or video itself. The first one attempts to find out the related artefacts generated by the processing unit inside a camera (such as sensor pattern noise, lens distortion, colour filter array artefacts, compression artefacts) or related artefacts generated by processing outside the camera (such as geometric transformations, contrast adjustment, blurring). The second type seeks to determine the authenticity of the image by detecting the specific fingerprints left by GAN. The last detection method is a very common strategy. On the one hand, it is used to detect unreal biological signals appearing in the forged content, including unnatural head movement [28], absence of reflex in the eye area [29], abnormal heart rate [30], mismatched eyebrows [31]. On the other hand, it utilizes hand-craft features and adopts shallow learning techniques to perform detection task. Local features like LBP, SIFT, and SURF have been used in prior works [3, 32, 33]. Akhtar et al. [4] selected 10 traditional local feature descriptors to describe image features, and used SVM to build a classification model.

2.2.2 | CNN-based methods

In recent years, deep learning has been widely used in many computer vision tasks [5, 12, 34, 35]. Zhou et al. [36] proposed a two-stream CNN for forged face detection. Afchar et al. [37] proposed the detection of videos with forged faces at a mesoscopic level of analysis. Rana et al. [38] fused multiple advanced deep learning models based on deep ensemble learning technology to create a composite classifier with excellent performance. Chang et al. [39] used the SRM filter layer to obtain the noise features of images and input them into the improved VGG16 network to detect fake content. Du et al. [40] constructed a novel detection method to improve the generalization accuracy by making predictions relying on correct forgery evidence. Nguyen et al. [8] trained a CNN model to detect both forged images and segment forged regions simultaneously, which belongs in a multi-task learning problem. Chen et al. [41] proposed a unified framework that takes into account the spatial features within a single frame and the temporal inconsistencies between frames. Similarly, Nguyen et al. [42] proposed a three-dimensional CNN model that can learn spatio-temporal features from an adjacent frame sequence in the video.

2.2.3 | Integration of traditional methods and CNN

Admittedly, CNN-based methods do have many advantages over traditional approaches. They can extract features of different scales from the input image. Their feature expression ability is stronger, so the abstract features can be automatically obtained. However, CNN-based methods are more costly in resource consumption, more difficult, and time-consuming during training. Specifically, in order to obtain high accuracy, they require a large number of training samples, various optimization techniques, and powerful hardware support. In addition, CNN is poor in both interpretability and feature extraction. Fortunately, traditional methods are superior in the above aspects. Therefore, CNN is treated as a feature extractor and fused with traditional methods in this paper. That is to say, the deep features extracted from CNN are used as supplementary features and fused with multiple hand-crafted features to enhance the detection capabilities of the established model. These features describe specific tampered artefacts and modification traces in the forged images, which can help effectively capture the inconsistent biological signals and unreal detailed information. Therefore, classification models such as Random Forest (RF) or other small-scale neural networks with small parameter spaces can fully meet the needs of classification task. Compared with deep learning models, these small-scale classifiers have lower requirements on the scale of training samples and have more obvious advantages in terms of computational complexity.

3 | METHODOLOGY

Here, the main concern is the detection of manipulated faces. The flowchart of the proposed method is illustrated in Figure 6. Specifically, for a given input face video, it is first converted into a series of image sequences, and then face detection is performed on each individual frame to crop out the face region. It is worth mentioning that the face is detected with Face Parts Detection tool based on CascadeObjectDetector of FrontalFaceCART, LeftEye, RightEye, Mouth, and Nose. Next, the feature vectors of the face are extracted, which is the key step of our method. Finally, the extracted feature vectors are fed into the RF to train the binary classification model.

3.1 | Face features extraction

The key idea of this step is to obtain representative discriminant features that can be distinguished from real faces by analyzing the characteristics of forged faces. From the perspective of image quality analysis, the extraction is conducted concerning texture features, gradient-based sharpness indicators, and frequency domain features related to image quality. Then they are fused with deep features to ensure effective capture of the inconsistent information generated by fake faces and realize the accurate distinction between these two types of facial images. The entire process of face features extraction is shown in Figure 7. In the preprocess, the cropped face is turned into a greyscale image, and its size is then normalized to 256 × 256. Note that, different from face recognition task, the cropped face does not need to be aligned here.
3.1.1 Texture features extraction based on structural weighting

As mentioned in the introduction, the fake face region generated by the deep generation network will produce obvious edge artefacts during the process of fusion with the surrounding scene. After blurring, the facial skin and organs (such as teeth, eyes) become too smooth, resulting in loss of texture details and making them no longer consistent with the structural distribution of the background. The texture features of the image are represented by the grey distribution of the central pixel and its surrounding spatial neighbourhood, reflecting the regular and periodic changes of the structural organization and arrangement properties of the image surface. Therefore, the image texture is one of the important features to expose the forgery phenomenon.

The classic LBP, a simple and effective local texture descriptor, can capture the rich structural information of the image and has grey-level invariance. It is currently widely used in the fields such as face detection and texture classification. Traditionally, LBP features are extracted from the original image, but in our method, the mean subtracted contrast normalized (MSCN) coefficients [43] calculated by formula (1) is used to extract LBP features.

\[ \hat{I}(i,j) = \frac{I(i,j) - \mu(i,j)}{\sigma(i,j) + C}, \]  
\[ \mu(i,j) = \sum_{k=-K}^{K} \sum_{l=-L}^{L} w_{kl} I_{k,l}(i,j), \]
\[ \sigma(i, j) = \sqrt{\sum_{k=-K}^{K} \sum_{l=-L}^{L} w_{k,l}[I_{k,l}(i,j) - \mu(i,j)]^2}, \]  

(3)

where, \( I \) is a grey-scale image with a size of \( M \times N, \) \( \mu \) represents the MSCN coefficient matrix, \( i \in \{1, 2, ..., M\}, \) \( j \in \{1, 2, ..., N\}, \) \( C (C = 1) \) is used to prevent the denominator of the fraction from being zero, \( w = \{w(k, l) | k = -K, ..., K; l = -L, ..., L\} \) symbolizes the Gaussian weighted filter window with central symmetry, \( K = L = 3. \)

Generalized Gaussian distribution (GGD) is often used in the statistical analysis of image signals, and its shape parameter \( \alpha \) and variance parameter \( \sigma^2 \) can be regarded as image features for classification or regression. Here, the zero-mean GGD is used to fit the distribution of the MSCN coefficient matrix, and its parameter estimates \( [\alpha, \sigma^2] \) are extracted as a set of features, characterizing a two-dimensional vector.

The circular LBP operator serves to compare the central pixel value with the pixel value of the remaining points in the circular neighbourhood with a radius of \( R. \) If the surrounding pixel value is greater than the central pixel value, the pixel point is marked as 1, otherwise 0. LBP calculation process is expressed as formulas (4) and (5):

\[ \text{LBP}_{P,R}(X, Y) = \sum_{j=0}^{P-1} 2^j S(x), \]  

(4)

\[ S(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}, \]  

(5)

where, \( R = 1, \) \( P = 8, (X,Y) \) represents the coordinates of the central pixel, \( x = x_i - x_c \) is the grey-scale difference between the \( i \)th pixel \( x_i \) and the central pixel \( x_c \) in the neighbourhood. In this case, a total of \( 2^P = 256 \) binary patterns can be generated. However, as \( P \) increases, the number of generated patterns will increase exponentially. Moreover, this calculation method does not have rotation invariance. By improving the original LBP operator, the rotation-invariant uniform pattern \( \text{LBP}_{P,R}^{uni2} \) is defined as:

\[ \text{LBP}_{P,R}^{uni2}(X, Y) = \begin{cases} \sum_{j=0}^{P-1} S(x), & U \leq 2 \\ P + 1, & \text{others} \end{cases}, \]  

(6)

where, \( U \) is the number of spatial transitions (that is, the number of 0–1 changes). In this case, the number of rotation-invariant uniform patterns is \( P + 1 = 9, \) and all non-uniform patterns are grouped into one. In this way, the texture histogram vector of each image is only 10-dimensional, greatly reducing the calculation and storage space. Therefore, this type of LBP operator is adopted in our method to extract texture features under the MSCN coefficients.

In order to make full use of the structure information of the forged image, the local structure map is constructed and used as a weighting factor to highlight the structural loss in the forged image when counting the normalized histogram of LBP features, as shown in formula (7). The construction method of the local structure map MP is used to extract the phase congruency (PC) [44] features from the facial grey-scale image, and then select the larger one of the PC and the MSCN coefficient at each pixel point, as shown in formula (8).

\[ H(k) = \sum_{j=1}^{M} \sum_{i=1}^{N} w_{i,j} \delta(x), \]  

(7)

\[ \delta(x) = \begin{cases} 1, & x = 0 \\ 0, & \text{others} \end{cases}, \]  

\[ \text{MP}(i, j) = \max(\text{PC}(i, j), \hat{I}(i, j)), \]  

(8)

where, \( x = \text{LBP}_{P,R}^{uni2} - k, k \in [0, 9] \) represents the number of generated LBP patterns, \( w_{i,j} \) is the weighting factor, the amplitude value of the MP.

The advantages of the local structure map constructed by the above method are shown in Figure 8. Figures 8a and 8e are a real face and fake face from the DFD dataset, respectively. Figures 8b, 8c, and 8d, respectively, show the MSCN coefficients, PC map and MP of Figure 8a, and Figures 8f, 8g, and 8h, respectively, represent the MSCN coefficients, PC map and MP of Figure 8e. It can be clearly observed that the MSCN coefficients focus on describing the fine details of the image, while the PC map serves well in capturing large-scale feature information such as the edges and lines. The combined local structure map MP can fully describe the rich edge structure and texture details of the image. Since PC is immune to the brightness and contrast of the image, it is very suitable for facial images taken under different lighting environments. Compared with the real image, this change can be clearly observed on the MSCN, PC and MP maps due to the fact that the forged image is blurred in the face-changing area. Especially for Figure 8h, not only are there artefacts visible to the naked eye at the boundary of the face-changing area, but at the same time, the contours of facial attributes (such as eyes, nose, and mouth) are no longer sharp and clear. Therefore, using the local structure map as the weighting factor of the LBP features is beneficial to highlight the structural loss of the forged image.

It is worth mentioning that since in observing objects, the human visual system follows coarse-to-fine pattern, multi-scale feature extraction can simulate this process. Therefore, low-pass filtering and \( 2 \times 2 \) downsampling are used in this paper to extract the LBP features of facial images at five scales. In this way, a 60-dimensional texture feature vector is generated for each input image.

### 3.1.2 Gradient-based sharpness measurement

When the texture distribution of the image is quite different, it is very effective to extract texture features. However, when
FIGURE 8 Examples of local structure map. Figures (a) and (e) are a real face and fake face from the DFD dataset, respectively. Figures (b), (c) and (d), respectively, show the MSCN coefficients, PC map and MP of Figure (a), and Figures (f), (g) and (h), respectively, represent the MSCN coefficients, PC map and MP of Figure (e).

the distinction between the texture information of the image is not clear, such as when the degree of density is similar, and when the thickness is similar, the texture features are usually difficult to accurately reflect the human visual perception. Different from texture, the sharpness indicator reflects the contrast between light and dark in the fine layers of the image, as well as the degree of virtuality and realness of the image contour boundary, which is considered one of the important indicators for evaluating image quality. Since the sharpness of masking-based manipulated images in the face-changing area has decreased to varying degrees, it is necessary to extract the sharpness indicator from the pixel level to jointly describe the characteristic of the forged images.

The grey value of the pixel at the edge of the blurred image changes greatly, so their gradient value is greater. In this paper, two gradient-based sharpness indicators are extracted as one of the features of detecting DeepFake.

The first sharpness indicator is the variance function, which reflects the dispersion degree of the image grey distribution. The smaller the variance, the less the range of grey value changes, and the lower the dispersion degree of the grey distribution would be, and vice versa. For an $M \times N$ grey-scale image $I$, the variance function can be expressed as:

$$\sigma^2 = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} [I(i,j) - \mu]^2,$$  \hspace{3cm} (9)

where, $\mu = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} I(i,j)$ represents the mean value of all pixels in image $I$.

The second sharpness indicator is the Tenengrad function. It uses the Sobel operator to extract the gradient magnitude of the image in the horizontal and vertical directions (obtained by convolution operation of image and two convolution kernels), denoted as $G_x(i,j)$ and $G_y(i,j)$, respectively. The expression of the Tenengrad function is as follows:

$$F = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} [G_x^2(i,j) + G_y^2(i,j)],$$  \hspace{3cm} (10)

where, $G_x(i,j)=[2 \ 0 \ -2] \ast I(i,j)$, $G_y(i,j)=[1 \ 0 \ -1] \ast I(i,j)$.

The larger the value calculated by the Tenengrad function, the higher the image clarity is.

Through the above process, a two-dimensional gradient-based sharpness feature vector is generated for each input image.

3.1.3 Frequency domain features extraction

Unlike masking DeepFake, for instance, identity swap and expression swap, the entire face synthesis uses a powerful GAN-based method to create highly realistic and high-quality facial images that do not exist in the real world. This manipulation does not involve image blur or other distortion operations. Although it is difficult to distinguish them from real images with the naked eye, they do unveil forged traces in some ways. A previous study [45] has revealed that real faces and fake faces show significantly different spectral distributions in the frequency domain. Inspired by it, the recognition problem of unmasking DeepFake is taken into account by extracting the frequency domain features of the image.

Our research involves the adoption of the frequency domain features extraction based on the contrast sensitivity function [46]. First, the grey-scale image is changed from the spatial domain to discrete cosine transform (DCT) coefficients. Then,
calculations are conducted concerning the contrast energy values in low frequency (LF), middle frequency (MF) and high-frequency (HF) regions in $4 \times 4$ DCT blocks. The expressions are defined as formulas (11)–(13):

$$L = \sum_{(u,v) \in LF} p(u,v),$$

$$M = \sum_{(u,v) \in MF} p(u,v),$$

$$H = \sum_{(u,v) \in HF} p(u,v),$$

where, $p(u, v)$ represents the normalized value of DCT coefficients at $(u, v)$. The three regions LF, MF and HF are divided according to their sensitivity to distortion [47].

The features of these three frequency domain components are combined through a dot multiplication operation to obtain a frequency domain feature map. In order to reduce the dimensionality of the feature space and lower the computational complexity of the model, the mean, standard deviation and entropy of the feature map are adopted as the extracted frequency domain features. In this way, a three-dimensional frequency domain feature vector is generated for each input image.

Note that the main component of an image is the low-frequency information, forming the basic grey level of the image; the middle-frequency information determines the basic structure of the image, forming the main edge structure; the high-frequency information forms the edges and details of the image, serving as a further enhancement to image content based on middle-frequency information. The masking DeepFake has changed the edge structure and detailed information of the face region. Therefore, the extraction of frequency domain features is also helpful to distinguish the masking type of fake images.

### 3.1.4 Deep features extraction

With the extensive development of deep learning in various fields, various CNN models have been developed to solve tasks, such as image classification, image recognition, and object detection. Compared with traditional methods of extracting hand-crafted features, CNN-based methods show powerful feature expression capabilities. To be more specific, the shallow network is used to learn local features like lines, textures, and details of images. As the number of layers increases, global features such as shape and contour are captured. The fully connected layer combines all the previously learned features, with the last layer acting as a classifier in the entire network.

Our research considers the advantages of CNN that can automatically extract features, so transfer learning is used to obtain 1000-dimensional deep features vector from the last fully connected layer of the VGG16 network pre-trained on ImageNet. When the computational resources are not adequate enough to train a deep learning model for several days on GPUs or TPUs, the adoption of transfer learning is a good choice. The adoption of the VGG16 can be explained by the fact that it uses small convolution kernels with the size of $3 \times 3$ in the convolution layer, greatly reducing the number of parameters to be learned and lowering the computational complexity of the model. In addition, the number of its convolutional channels increases by a factor of two, making up for the information loss after convolution in depth. The deep features of the facial image extracted by the VGG16 network are used as a supplement to the hand-crafted features, enhancing the feature description ability of the forged image.

### 3.2 Features fusion and binary classification model construction

After Section 3.1, the 3-dimensional frequency domain feature vector (denoted as v1), 2-dimensional gradient-based sharpness vector (denoted as v2), 60-dimensional texture feature vector (denoted as v3), and 1000-dimensional deep feature vector (denoted as v4) are obtained. They are then fused to generate a representative feature vector $f \approx [v1, v2, v3, v4]$ to describe each facial image, a total of 1065 dimensions. As previously described, these extracted features unveil specific tampered artefacts or traces in the forged image, effectively capturing inconsistent biological signals and unreal details. Therefore, a small-scale classification network can completely meet the classification needs. SVM has been used as the classification model in many methods [4, 20]. In our method, RF tool is applied to perform the task of forgery recognition, for its good performance in processing high-dimensional data, its robustness to noise and outliers in the process of learning classification, When more decision trees are added, the training error and generalization error of the model will gradually converge, effectively reducing the phenomenon of overfitting. In subsequent experiments, it has also been proved that the use of RF in our method is better than SVM.

In the training phase of the model, the feature vectors extracted from all real and fake facial images on the database and the corresponding label sets are input into RF together. In the testing step, the input facial image is first subjected to feature extraction, and then the feature vector is fed into the well-trained RF classification model. The output is the predicted label value. Here, the parameters of RF are set as the value $(ntree, mtry) = (500, 3)$.

### 4 EXPERIMENTAL SETUP AND RESULTS

This section concerns an introduction of several mainstream DeepFake databases and the metrics used to evaluate the performance of the classifier, followed by a series of experiments carried out to prove the effectiveness and reliability of the proposed scheme.
TABLE 1  Summary of the four manipulation databases

| Databases              | Number of Participants | Release Date  | Source        | Manipulation type | RealVideos (Frames) | FakeVideos (Frames) | Extracted Real Frames | Extracted Fake Frames |
|------------------------|------------------------|---------------|---------------|-------------------|--------------------|---------------------|-----------------------|-----------------------|
| Celeb-DF               | 59                     | November 2019 | YouTube       | Identity swap     | 590 Videos(225.4k) | 5 639 Videos(2116.8k) | 7000                  | 7000                  |
| DFD                    | 28                     | September 2019| Actors        | Identity swap     | 363 Videos(315.4k) | 3068 Videos(2242.7k) | 5000                  | 5028                  |
| Deepfake-TIMIT-HQ      | 32                     | December 2018 | Actors        | Identity swap     | 320 Videos(34.0k)  | 320 Videos(34.0k)   | 6400                  | 6400                  |
| Deepfake-TIMIT-LQ      | 32                     | December 2018 | Actors        | Identity swap     | 320 Videos(34.0k)  | 320 Videos(34.0k)   | 6400                  | 6400                  |
| Face Synthesis         | —                      | —             | —             | Entire face synthesis | 20k Images         | 20k Images           | 12,000                | 12,000                |

4.1  | Manipulation databases

Experiments were conducted on three publicly available manipulation datasets, namely Deepfake-TIMIT [48], Celeb-DF [16], and DFD [15], and the Face Synthesis database was constructed by our research group. These databases cover two typical types of facial forgery (name identity swap and entire face synthesis) and tampered images with different visual qualities. Table 1 lists the basic information of these databases.

4.1.1  | Deepfake-TIMIT database

The manipulated type of this database belongs to identity swap technique, among which fake faces were generated using the open source GAN-based face-swapping tool. This database involved 16 subjects with similar appearances from VidTIMIT database [49] to form a video set of 32 subjects. Each subject had 10 original videos, corresponding to 320 real videos. Fake videos were generated by two models: a lower quality (LQ) with 64 × 64 input/output size model, and higher quality (HQ) with 128 × 128 size model. So, a total of 640 face-changing videos were created. In the experiment, 20 consecutive frames were taken for each video (real and fake). Eventually, 6400 real images, 6400 HQ fake images, and 6400 LQ fake images were obtained. To facilitate description, the Deepfake-TIMIT database is into Deepfake-TIMIT-LQ database and Deepfake-TIMIT-HQ database according to the different resolutions of the forged images, used to evaluate the detection performance of the classifier on LQ and HQ fake images, respectively.

4.1.2  | Celeb-DF database

The Celeb-DF database, a new database released at the end of 2019, adopts an improved DeepFake synthesis method to generate tempered videos. It is considered to be the manipulation database with better visual quality, in which the forged images have almost no splicing edges, colour mismatches, and other visible visual artefacts, and even the most advanced detection methods perform generally on it. This database contains 590 real videos of about 59 celebrities downloaded from Youtube and 5639 synthesis videos. In our experiment, 408 real videos and 795 synthesis videos were selected randomly from the Celeb-DF database. One frame was taken every 20 frames for each real video, and one frame every 45 frames for each fake video. A total of 11,541 real frames and 7039 fake frames were taken. In order to balance the samples, 7000 real frames and 7000 fake frames were randomly selected again from all the extracted image frames.

4.1.3  | DFD dataset

DFD dataset, a part of FaceForensics++ dataset [2], is one of the most popular databases belonging to identify swap manipulation type. This database contains 363 original sequences from 28 paid actors in 16 different scenarios, and 3068 fake videos generated through the DeepFake FaceSwap GitHub implementation. Most of the original videos in this database are frontal and unobstructed face frames, so the generated tampered video looks extremely realistic. In our experiment, 1676 fake frames and 2500 real frames were randomly taken. In order to balance the samples, all fake frames were flipped horizontally and vertically for data amplification, and all real frames were flipped vertically, so altogether, 5028 fake frames and 5000 real frames were created.

4.1.4  | Face Synthesis database

This database was constructed by our research group to evaluate the classification ability of the detection model to the entire face synthetic forgery. The real faces in the Face Synthesis database come from two popular public datasets CelebA [24] and Flickr-Faces-HQ [19]. The two datasets 100K-Faces [50] and www.thispersondoesnotexist.com provide us with fake faces, and all fake faces were generated using StyleGAN architecture. 6000 faces were randomly selected from each sub-dataset. Therefore, the Face Synthesis database consists of 12,000 real faces and 12,000 fake faces. Figure 9 depicts some samples on the Face Synthesis database.

4.2  | Evaluation metrics

4.2.1  | Accuracy and error rate

These two metrics are commonly used to evaluate the performance of classification model. The accuracy can be expressed...
where, \( TP \) is the number of real samples correctly classified by the classifier, \( TN \) is the number of fake samples correctly classified by the classifier, \( FP \) is the number of fake samples incorrectly marked as real samples, and \( FN \) is the number of real samples incorrectly marked as fake samples. The error rate of the classifier can be calculated with \( 1 - \text{Acc} \).

4.2.2 Receiver operating characteristic curve and area under curve

The receiver operating characteristic (ROC) curve, a visual tool for evaluating the pros and cons of a two-class model, shows the trade-off relationship between the true positive rate (TPR, \( TPR = \frac{TP}{TP + FN} \)) and the false positive rate (FPR, \( FPR = \frac{FP}{FP + TN} \)) of the model, and can easily detect the influence of any threshold on the generalization performance of the classifier. The closer the ROC curve is to the upper left corner, the higher the accuracy of the model. Conversely, the closer the ROC curve is to the diagonal, the lower the accuracy of the model. When two or more classifiers are compared, in order to quantitatively measure the results of ROC, area under curve (AUC) is used to represent the area under the ROC curve. The larger the AUC value, the higher the accuracy of the classifier, and vice versa.

### Table 2

| Databases               | Acc  | Error Rate | AUC  |
|-------------------------|------|------------|------|
| Celeb-DF                | 92.13| 7.87       | 92.55|
| DFD                     | 99.44| 0.56       | 99.33|
| Deepfake-TIMIT-HQ       | 100  | 0          | 99.98|
| Deepfake-TIMIT-LQ       | 99.96| 0.04       | 99.85|
| Face synthesis          | 89.25| 10.81      | 89.17|

4.3 Results and analysis

#### 4.3.1 Classification performance results

In this part, three metrics of Acc, Error Rate and AUC are listed on four manipulation databases to evaluate the classification performance of the proposed model. In order to obtain stable and reliable results, 1000 iterations were performed on each experiment, and the median was finally taken. Similar to most methods, for each database, 80% of the samples were randomly selected for training and the remaining 20% were used for testing. The experimental results are shown in Table 2.

In order to evaluate the impact of the number of training samples on the classification results, different training-test setups were adopted to carry out experiments on each database. The results are depicted in Figure 10. Three experiments were performed on each database, that is, 90%, 80% and 50% of the samples were randomly selected for training, and the remaining samples were regarded as the test set. Obviously, for each database, the classification accuracy of the model increases as the number of training samples increases on each database. For the Deepfake-TIMIT-HQ database, the accuracy of our scheme

![Figure 9](image1.png) **Figure 9** Examples of images from the Face Synthesis database. The first two rows are real faces and the last two rows are fake faces generated using StyleGAN architecture.

![Figure 10](image2.png) **Figure 10** Accuracy scores of our method under different training-test setups.
TABLE 3  AUC scores (%) of various methods on compared datasets

| Methods               | Release date | Celeb-DF | Deepfake-TIMIT-HQ | Deepfake-TIMIT-LQ | DFD   |
|-----------------------|--------------|----------|-------------------|-------------------|-------|
| Meso-4 [37]           | 2018.09      | 54.8     | 68.4              | 87.8              | 76.0  |
| Mesolnception-4       | 2018.09      | 53.6     | 62.7              | 80.4              | 75.9  |
| HeadPose [28]         | 2018.11      | 54.6     | 53.2              | 55.1              | 56.1  |
| Two-stream NN [36]    | 2018.03      | 53.8     | 73.5              | 83.5              | 52.8  |
| FWA [13]              | 2018.11      | 56.9     | 93.2              | 99.9              | 74.3  |
| DSP-FAW               | 2019.11      | 64.6     | 99.7              | 99.9              | 81.8  |
| VA-MLP [29]           | 2019.01      | 55.0     | 62.1              | 61.4              | 69.1  |
| VA-LogReg             | 2019.01      | 55.1     | 77.3              | 77.0              | 77.2  |
| Xception-raw [2]      | 2019.01      | 48.2     | 54.0              | 56.7              | 53.9  |
| Xception-c23          | 2019.01      | 65.3     | 94.4              | 95.9              | 85.9  |
| Xception-c40          | 2019.01      | 65.5     | 70.5              | 75.8              | 65.8  |
| Multi-task [8]        | 2019.06      | 54.3     | 55.3              | 62.2              | 54.1  |
| Capsule [34]          | 2019.10      | 57.5     | 74.4              | 78.4              | 64.0  |
| DFT-MF [12]           | 2020.04      | 71.25    | 73.1              | 98.7              | —     |
| FSSPOTTER [41]        | 2020.06      | 77.6     | 98.5              | 99.5              | —     |
| Face X-ray [5]        | 2020.06      | 74.76    | —                 | —                 | —     |
| NA-VGG [39]           | 2020.09      | 85.7     | —                 | —                 | —     |
| VTD-Net [35]          | 2020.07      | 90.2     | —                 | —                 | —     |
| Ours                  | —            | 92.55    | 99.98             | 99.85             | 99.33 |

reaches 100% when the training samples account for 80% and 90% of the total.

4.3.2  Comparison with state-of-the-art methods

In this section, to show that our method has better classification performance, a comparison was made between our method and several existing advanced models on different databases.

First of all, the proposed method was compared with eighteen top performing forgery detection models on three latest databases (Celeb-DF, Deepfake-TIMIT and DFD). Because the Celeb-DF database contains tampered videos with better visual quality, the visual artefacts are not obvious, so it is difficult to achieve the desired detection accuracy on this database. The Deepfake-TIMIT database contains both LQ and HQ forged content, which is mainly used to evaluate the ability of the detection model to recognize fake images with different qualities. The comparison results are shown in Table 3. Bold faces correspond to the top performance. Note that the 18 models compared are relatively well-known methods in the past 3 years. Clearly, the classification performance of our model is far superior to the other mainstream methods. Particularly, the AUC score of our method is 2.35% higher than the best AUC results of the 18 methods on the Celeb-DF dataset. For the Deepfake-TIMIT-LQ dataset, the result of our method is obviously very close to the best result. Most importantly, the existing models rarely achieve high detection accuracy on both HQ and LQ forged images, while our method shows an absolute advantage in this regard. The last column in Table 3 depicts the AUC scores of different methods on the DFD database. It can be clearly seen that, our method far exceeds other methods, and the AUC is 13.43% higher than the second-ranked method, showing that our method is effective in detecting forged images created through DeepFakeFaceSwap technology.

In order to show the advantages of the proposed scheme more intuitively, Figure 11 illustrates the AUC scores of our method and the average frame-level AUC performances of other 18 detection methods on each dataset. Figure 12 shows average AUC performance of each detection method on all evaluated datasets. Celeb-DF is considered to be the well-known face swap database with better visual quality. The average AUC of the existing methods on this database is about 63%, while that
of our method reaches 92.55%, which is a great improvement. The average AUC scores of the existing detection methods on the DFD, Deepfake-TIMIT-HQ and Deepfake-TIMIT-LQ databases are 63.35%, 74.02% and 80.81%, respectively, while the AUC scores calculated by our method are 99.33%, 99.98% and 99.85%, respectively. What is more, the average AUC score of our method on all evaluated databases is also significantly higher than those of other eighteen methods, concretely, the average score of our method is higher than that of the second-ranked method by nearly 7%. In summary, compared with the existing methods, the proposed scheme shows an overwhelming advantage in terms of accuracy.

In addition, the difficulty of detecting manipulations has been evaluated in a variety of studies based on the entire face synthesis. Table 4 shows a comparison of the most relevant methods in this field. Although our method is not the best one and only reaches the medium level, the average score of our method is higher than that of the second-ranked method by nearly 7%. In summary, compared with the existing methods, the proposed scheme shows an overwhelming advantage in terms of accuracy.

### 4.3.3 Single feature performance analysis

Identity swap is the most popular type of forgery, so in order to determine the impact of individual feature component on the classification results, feature separation experiments were conducted on three identity swap databases. Specifically, only one type of feature was adopted to train the classification model in each experiment. It was decided that 80% of samples were used for training and 20% for testing. The accuracy results of each experiment are displayed in Table 5. Clearly, texture features play a major role in this classification task, showing that the proposed texture extractor can effectively capture the structural loss of identity swap forged content. In additional, the performance of deep features extracted on the pre-trained VGG16 model is slightly inferior to texture features, being a comprehensive representation of the texture, details and edge contour features. All in all, identity swap forgery destroys the distribution of texture structure in the original image. In the process of detecting forgery, the modification traces can be easily unveiled by extracting texture-related features, and compared with automatically obtained abstract features. The detection task mainly targets at the manually designed texture features.

### 4.3.4 Comparison of performance using different texture operators

One of the contributions in this paper is proposition of a novel texture features descriptor (here denoted as method “wlbp”) based on the classic LBP operator. As described in Section 3.1.1, the local structure map is constructed and adopted as weighting factors to highlight the structural loss in the forged image when counting histogram features, so as to expose the changes in texture distribution that may occur during the forgery process. In order to highlight the advantages of wlbp, it was compared with a texture descriptor that does not use local structure map for weighting (here denoted as method “lbp”). Specifically, the texture extractor in our proposed scheme adopted lbp method, while the other operations remained unchanged. Afterwards, the performance was compared with our original scheme. The experimental results are shown in Table 6. The second row represents our original scheme, and the third row shows the modified scheme. Clearly, the performance of the method using wlbp is significantly better than that using lbp, demonstrating the effectiveness of the wlbp in describing the local texture features of the image.

![Figure 12](image.png)

**Figure 12** Average AUC (%) performance of each detection method on all evaluated datasets

|TABLE 4| AUC comparison between our model and some other methods |
|---|---|---|
|Methods| Release date| AUC |
|SATURATION Cues based [11]| 2018.12| 70.0% |
|FakeSpotter [51]| 2020.07| 84.78% |
|Incremental learning [9]| 2019.10| 99.3% |
|GAN-related features [10]| 2019| 99.50% |
|Ours—|—|89.17% |

|TABLE 5| Accuracy scores (%) of feature separation experiments |
|---|---|---|---|---|---|
|Databases| Texture features| Sharpness Indicators| Frequency Domain features| Deep features| All |
|Celeb-DF| 91.25| 81.07| 84.96| 89.89| 92.13 |
|DFD| 99.39| 88.15| 89.76| 99.31| 99.44 |
|Deepfake-TIMIT-HQ| 99.62| 65.53| 73.63| 99.84| 100 |
|Deepfake-TIMIT-LQ| 99.88| 65.64| 75.51| 99.62| 99.96 |
the MSCN coefficients of 13a,e, which correspond to the lbp method mentioned above. Figure 13c,g shows the histograms of the local structure maps constructed by Figure 13a,e, namely, the weighting factors. Figures 13d and 13h, respectively, describe the weighted LBP histograms of (a) and (e) under the MSCN coefficient, which correspond to the wlbp method. It is worth mentioning that in the histogram, patterns 0 and 8 represent spots or noises patterns 1 to 7 reflect the corners and edges of the image, and pattern 9 represents non-uniform high-frequency information. It can be seen that, compared with the histograms b and f, the weighted LBP histograms d and h highlight the structural information of the original image, meaning that the noise information (pattern 0) and edge structure information (pattern 1) have been enhanced. Compared with the histograms b and d, due to the blur or smoothing operation in the fake image e, the corner and edge information in the histograms f and h are weakened, meaning that the proportion in modes 2 to 6 is reduced.

In summary, when the proposed texture descriptor uses MSCN coefficients to calculate the LBP histogram features weighted by the local structure map, the local texture information can be effectively represented and the structural loss of the forged image can be emphasized.

TABLE 6 Comparison of AUC (%) performance using different texture features

| Methods     | Celeb-DF | Deepfake-TIMIT-HQ | Deepfake-TIMIT-LQ | DFD | Face Synthesis |
|-------------|----------|-------------------|-------------------|-----|---------------|
| Ours-wlbp   | 92.55    | 99.98             | 99.85             | 99.33| 89.17         |
| Ours-lbp    | 90.84    | 99.62             | 99.70             | 99.14| 87.94         |

FIGURE 13 Examples of texture descriptor. Figures (a) and (c) are a real face and fake face from the DFD database, respectively. Figures (b) and (f), respectively, represent the LBP histograms calculated under the MSCN coefficients of (a) and (c). Figures (c) and (g) shows the histograms of the local structure maps constructed by (a) and (c). Figures (d) and (h), respectively, describe the weighted LBP histograms of (a) and (c) under the MSCN coefficient, which correspond to the wlbp method.

4.3.5 Comparison of RF and SVM classifiers

In order to prove that the adoption of RF tool in our method is more sensible than using SVM, the performances of these two classifiers on four databases were made. Figure 14 depicts the accuracy computed by using RF and SVM with two different kernel functions (i.e. RBF and Sigmoid). Similarly, in each experiment, 80% of the samples were selected randomly on the database as the training set, and the remaining 20% as the testing set. 1000 iterations were performed for each experiment, and the median was finally taken. Obviously, the performance of RF is better than any SVM type on any database.

4.3.6 Computational complexity analysis

Time taken or computational complexity is indeed one of the important issue worth considering in machine learning. Table 7 lists the average time taken for each image passing through
each component in our scheme. Our proposed scheme was programmed with Matlab 2019 on the computer with Intel(R) Xeon(R) Gold 6248 CPU and 256GB RAM. The extraction of the first three features takes a shorter time because there are no complex calculations involved. The extraction speed of deep features is relatively slower, depending on the configuration of the computer hardware. The training process takes far less time, since RF itself contains much fewer parameters than CNN model. The average time taken to train an image is only about 0.0071 s. Similarly, it only takes 0.0002 s to test the authenticity of an image. Generally speaking, the overall time taken is acceptable.

If there are high requirements on the running time of the model, it can be achieved by ignoring the extraction of deep features, just sacrificing a small loss of accuracy. In this case, the computational complexity of the proposed scheme is generally similar to the existing methods based on manually extracting features, significantly lower than the CNN-based methods. This is an alternative solution that can significantly reduce computational complexity.

4.3.7 Image quality assessment

Since our proposed method is to analyze the visual characteristics of fake facial regions from the aspect of causing image quality degradation, it is also suitable for evaluating the quality of distorted images. To prove this, experiment was conducted on a mainstream Image Quality Assessment (IQA) database CSIQ [52]. The database contains 30 reference images; each reference image has 6 distortion types; each distortion type has 4 or 5 distortion levels. Therefore, there are altogether 866 distortion images. The experimental process is as follows. First of all, all the features of the distorted images were extracted according to the flowchart of Figure 7. Then, 80% of them were picked out for training, meaning that their feature vectors and the corresponding subjective scoring values (or labels) were input into the RF strategy to construct a regression mapping model. Finally, the feature vectors of the remaining 20% of the images were fed into the well-trained RF model, and the output was the image quality predicted by the model. The Pearson linear correlation coefficient (PLCC) indicator was used to quantitatively describe the prediction accuracy of the model. The closer the PLCC score is to 1, the higher the prediction accuracy of the model. After 1000 iterations, the median result is PLCC = 0.9255, which is a gratifying result. Note that by taking into account the colour features of the image and the human visual attention mechanism, better results will be obtained.

5 CONCLUSIONS

This study presents a deep forgery discriminator based on image degradation analysis. By analyzing the visual characteristics of the synthesized faces, several handicraft features related to image quality are extracted, which have a great influence on distinguishing real images from fake ones. Specifically, the local structure map is delicately constructed and used as a weighting factor to highlight the structural loss in the forged image while extracting the texture features. Two gradient-based sharpness indicators are adopted to describe the degree of virtuality and realness of the image contour boundary. Similarly, the contrast energy values are captured under different frequency bands by taking into account the information changes in the frequency domain of the forged image. Last but not the least, 1000-dimensional deep features obtained from VGG16 are merged with the above three manual features to effectively expose the fake details and forged traces generated by the synthetic faces. Finally, the RF classification model is well trained to accurately distinguish real face from synthetic ones.

The experimental results show that the proposed scheme is superior to many advanced methods. It achieves higher detection accuracy. For forged faces of different quality, our scheme outdoes the most advanced models in that it can accurately distinguish not only the low-quality faces, but the faces with high resolution and better visual quality as well. Furthermore, it has excellent comprehensive performance, it is suitable for both identity swap DeepFake and entire face synthesis DeepFake, possessing more extensive applications.

Our further research efforts will focus on researching more advanced forgery detection technology with better generalization, and also on studies about the reverse traceability of forged algorithms and forged videos to achieve breakthrough in anti-artificial intelligence technology.

ACKNOWLEDGEMENTS

This work was funded by the National natural Science Foundation of China (NSFC) with grant numbers 61806215 and 61671459, and National University of Defense Technology Scientific Research Project with grant number ZK20-48.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

CONFLICT OF INTEREST

The authors declare no conflict of interest.
REFERENCES

1. Thakur, R., Rohilla, R.: Recent advances in digital image manipulation detection techniques: A brief review. Forensic Sci. Int. 312, 110331 (2020)
2. Rossler, A., et al.: Faceforensics++: Learning to detect manipulated facial images. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
3. Patel, K., et al.: Secure face unlock: Spoof detection on smartphones. IEEE Trans. Inf. Forensics Secur. 11(10), 2268–2283 (2016)
4. Akhtar, Z., Dassopu, D.: A comparative evaluation of local feature descriptors for Deepfakes detection. 2019 IEEE International Symposium on Technologies for Homeland Security (HST) (2019)
5. Guarnera, L., Giudice, O., Battiato, S.: Deepfake detection by analyzing convolutional traces. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2020)
6. Guo, Z., et al.: Fake face detection via adaptive manipulation traces extraction network. Comput. Vis. Image Underst. 204, 103170 (2021).
7. Chintha, A., et al.: Recurrent convolutional structures for audio spoof and video deepfake detection. IEEE J. Sel. Topics Signal Process. 14(5), 1024–1037 (2020)
8. Nguyen, H.H., et al.: Multi-task learning for detecting and segmenting manipulated facial images and videos. 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS) (2019)
9. Marra, F., et al.: Incremental learning for the detection and classification of GAN-generated images. 2019 IEEE International Workshop on Information Forensics and Security (WIFS) (2019)
10. Yu, N., Davis, L., Fritz, M.: Attributing fake images to GANs: Learning and analyzing GAN fingerprints. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
11. McCluskey, S., Allbright, M.: Detecting GAN-generated imagery using saturation cues. 2019 IEEE International Conference on Image Processing (ICIP) (2019)
12. Jafar, M.T., et al.: Digital forensics and analysis of Deepfake videos. 2020 11th International Conference on Information and Communication Systems (ICICS) (2020)
13. Karras, T., et al.: Progressive growing of GANs for improved quality, stability, and variation. 2018 International Conference on Learning Representations (ICLR) (2018)
14. He, K., et al.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
15. Google AI: Contributing Data to Deepfake Detection Research. (2019). https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html
16. Li, Y., et al.: Celeb-DF: A large-scale challenging dataset for Deepfake forensics. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
17. Antipov, G., Baccouche, M., Dugelay, J.: Face aging with conditional generative adversarial networks. 2017 IEEE International Conference on Image Processing (ICIP) (2017)
18. Tewari, A., et al.: Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. 2017 IEEE International Conference on Computer Vision (ICCV) (2017)
19. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
20. Kharbat, F., et al.: Image feature detectors for Deepfake video detection. 2019 IEEE/ACIS 16th International Conference on Computer Systems and Applications (AICCSA) (2019)
21. He, Z., et al.: Attagan: Facial attribute editing by only changing what you want. IEEE Trans. Image Process. 28(11), 5464–5478 (2019)
22. Cho, W., et al.: Image-to-image translation via group-wise deep whitening-and-coloring transformation. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

23. Choi, Y., et al.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
24. Karras, T., et al.: Progressive growing of GANs for improved quality, stability, and variation. arXiv:1710.10196 (2017)
25. Zhu, J., et al.: Unpaired image-to-image translation using cycle-consistent adversarial networks. 2017 IEEE International Conference on Computer Vision (ICCV) (2017)
26. Galvan, F., et al.: First quantization matrix estimation from double compressed JPEG images. IEEE Trans. Inf. Forensics Secur. 9(8), 1299–1310 (2014)
27. Giudice, O., et al.: 1-D DCT domain analysis for JPEG double compression detection. Image Analysis and Processing – ICIAP 2019 (2019)
28. Yang, X., Li, Y., Lyu, S.: Exposing deep fakes using inconsistent head poses. 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019)
29. Matern, F., Reiss, C., Stamminger, M.: Exploiting visual artifacts to expose Deepfakes and face manipulations. 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW) (2019)
30. Tulyakov, S., et al.: Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
31. Nguyen, H.M., Derakhshani, R.: Eyebrow recognition for identifying Deepfakes. 2020 International Conference of the Biometrics Special Interest Group (BIOSIG) (2020)
32. Pereira, T.d.F., et al.: Can face anti-spoofing countermeasures work in a real world scenario. 2013 International Conference on Biometrics (ICB) (2013)
33. Boulkenafet, Z., Komulainen, J., Hadid, A.: Face anti-spoofing using speed-up robust features and Fisher vector encoding. IEEE Signal Process. Lett. 24, 141–145 (2016)
34. Nguyen, H., Yamagishi, J., Echizen, I.: Use of a capsule network to detect fake images and videos. arXiv:1910.12467 (2019)
35. Yang, T., et al.: VTD-Net: Depth face forgery oriented video tampering detection based on convolutional neural network. 2020 39th Chinese Control Conference (CCC) (2020)
36. Zhou, P., et al.: Two-stream neural networks for tampered face detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017)
37. Afchar, D., et al.: Mesonet: A compact facial video forgery detection network. 2018 IEEE International Workshop on Information Forensics and Security (WIFS) (2018)
38. Rana, M.S., Sung, A.H.: DeepFakeStack: A Deep Ensemble-based Learning Technique for Deepfake Detection. 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), 70-75 (2020)
39. Chang, X., et al.: Deepfake face image detection based on improved VGG convolutional neural network. 2020 39th Chinese Control Conference (CCC) (2020)
40. Du, M., et al.: Towards generalizable Deepfake detection with locality-aware autoencoder. Proceedings of the 29th ACM International Conference on Information and Knowledge Management (2020)
41. Chen, P., et al.: Fspotter: Spotting face-swapped video by spatial and temporal clues. 2020 IEEE International Conference on Multimedia and Expo (ICME) (2020)
42. Nguyen, H.X., et al.: Learning spatio-temporal features to detect manipulated facial videos created by the deepfake techniques. Forensic Sci. Int.: Digital Invest. 36, 301108 (2021)
43. Mittal, A., et al.: No-reference image quality assessment in the spatial domain. IEEE Trans. Image Process. 21(12), 4695–4708 (2012)
44. Yang, T., et al.: Exposing deep fakes using inconsistent head poses. 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019)
45. Durall, R., et al.: Unmasking Deepfakes with Simple Features. arXiv:1911.00686 (2019)
46. Bae, S., Kim, M.: A novel image quality assessment with globally and locally consistent visual quality perception. IEEE Trans. Image Process. 25(5), 2392–2406 (2016)
47. Bae, S., Kim, M.: A novel DCT-based JND model for luminance adaptation effect in DCT frequency. IEEE Signal Process. Lett. 20(9), 893–896 (2013)
48. Korshunov, P., Marcel, S.: Deepfakes: A new threat to face recognition? Assessment and detection. arXiv:1812.08685 (2018)
49. Sanderson, C., Lovell, B.C.: Multi-region probabilistic histograms for robust and scalable identity inference. 2009 International Conference on Biometrics (2009)
50. 100,000 faces generated by Ai. (2018). https://generated.photos
51. Wang, R., et al.: Fakespotter: A simple yet robust baseline for spotting AI-synthesized fake faces. arXiv:1909.06122 (2020)
52. Larson, E.C., Chandler, D.M: Categorical image quality (CSIQ) database. (2009). http://vision.okstate.edu/csiq

How to cite this article: Yu, M., et al.: Deep forgery discriminator via image degradation analysis. IET Image Process. 1–16 (2021).
https://doi.org/10.1049/ipr2.12234