Hindawi
Security and Communication Networks
Volume 2022, Article ID 7460330, 12 pages
https://doi.org/10.1155/2022/7460330

Research Article

Detection of Face Morphing Attacks Based on Patch-Level Features and Lightweight Networks

Min Long,[1] **Xuan Zhao,**[1] **Le-Bing Zhang,**[2] **and Fei Peng**[3]

[1]*School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China*
[2]*College of Computer Science and Engineering, Huaihua University, Huaihua 418000, China*
[3]*College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China*

Correspondence should be addressed to Min Long; caslongm@aliyun.com

Received 5 November 2021; Accepted 3 February 2022; Published 11 March 2022

Academic Editor: Beijing Chen

Aiming at the problem of face morphing attack detection under mobile and resource-constrained conditions, a face morphing detection method based on patch-level features and lightweight networks is proposed. It utilizes the combination of three blocks' structures for learning. By outputting the probability of each bona fide or morphed face patches, the whole face features are integrated for recognition. Experimental results and analysis show this method can significantly improve the detection accuracy of face morphing attacks. Compared with the existing methods, this method has the characteristics of high computational efficiency and strong robustness. It has great application potential in enhancing the security of the face recognition system.

1. Introduction

As an important biometric technology, face recognition has been widely used for banks, hotels, transportation, and other areas for identity verification. After the human's face was chosen by International Civil Aviation Organization (ICAO) as a biometric feature in electronic machine-readable travel documents (eMRTD) for assisting identity verification, face recognition technology was gradually applied to Automatic Border Control (ABC) system [1]. Recently, a variety of attacks against face recognition systems appeared, among which face morphing attack posed a serious threat to the security of the existing face recognition systems (FRS) [2].

Face morphs include splicing morphs, complete morphs [3], and combined morphs [4]. Generally, a morphed face image is generated by two subjects. Complete morphs are a result of warping and blending the entire image. Splicing morphs use the convex hull representing a face and it is cut from the input images. Combined morphs use Poisson image editing to hide face and background. Then warp and blend operations are performed. For splicing morphs and combined morphs, the morphed image looks realistic because it is performed only in the face areas of two subjects, while for the complete morphs, the morphing operation is performed on the entire face, which usually leads to spurious shadows and tremendous visual inconsistencies in the hair region; therefore complete morphs are not appropriate for the morphing attack. Ferrara et al. showed the feasibility of face morphing attacks [2]. A criminal and an accomplice generate a morphed face image, and it is visually similar to the face images of criminal and accomplice, and it has both biological characteristics.

An example of a morphed facial image is shown in Figure 1. The pictures are from the publicly available face dataset Utrecht ECVP (http://omen.cs.unimagdeburg.de/disclaimer/index.php). If the morphed face image is used to apply for eMRTD, both the criminal and the accomplice can use this eMRTD to cross the boundary, as well as the FRS. Furthermore, the subsequent studies also proved the vulnerability of the FRS to face morphing attacks [5–7].

To countermeasure face morphing attacks, some methods have been proposed in recent years. Typical

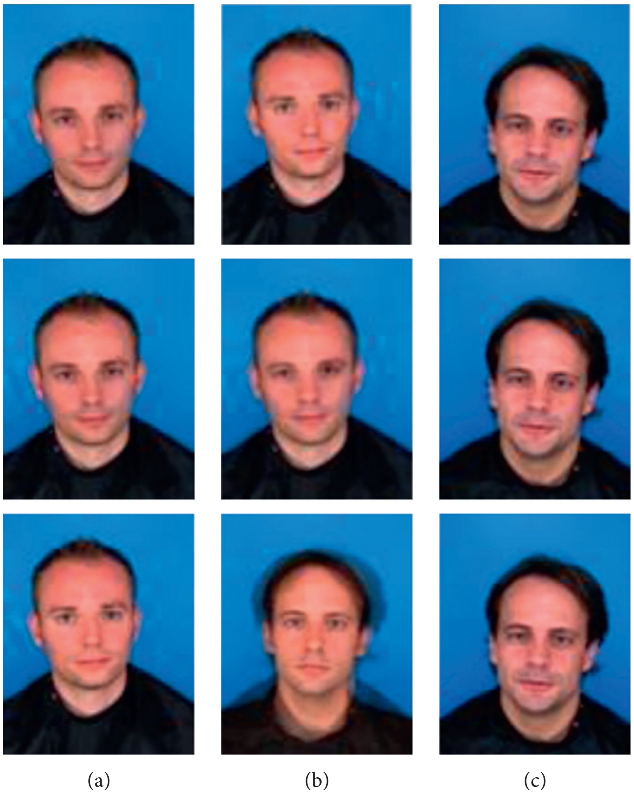

Figure 1: Example for a morphed face image (b) of subject 1(a) and subject 2(c). The middle face images in the first, second, and third row are called a splicing morphed image, combined morphed image, and complete morphed image, respectively. (a) Subject 1. (b) Morph. (c) Subject 2.

approaches include texture difference-based methods [8–17], image source feature-based methods [18–21], image quality-based methods [3, 22–24], and deep learning-based methods [25–37]. Among them, deep learning-based methods generally achieve good detection performance due to their strong feature description and learning ability, and deep learning can use nonlinear models to convert the original input data layer by layer into high-level abstract features. Therefore, methods based on deep learning have generally achieved good detection performance in the fields of image classification [38], image forensics [39, 40], face antispoofing [41, 42], and face morph attack detection. However, since deep learning models generally have large computing costs and high source requirements, the existing deep learning-based methods cannot be applied to resource-constrained application scenarios, such as mobile and embedded systems. To solve this problem, some lightweight networks such as SENet [43], SqueezeNet [44], MobileNetV2 [45], MobileNet-V3 [46], ShuffleNet [47], ShuffNetV2 [48], and Xception [49] have been successively proposed, and they are designed for common image recognition tasks.

To detect morphed face images, a novel method based on face patch-level features and lightweight networks is proposed in this paper. This method aims to make full use of the existing dataset and extract features by using lightweight convolution network. The main contributions are summarized as follows.

(i) A face patch-level feature learning approach is proposed and a two-level classification model is adopted. The use of patch-level features can expand the dataset and facilitate the extraction of identification information. The first-level classification uses the lightweight network to output the probability value of the face patch, and the second level of classification integrates the patch features of the whole face for discrimination. The two-level classification helps to improve the detection performance.

(ii) A lightweight network architecture for morphing attack detection is designed. The network adopts a combination of three-block structures, and a lighter ECA-Net attention mechanism module is added to the inverted residual structure, which can reduce the number of network parameters and maintain detection accuracy.

(iii) A new face morphed dataset named FERET_M is constructed based on the existing dataset. It includes 606 bona fide and 674 morphed face images. The experiments and cross-validation on three datasets demonstrate that the proposed method can achieve better face morphing attack detection performance compared with the existing methods.

The rest of the paper is organized as follows. The related work is introduced in Section 2. The proposed method is described in Section 3. Experimental results and discussion are provided in Section 4. Finally, some conclusions and future works are drawn in Section 5.

2. Related Work

Currently, the existing face morphing attack detection methods can be divided into four categories according to the detection features used, namely, texture difference-based methods, image source feature-based methods, image quality-based methods, and deep learning-based methods.

2.1. Texture Difference-Based Method. In the proposed secure scan design, the test authorization code is used to manage scan operation. Normal scanning operation can only be enabled by entering the correct test authorization code. Due to the different texture features between the morphed face image and the bona fide face image, some image descriptors, such as Local Binary Patterns (LBP) [8], Binarized Statistical Image Features (BSIF) [9], and Histogram of Oriented Gradient (HOG) [10], have been successively used for face morphing attack detection. In [11], the authors proposed to identify the authenticity of the face image using BSIF and Support Vector Machine (SVM). Scherhag et al. proposed a multialgorithm fusion approach to detect morphing attacks using LBP, BSIF, HOG, and other features [12]. Then, a morphed face detection scheme based on hybrid color features was put forward in [13]. A novel algorithm is presented to detect the morphed images by leveraging the collaborative representation of microtexture features and deriving the information from different color spaces [14]. In [15], the authors designed a morph attack detection algorithm that leveraged an undecimated 2D Discrete Wavelet Transform (2D-DWT) for identifying morphed face images. Another work [16] also employed 2D-DWT to highlight inconsistencies between a real and a morphed image. In [17], a texture difference-based method was the morphed face detection using facial landmarks. The current texture difference-based methods were all tested by a single image dataset, and they have poor adaptability.

2.2. Image Source Feature-Based Method. Motivated by image source identification, Zhang et al. proposed to detect morphed faces by using the Fourier spectrum of sensor pattern noise (FS-SPN) [18]. FS-SPN is extracted based on guided image estimation, and the statistics of the specific frequency difference between the morphed face image and the bona fide face image are obtained and then input to SVM for morphed face detection. In [19–21], Photo Response Non-Uniformity (PRNU) is implemented for morphed face detection. More specifically, the method in [20] is mainly focused on the variance of PRNU-based features across image cells, while the method in [19] is mainly based on the analysis of spectral variation of PRNU caused by the morphing process. In [21], it analyzes the spatial and spectral features from PRNU patterns across image cells. Differences features between bona fide and morphed images are estimated during a threshold-selection stage using the Dresden image database which is specifically built for PRNU analysis in digital image forensics. However, image source features-based methods are influenced by individual face morphing algorithms, and the detection accuracy needs to be improved.

2.3. Image Quality-Based Methods. Neubert et al. proposed to detect face morphing attacks based on image degradation [22]. By continuously analyzing the degradation process (e.g., JPEG compression) and manually generating several reference face images, the differences between the reference images and the original images are analyzed for face morphing attacks detection. Based on the observation that real face images comply with Benford's law, while morphed face images do not follow this law, Makrushin et al. proposed to detect face morphing forgeries based on Benford's law [23]. The detection is carried out by fitting a logarithmic curve to nine Benford features extracted from quantized DCT coefficients, which is the decomposition of JPEG compressed original and morphed face images. Besides, an automatic morphed face generation and detection of visually faultless facial morphs method was proposed in [3]. In [24], a demorphing approach is proposed. The live face image captured in real-time is subtracted from the morphed face image stored in the electronic document to obtain a demorphed image, and then the demorphed image is compared with the live face image by a specified threshold to determine whether it is the bona fide face. However, since the generation of the demorphed image depends on the prior knowledge of the morphing operation and parameters, the accuracy of demorphing is very limited.

2.4. Deep Learning-Based Methods. Presently, there are two types of approaches for detecting face morphing attacks using deep neural networks. One is to train a neural network from scratch, and the other is to use a pretrained neural network. In [25], a pretrained VGG19 [26] network was utilized to detect face morphing attacks. In [27], three convolutional neural networks, which were VGG19, GoogLeNet [28], and AlexNet [29], were utilized for face morphing attacks detection. After that, another morphing attacks detection method based on CNNs was put forward in [30]. The feature level fusion of the first fully connected layers of VGG19 and AlexNet is specifically fine-tuned. In [31], the research showed that the facial features calculated by the general face recognition system can be used for morphing detection, while in [32], it was demonstrated that the features extracted by the deep learning-based FRS are feasible for differential morphing attack detection. A face morphing attack using Generative Adversarial Networks [33] was proposed in [34]. From a new perspective, an innovative demorphing approach using a Generative Adversarial Network, which was named as FD-GAN, was proposed by Peng et al. [35]. The symmetric dual network architecture and two-level recovery losses were utilized to separate the identity feature of the morphing accomplices. Besides, a partial face manipulation-based morphing attack was proposed to compromise the uniqueness of face templates in [36]. The authors of [37] presented a novel differential morph detection framework by utilizing landmark and appearance disentanglement.

From the above analysis, the detection error rate of the general descriptors used for cross database face morphing attack detection is still high, and the generalization of the traditional algorithms is poor. The image source features-based methods are greatly affected by face morphing processes, and the detection performance of the image quality features-based methods is far from satisfactory. Although the deep learning-based face morphing detection methods can achieve good performance, it needs enough datasets support. Meanwhile, the datasets samples are limited in number of faces and attack types. If it is only effective for a single morphed attack method, the algorithm is easy to overfit. Furthermore, most of the existing methods utilize whole face as the input of face morphing detection, and the patch-level features of the face are not fully considered. Because some algorithms can only achieve good detection performance on a single dataset, it is necessary to improve the robustness and generalization ability across datasets. Therefore, a face morphing attack detection method based on patch-level features and lightweight networks is proposed in this paper to improve the generality of the algorithm.

3. The Proposed Method

In this section, we will introduce the details of the proposed method. The overall framework is illustrated in Figure 2, where face image preprocessing, lightweight network feature extraction, and two-level classification are the essential parts of the proposed method.

3.1. Face Image Preprocessing. The motivations of using patch-level features for image preprocessing are summarized as follows:

(i) Discriminative information of face morphing detection is presented in the whole facial area, while the background region is redundant and will interfere with the information. The whole face is divided into several fixed nonoverlapping regions and the patch-level features of the face are used as the input, which is helpful to extract more identification information from the network. Furthermore, patch-level feature analysis has achieved good detection performance in image source feature-based methods [19, 20].

(ii) The patch-level features can be used to perform data argumentation to datasets. For example, only 81 bona fide samples in FEI_M dataset are used for training. After a segmentation with a specific patch number N, the bona fide samples increase to 81 × N. Data augmentation can effectively increase the amount of training data and improve the generalization and robustness of the detection model based on deep learning.

(iii) Face morphing detection can be regarded as a binary classification problem. The use of patch-level features can effectively train the network model proposed in this paper. Meanwhile, BagNet [38] shows that a powerful image classification model can be developed by using patch-level features.

Firstly, the face detection algorithm is used to detect the input face image. Figure 3 shows the processing procedure of the face datasets. For all bona fide or morphed faces, we use OpenCV tool library to detect faces and obtain face images and face rectangles. The face rectangle lacks a certain background area, so the interference information is reduced. Then, for the detected face images, the face location algorithm in the Dlib (http://dlib.net/) toolkit is used to detect 68 key points of the face, and the coordinates of the key points are obtained. After obtaining the face region according to the key point coordinates and face rectangle, the face is divided into different patches by using the face patch-level algorithm. Specifically, the length N of the local feature to be designed is set to be 16 in this paper. Then, according to the input requirements of the lightweight network, the cropped face area is determined by the coordinates of the eye key points, so as to obtain cropped local patch with the same size 96 × 96 pixels [41]. The usual patch-based approach divides the whole face into several fixed overlapping regions. Each patch is used to train an independent subnetwork. In this paper, for each image, we train a lightweight CNN on random patches extracted from the faces. We randomly crop the face region according to the position of the eyes without scaling the area and maintain the original resolution, so as to maintain the discrimination ability.

3.2. Lightweight Network. In the face morphing attacks detection algorithms, the networks with good detection performance are generally very complex and have high requirements for hardware resources. The existing face morph detection networks have the problems of large parameters and weak generalization ability. Therefore, this paper focuses on the lightweight of network. One way is to compress the trained network to obtain a relatively small model. Another way is to design a small, representative lightweight network design for training.

Therefore inspired by FeatherNets [42], a lightweight network is proposed. The following is a detailed introduction to the lightweight network.

3.2.1. The Components of the Proposed Network. The proposed lightweight network is shown in Figure 4. It consists of Block_1, Block_2, and Block_3. In particular, Block_1 is reverse residual structure mentioned in MobileNetV2 [45], which is mainly composed of Depthwise Separable Convolutions including a 1×1 pointwise convolution, 3×3 depthwise convolution, 1×1 pointwise convolution, and residual concatenation. Given the kernel size is $n \times n$, k and k' are the input and output channels, respectively. The parameter of standard convolution is $n \times n \times k \times k$', and the Depthwise Separable Convolution is $n \times n + k \times k$'.

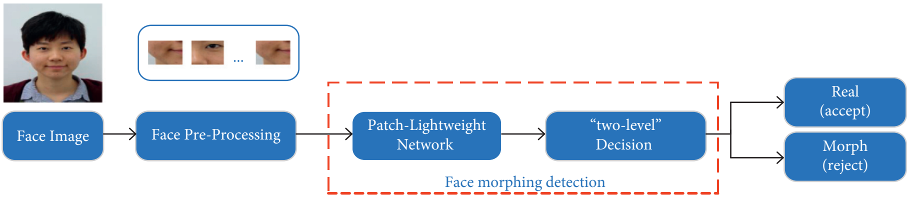

Figure 2: The overall framework of the proposed scheme.

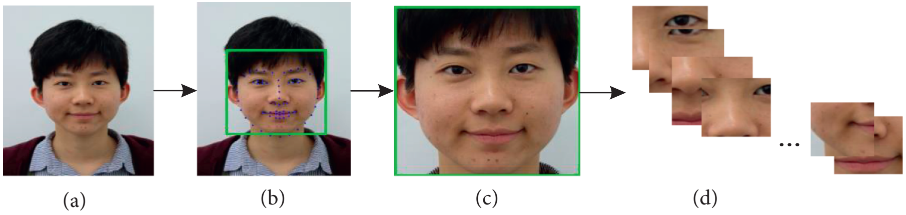

Figure 3: Preprocessing procedure of face dataset. (a) Input face image. (b) Face location and location of 68 key points. (c) Rectangular frame and face cropping. (d) Extract the face image patches; in this paper, the block length N is 16.

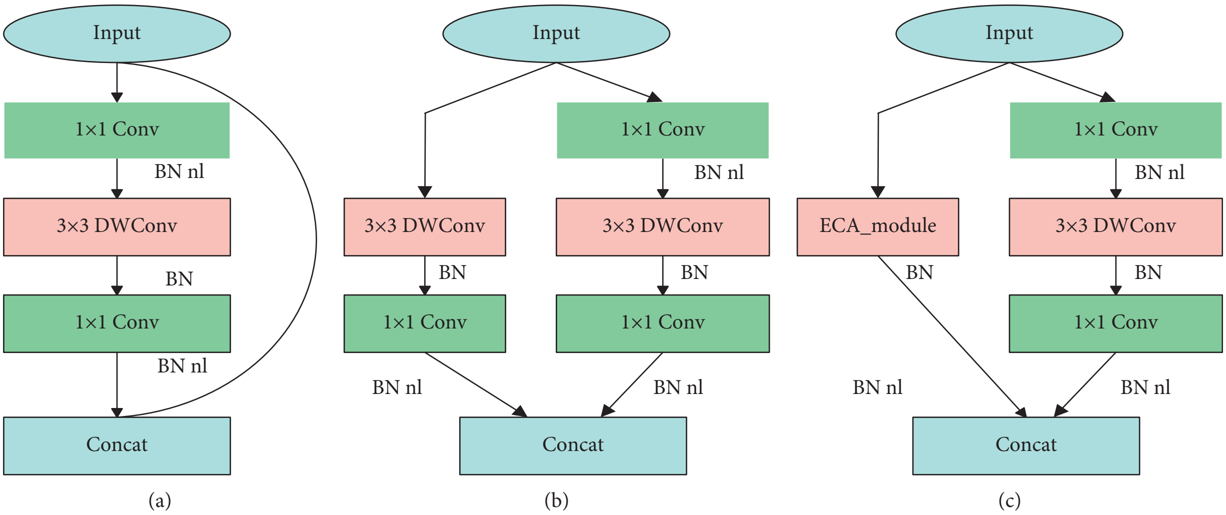

Figure 4: The main components of the proposed lightweight network. (a) Block_1. (b) Block_2. (c) Block_3.

Therefore, the deep separable convolution is used as the main convolution in this paper. Block_2 is composed of two parallel branches. The right branch is a Depthwise Separable Convolution, while the left branch includes a 3 × 3 depthwise convolution and a 1 × 1 pointwise convolution. The two branches are connected, and it can learn more features through the two branches. As illustrated in Xception [49], deepening the network width can improve the network performance. Thus, Stride = 2 is assigned in the depthwise convolution of Block_2, and it is also named as the downsampling module. Here, Block_3 is specifically designed in this paper. In order to solve the problem of losing important feature information in the reduced convolution calculation, the channel attention module helps the information flow in the network by learning the information to be emphasized or suppressed. As a result, it adds the ECA-Net module [50] to the reverse residual structure.

Attention mechanisms are essentially similar to human selective visual attention. Generally speaking, it pays more attention to key points and ignores other unimportant factors. For example, when we look at a picture of someone holding a table and a tree, when we look at it with our eyes, people will become the focus of what we see. But for machine detection, all objects are of equal importance.

As shown in Figure 3, although most face morph attack detection algorithms perform face cropping preprocessing steps, the importance of the four corners in the face and the image is different, because in the existing face fusion algorithms, only the face part is partially morphed, and the edge information such as hair is redundant information. Therefore, we believe that the attention mechanism can improve the detection effect of face morph attacks, allows the network to adaptively learn more important features, and is suitable for face morph attack detection, so we introduce the attention mechanism into the lightweight network.

ECA-Net is an improvement of SENet. It does not reduce the channel dimension, and it generates channel weights by performing 1D convolution with size k. Here, k is adaptively determined by a function of channel dimension C, and it is a more lightweight channel attention mechanism, which has been successfully employed to improve the accuracy of CNN classification models. Therefore, the Block_3 can significantly reduce the number of parameters while maintaining performance gains. Attention-based layers can help to learn feature regions to detect bona fide and morphed faces. Thus, Block_1, Block_2, and Block_3 constitute the main components of the proposed lightweight network.

3.2.2. The Network Architecture. The details of the proposed lightweight network architecture are listed in Table 1. At the beginning of the network, the input dimension of the patch-level features of the face is $96 \times 96 \times 3$, and the regular convolution Conv2d (stride = 2) is used to extract and retain more features. The number of convolution kernels is 16, and the size of the convolution kernel is 3×3. Then, the downsampling strategy of the Block_2 is used to reduce the input to $24 \times 24 \times 24$, which allows rapid reduction of the feature map size. It greatly reduces the calculation workload and parameters and can achieve faster speed. Block_1 module is the simplest one and it requires the least parameters among the three modules, so Block_1 is often used in the network. Block_3 is applied after Block_2 to minimize the loss of feature information due to downsampling of the channel attention mechanism. Block_3 can be used for all input channels above 40 to enhance the information interaction between deep network channels. Additionally, an ECA-Net module is also inserted after the second normal convolution layer to improve the accuracy of the model with appropriate parameters. The subsequent pool layer and the regular convolution layer are the final processes. The use of pool layers simplifies the model complexity and reduces computation and memory consumption. Finally, a fully connected layer acts as a classifier for the whole network.

In the deeper layers of the network, *H*-swish activation function [46] is used, and it is defined by

$$h - \text{swish}[x] = x\frac{\text{RELU6}(x+3)}{6}. \tag{1}$$

The use of *H*-swish can effectively improve the accuracy of deep networks. Therefore, due to the computation complexity, *H*-swish activation function is replaced with Rectified Linear Unit and Sigmoid only when the input channel of the proposed network is not less than 40. That is, in the proposed lightweight network, *h*-swish() is used when nl = HS, and the ReLU() activation function is used when nl = RE.

3.3. Classification. The input faces will be divided into face patches with fixed size. For each patch, it will be input to the proposed lightweight network and converted into probability value of each face patch. The first-level judgment is performed on each patch to determine whether it is a bona fide one or a morphed one, and then the second-level judgment is performed to make a final decision. If the number of face patches determined as a bona fide one is greater than that of face patches determined as morphed ones, this input face is determined as a bona fide face; otherwise, the input face is determined as a morphed face.

4. Experiment

In this section, we evaluate the effectiveness of the proposed method. Since complete morphs will lead to apparent artifacts, they are not suitable for a morphing attack unless they have undergone a manual retouch. Thus, the experiments are only carried out for splicing morphs and combined morphs.

4.1. Datasets. In the experiments, three face morph datasets FEI_M [18], HNU_FM [35], and a self-built FERET_M dataset are used.

Each of the three datasets is divided into a noncrossed training set, a validation set, and a test set, respectively. The training set is used to train the network model, the validation set is applied to adjust the hyperparameters, and the test set is used to verify performance of the model.

The FEI_M dataset is based on the public FEI face datasets, constructed according to the face morph algorithm in [3] with a fusion factor $\alpha = 0.5$. It consists of 200 subjects with 100 females and 100 males. FERET_M dataset is a new dataset built by us, and it is derived from the FERET dataset [51], which is a public face image dataset containing male and female images taken under different acquisition conditions with varying poses, facial expressions, and ages. The subjects suitable for face morph are manually selected from the color FERET dataset to build the new face morphing dataset with the standard morphing algorithm [4], where the pixel fusion factor α is fixed as 0.5. It consists of 303 subjects with 98 females and 205 males. The details of the FEI_M dataset and FERET_M dataset are listed in Table 2.

HNU_FM dataset is composed of four subdatasets FaceMDB1, FaceMDB2, FaceMDB3, and FaceMDB4, which are generated according to different pixel fusion factors α and location fusion factors β. There are 63 subjects, including 27 females and 36 males. A detailed description of the HNU_FM dataset is provided in Table 3. For FaceMDB1 dataset, the pixel fusion factors α and the location fusion factors β are both 0.5. For the FaceMDB2 dataset, the pixel fusion factor α is fixed to 0.5 and the location fusion factor β varies between 0.1 and 0.9. For the FaceMDB3 dataset, the pixel fusion factor α varies between 0.1 and 0.9, and the location fusion factor β is fixed to 0.5. For FaceMDB4, the pixel fusion factor α and the location fusion factor β both vary between 0.1 and 0.9.

4.2. Experimental Setup and Evaluation Criteria. Instead of the input of the traditional convolutional neural networks, the input size is 96×96 and Stochastic Gradient Descent (SGD) is utilized as the optimizer. The learning rate starts from 0.1 and it is adjusted using cosine annealing [52] with a cross-entropy loss function. Weight decay and momentum are set to 0.0001 and 0.9, respectively.

Table 1: The details of the proposed lightweight network.

Input	Operator	Exp size	#out	NL
96 × 96 × 3	Conv2d	—	16	RE
48 × 48 × 16	Block_2	24	24	RE
24 × 24 × 24	Block_1	24	24	RE
24 × 24 × 24	Block_3	24	24	RE
24 × 24 × 24	Block_2	96	40	RE
12 × 12 × 40	4 × Block_1	160	40	RE
12 × 12 × 40	Block_3	160	40	RE
12 × 12 × 40	Block_2	120	48	HS
6 × 6 × 48	Block_3	144	48	HS
6 × 6 × 48	Block_3	240	96	HS
6 × 6 × 96	Block_3	480	96	HS
6 × 6 × 96	Conv2d	—	256	HS
6 × 6 × 256	Pool,6 × 6	—	—	—
1 × 1 × 256	Conv2d	—	1024	HS
1 × 1 × 1024	Conv2d	—	2	

Table 2: FEI_M dataset and FERET_M dataset.

Dataset	Subset	#Bona fide	#Splicing morphs
FEI_M	Training set	81	6480
	Validation set	20	380
	Testing set	99	3702
FERET_M	Training set	459	516
	Validation set	46	54
	Testing set	101	104

Table 3: HNU_FM dataset.

Dataset	Subset	#Bona fide	#Splicing morphs
FaceMDB1	Training set	1121	1121
	Validation set	564	330
	Testing set	566	377
FaceMDB2	Training set	1121	1125
	Validation set	564	567
	Testing set	566	567
FaceMDB3	Training set	1121	1125
	Validation set	564	567
	Testing set	566	567
FaceMDB4	Training set	1121	1134
	Validation set	564	567
	Testing set	566	567

The standardized ISO metrics Attack Presentation Classification Error Rate (APCER), Bona Fide Presentation Error Rate (BPCER), and Average Classification Error Rate (ACER) are used to evaluate the overall detection performance. APCER, BPCE, and ACER are, respectively, defined as follows:

APCER: proportion of attack presentations incorrectly classified as bona fide presentations in a specific scenario.

BPCER: proportion of bona fide presentations incorrectly classified as presentation attacks in a specific scenario.

ACER: the average of the sum of APCER and BPCER.

4.3. Experimental Results and Discussion. As complete morphs can lead to obvious artifacts unless they are manually trimmed, they are not suitable for morphing attacks. Thus, the experiments are only performed to splicing morphs and combined morphs.

Similarly, the general steps of automatically generating the morphed facial image are as follows [35]. (a) Locate the key points of the morphed facial image, (b) warp the morphed facial image through Delaunay triangulation, and (c) blend the morphed facial image through pixel fusion.

For nondeep learning methods, public codes are often used for preprocessing and experiments [18]. According to the face deformation algorithm, the identification information of face deformation attack detection often exists in the face, and the background area is redundant, which is the interference information of face fusion attack detection. Therefore, in the preprocessing stage for the method based on deep learning, the face detector of Dlib library (http://dlib.net/) is used to detect the face of the input image, and the detected face is aligned with the face. It combines with clipping operations to ensure that the algorithm is only used for face regions. In this paper we add patch-level processing to face preprocessing.

4.3.1. Single Dataset Test. To evaluate the detection performance of the proposed method, a comparative analysis is performed on the HNU_FM, FEI_M, and FERET_M datasets with 7 methods.

The experimental results on the HNU_FM dataset are listed in Table 4. It can be found that FS-SPN [18] can achieve the best APCER, while ResNet50 [53], MobileNetV2 [45], and the proposed method can obtain the best BPCER from FaceMDB1 to FaceMDB2 datasets (all are 0). MobileNetV2 [45] achieved the best ACER on FaceMDB2, FS-SPN [18] achieved the best ACER on FaceMDB4, and the best detection results on FaceMDB1 and FaceMDB3 belonged to the proposed method. Meanwhile, comparing the results on FaceMDB1 with those on FaceMDB3, the overall detection performance on FaceMDB3 is better than that on FaceMDB1. This is because the pixel fusion factor α of the FaceMDB3 dataset is not 0.5, and they are less likely to pass the face recognition system. In addition, the experimental results also demonstrate that the location fusion factor β has little impact on the detection.

The experimental results on FEI_M dataset are listed in Table 5. It can be found that FS-SPN [18] can achieve the best BPCER and ACER, while for the proposed method, it can obtain the best BPCER, and its ACER ranks the second after FS-SPN [18]. Since the number of bona fide on the FEI_M dataset is much smaller than the number of morphed face images, the samples of the dataset are unbalanced. Therefore, in general, BPCER is higher than APCER.

The experimental results on FERET_M dataset are listed in Table 6. It can be found the proposed method can achieve the best BPCER and ACER.

Considering the results on three image datasets, FS-SPN [18] can obtain the best performance among the nondeep learning methods. However, the detection performance of the deep learning-based methods is better than that of the nondeep learning methods on the whole. As the data volume is important to the performance of deep learning-based methods, the results on large sample image dataset are more convincing than those on small sample image dataset. The results of the proposed method on HNU_FM and FERET_M outperform most of the existing methods, which indicates the good performance of the proposed method on single dataset test.

4.3.2. Across Dataset Test. To evaluate the generality of the proposed method, comparative experiments are conducted to seven methods. In the across dataset test, one of three image datasets FEI_M, FaceMDB1, and FERET_M is used as training dataset, and the other two image datasets are used as test datasets; the results are listed in Tables 7–9, respectively.

As shown in Table 7, when the detection methods are trained by FEI_M and tested by FaceMDB1, it can be found that FS-SPN [18] can achieve the best APCER, while the proposed method can obtain the best BPCER. When the detection methods are trained by FEI_M and tested by FERET_M, FS-SPN [18] can achieve the best APCER, while the proposed method can obtain the best BPCER and ACER.

Although the APCER of FS-SPN [18] is low, the BPCER is very high. The experimental results based on nondeep learning methods further illustrate that the generalization performance of traditional texture feature methods is poor. From the experimental results of ResNet [53] and MobileNetV2 [45], the method based on deep learning has good generalization ability.

Because of the imbalance of the FEI_M dataset, larger-scale samples may lead to overfitting and it is also easy to reduce the generalization ability of the model. When it is trained on the FEI_M dataset, the detection accuracy of most samples is high, and that of a few samples is low. Due to the use of a large number of datasets for enhancement, the method proposed in this paper suppresses this problem to a certain extent. Therefore, from the overall experimental results, a better ACER can be achieved.

As seen in Table 8, when the detection methods are trained by FaceMDB1 and tested by FEI_M dataset, it can be found that FS-SPN [18] can achieve the best APCER, MobileNetV2 [45], and the proposed method can obtain the best BPCER. MobileNetV2 [45] can achieve the best ACER. When the detection methods are trained by FaceMDB1 and tested by FERET_M, the proposed method can obtain the best BPCER and the best ACER.

From the overall experimental results in Table 8, the generalization performance of HOG [9], BSIF [10], and FS-SPN [18] tested on the FEI_M and FERET_M datasets is also very poor. The ACER results of the three methods are about 50%, which is close to random guess. The detection performance of the method based on deep learning is also better than the above three methods.

As seen in Table 9, when the detection methods are trained by FERET_M and tested by FEI_M, it can be found that BSIF [10] can achieve the best APCER, the proposed method can obtain the best BPCER, and ResNet50 [53] can achieve the best ACER. When the detection methods are trained by FERETM and tested by FaceMDB1, BSIF [10] can achieve the best APCER and the proposed method can obtain the best ACER. MobileNetV2 [45] and the proposed method can obtain the best BPCER.

From the experimental results of Table 8 and Table 9, the difference between the datasets has a great impact on the detection results. When tested on the FaceMDB1 dataset (trained on the FERET_M dataset), the detection performance of all algorithms is much lower than that on the FERET_M dataset. This result is attributed to the difference between the bond fide of the two datasets.

In summary, when the detection methods are tested on FERET_M and FEI_M datasets, the overall detection performance is very poor due to the small number of the validation sets in the two datasets. Thus, it can be found that the number of positive and negative samples in the dataset has significant impact on the generalization performance of the face morphing attack detection, especially for methods based on deep learning. It can be also found that the model

Table 4: Performance comparison of different methods on HNU_FM dataset (%).

Method	FaceMDB1			FaceMDB2			FaceMDB3			FaceMDB4		
	APCER	BPCER	ACER	APCER	BPCER	ACER	APCER	BPCER	ACER	APCER	BPCER	ACER
HOG [9]	41.33	8.33	24.83	48.68	0.17	24.43	45.86	0.88	23.37	48.68	0.35	24.52
BSIF [10]	28.64	16.96	22.80	35.44	13.25	24.35	30.15	30.56	30.36	31.39	37.27	34.33
FS-SPN [18]	**0**	1.86	0.93	**0**	2.12	1.06	**0**	2.30	1.15	**1.40**	0.18	**0.79**
VGG19 [26]	27.32	1.22	14.27	23.28	0.88	12.08	37.21	0.35	18.78	36.86	1.06	18.96
ResNet50 [53]	3.98	**0**	1.95	2.12	**0**	1.06	2.47	**0**	1.24	3.35	0	1.68
SqueezeNet1_1 [39]	7.96	**0**	3.98	5.29	0.88	3.09	2.29	0.35	1.32	5.47	5.30	5.39
MobileNet V2 [40]	4.51	**0**	2.26	0.70	**0**	**0.35**	11.99	1.06	6.53	6.02	**0**	3.01
Proposed	1.33	**0**	**0.67**	1.76	**0**	0.88	1.23	**0**	**0.62**	1.59	**0**	0.80

Table 5: Performance comparison of different methods on FEI_M dataset (%).

Method	APCER	BPCER	ACER
HOG [9]	9.33	11.01	10.17
BSIF [10]	3.13	13.13	8.13
FS-SPN [18]	1.12	**0**	**0.56**
VGG19 [26]	1.94	4.04	2.99
ResNet50 [53]	**0**	3.03	1.56
SqueezeNet1_1 [44]	0.27	5.05	2.66
MobileNet V2 [45]	**0**	3.03	1.52
Proposed	1.98	**0**	0.99

Table 6: Performance comparison of different methods on FERET_M dataset (%).

Method	APCER	BPCER	ACER
HOG [9]	38.46	33.66	36.06
BSIF [10]	16.35	29.70	23.03
FS-SPN [18]	19.80	15.38	17.59
VGG19 [26]	23.07	17.68	20.38
ResNet50 [53]	12.73	7.07	11.80
SqueezeNet1_1 [44]	14.59	10.42	12.51
MobileNetV2 [45]	9.91	3.89	6.9
Proposed	9.91	**1.98**	**5.95**

Table 7: Comparative results of cross dataset test (trained by FEI_M) (%).

Method	Trained by FEI_M					
	Tested by FaceMDB1			Tested by FERET_M		
	APCER	BPCER	ACER	APCER	BPCER	ACER
HOG [9]	15.38	72.61	44.00	21.78	72.12	46.95
BSIF [10]	14.42	66.61	40.52	2.97	89.11	46.04
FS-SPN [18]	**5.30**	68.90	37.10	**1.98**	89.11	45.55
VGG19 [26]	18.04	57.07	37.55	25.96	69.31	47.64
ResNet50 [53]	7.96	42.40	25.18	8.65	50.50	29.58
SqueezeNet1_1 [44]	10.61	44.17	27.39	14.42	54.46	34.44
MobileNet V2 [45]	7.96	35.51	21.74	12.50	35.64	24.07
Proposed	21.22	**0**	**10.61**	27.72	**0**	**13.86**

Table 8: Comparative results of cross dataset test (trained by FaceMDB1) (%).

Method	Trained by FaceMDB1					
	Tested by FEI_M			Tested by FERET_M		
	APCER	BPCER	ACER	APCER	BPCER	ACER
HOG [9]	28.40	64.00	46.20	77.88	34.65	56.27
BSIF [10]	29.92	55.56	42.74	72.12	27.72	49.92
FS-SPN [18]	**0.63**	92.93	46.78	77.88	3.96	40.92
VGG19 [26]	25.77	45.45	35.61	52.88	39.60	46.24
ResNet50 [53]	15.46	30.30	22.88	37.50	25.74	31.62
SqueezeNet1_1 [44]	17.53	34.34	25.94	40.38	29.70	35.04
MobileNet V2 [45]	8.14	**0**	4.07	12.73	7.07	9.9
Proposed	9.07	**0**	4.54	**8.65**	**3.89**	**6.27**

Table 9: Comparative results of cross dataset test (trained by FERET_M) (%).

Method	Trained by FERET_M					
	Tested by FEI_M			Tested by FaceMDB1		
	APCER	BPCER	ACER	APCER	BPCER	ACER
HOG [9]	7.60	69.70	38.65	44.03	21.20	32.62
BSIF [10]	0.14	70.78	35.46	**3.71**	47.09	25.40
FS-SPN [18]	8.79	70.78	39.79	50.66	20.49	35.58
VGG19 [26]	19.57	50.50	34.81	26.79	27.56	27.18
ResNet50 [53]	9.27	35.35	22.31	14.85	16.43	15.64
SqueezeNet1_1 [44]	12.95	38.38	25.67	15.92	18.55	17.24
MobileNet V2 [45]	25.30	20.20	22.75	11.52	**8.06**	9.79
Proposed	73.23	2.02	37.63	7.96	**8.06**	**8.01**

Table 10: Comparison of Params and FLOPS.

Method	Params	FLOPS
VGG19 [26]	143.6M	19.67G
ResNet50 [53]	23.51M	3800M
SqueezeNet1_1 [44]	1.24M	357.30M
MobileNetV2 [45]	2.23M	306.17M
Proposed	0.57M	25.10M

strategy trained on the FERET_M dataset is better than the other two datasets. This is because the face images in FERET_M dataset contain more changes. The above results show that the generalization ability of deep learning-based methods outperforms nondeep learning methods, and the proposed method can achieve the better overall performance. However, it can be seen from the results that the proposed method is more biased towards bona fides, so the generalization performance still needs to be improved.

4.4. Analysis of Network Model Parameters. To further verify whether the proposed network meets the requirements of a lightweight network, the parameter sizes of different networks are compared and analyzed. The metrics for evaluating the size of network parameters include the parameter size of the network model generated by Pytorch (Parameter). In addition, the number of floating-point operations per second (FLOPS) is also presented. The Params and FLOPS for different models are listed in Table 10. It can be seen that the parameter size of the proposed method is only 0.57M and its FLOPS is only 25.10M, which are significantly lower than those of the other networks.

5. Conclusion and Future Works

In this paper, a face morphing attack detection method based on patch-level features and lightweight networks is proposed. By using patch-level features, not only is the dataset increased, but the ability of face representation is improved. Meanwhile, the proposed three-block combined lightweight networks help to reduce the number of network parameters. The experiments on 3 datasets and comparative analysis with some state-of-the-art methods show that the proposed method can achieve better detection performance with less network model parameters and operations. Furthermore, the cross datasets test also illustrates the good robustness ability of the proposed method. However, since the dataset in this paper is not comprehensive enough and the postprocessing of morphing attack using digital images is not considered, the performance of the method needs further analysis and validation. In the evaluation criteria, the threshold relationship in the selected indicators is not considered. These are the places we will consider in the future. Our future work will also focus on how to enhance the robustness and generality of lightweight network against morphing attacks.

Data Availability

The data used to support the findings of the study are available from the corresponding author (caslongm@aliyun.com) upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 62072055, U1936115, and 92067104) and the Changsha Key Research and Development Program(no. kq2004004).

References

[1] ICAO, *Biometric Deployment of Machine Readable Travel Documents*, TAG MRTDINTWG, Canada, 2004.

[2] M. Ferrara, A. Franco, and D. Maltoni, “The magic passport,” in *Proceedings of the IJCB*, pp. 1–7, Clearwater, FL, USA, 29 Sept.-2 Oct. 2014.

[3] A. Makrushin, T. Neubert, and J. Dittmann, “Automatic generation and detection of visually faultless facial morphs,” in *Proceedings of the 12th Int.Joint Conf. Comput. Vis. Imag. Comput. Graph. Theory Appl.*, pp. 39–50, 2017.

[4] T. Neubert, A. Makrushin, M. Hildebrandt, C. Kraetzer, and J. Dittmann, “Extended StirTrace benchmarking of biometric and forensic qualities of morphed face images,” *IET Biometrics*, vol. 7, no. 4, pp. 325–332, 2018.

[5] D. J. Robertson, R. S. S. Kramer, and A. M. Burton, “Fraudulent id using face morphs: experiments on human and automatic recognition,” *Plos One*, vol. 12, no. 3, p. e0173319, 2017.

[6] M. Gomez-Barrero, C. Rathgeb, U. Scherhag, and C. Busch, "Is your biometric system robust to morphing attacks?" in *Proceedings of the IWBF*, pp. 1–6, Coventry, UK, 4-5 April 2017.

[7] C. Rathgeb, K. Pöppelmann, and C. Busch, "Face morphing attacks: a threat to eLearning?" in *Proceedings of the EDUCON*, pp. 1149–1154, Vienna, Austria, 21-23 April 2021.

[8] 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.

[9] C. Shu, X. Ding, and C. Fang, "Histogram of the oriented gradient for face recognition," *Tsinghua Science and Technology*, vol. 16, no. 2, pp. 216–224, 2011.

[10] J. Kannala and E. Rahtu, "BSIF: Binarized statistical image features," in *Proceedings of the ICPR*, pp. 1363–1366, Tsukuba, Japan, 11-15 Nov. 2012.

[11] R. Raghavendra, K. B. Raja, and C. Busch, "Detecting morphed face images," in *Proceedings of the BTAS*, pp. 1–7, Niagara Falls, NY, USA, 6-9 Sept. 2016.

[12] U. Scherhag, C. Rathgeb, and C. Busch, "Morph detection from single face image: a multi-algorithm fusion approach," in *Proceedings of the ICBEA*, pp. 6–12, Amsterdam, Netherlands, May 16-18, 2018.

[13] R. Ramachandra, S. Venkatesh, K. B. Raja, and C. Busch, "Towards making morphing attack detection robust using hybrid scale-space colour texture features," in *Proceedings of the ISBA*, pp. 1–8, Hyderabad, India, 22-24 Jan. 2019.

[14] R. Ramachandra, K. B. Raja, S. Venkatesh, and C. Busch, "Face morphing versus face averaging: vulnerability and detection," in *Proceedings of the IEEE Int. Joint Conf. Biometr. (IJCB)*, pp. 555–563, Denver, CO, USA, 1-4 Oct. 2017.

[15] B. Chaudhary, P. Aghdaie, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Differential morph face detection using discriminative wavelet sub-bands," in *Proceedings of the CVPRW*, pp. 1425–1434, Nashville, TN, USA, 19-25 June 2021.

[16] P. Aghdaie, B. Chaudhary, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Detection of morphed face images using discriminative wavelet sub-bands," in *Proceedings of the IWBF*, pp. 1–6, Rome, Italy, 6-7 May 2021.

[17] U. Scherhag, D. Budhrani, M. Gomez-Barrero, and C. Busch, "Detecting morphed face images using facial landmarks," in *Proceedings of the ICVISP*, Cherbourg, France, July 2018.

[18] L. B. Zhang, F. Peng, and M. Long, "Face morphing detection using Fourier spectrum of sensor pattern noise," in *Proceedings of the ICME*, pp. 1–6, San Diego, CA, USA, 23-27 July 2018.

[19] L. Debiasi, U. Scherhag, C. Rathgeb, A. Uhl, and C. Busch, "PRNU based detection of morphed face images," in *Proceedings of the IWBF*, pp. 1–7, Sassari, Italy, 7-8 June 2018.

[20] L. Debiasi, C. Rathgeb, U. Scherhag, A. Uhl, and C. Busch, "PRNU variance analysis for morphed face image detection," in *Proceedings of the BTAS*, pp. 1–9, Redondo Beach, CA, USA, 22-25 Oct. 2018.

[21] U. Scherhag, L. Debiasi, C. Rathgeb, C. Busch, and A. Uhl, "Detection of face morphing attacks based on PRNU analysis," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 4, pp. 302–317, 2019.

[22] T. Neubert, "Face morphing detection: an approach based on image degradation analysis," in *Proceedings of the IWDW*, pp. 93–106, Magdeburg, Germany, August 2017.

[23] A. Makrushin, C. Kraetzer, T. Neubert, and J. Dittmann, "Generalized Benford's law for blind detection of morphed face images," in *Proceedings of the 6th ACM Workshop Inf. Hiding Multimedia Security (IH & MMSec)*, pp. 49–54, Innsbruck, Austria, June 20-22, 2018.

[24] M. Ferrara, A. Franco, and D. Maltoni, "Face demorphing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 1008–1017, 2018.

[25] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, "Accurate and robust neural networks for security related applications exampled by face morphing attacks," in *Proceedings of the CVPR*, pp. 1–16, Salt Lake City, UT, USA, June 2018.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the Comput. Vis. Pattern Recognit*, ICLR, San Diego, CA, USA, May 2015.

[27] C. Seibold, W. Samek, A. Hilsmann, and P. Eisert, "Detection of face morphing attacks by deep learning, digital forensics and watermarking," in *Proceedings of the IWDW*, pp. 107–120, Magdeburg, Germany, August 2017.

[28] C. Szegedy, W. Liu, Y. Q. Jia et al., "Going deeper with convolutions," in *Proceedings of the CVPR*, pp. 1–9, Boston, MA, 7-12 June 2015.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[30] R. Raghavendra, K. B. Raja, S. V enkatesh, and C. Busch, "Transferable deep-CNN features for detecting digital and print-scanned morphed face images," in *Proceedings of the CVPRW*, pp. 1822–1830, Honolulu, HI, USA, 21-26 July 2017.

[31] L. Wandzik, G. Kaeding, and R. V. . Garcia, "Morphing detection using a general-purpose face recognition system," in *Proceedings of the EUSIPCO*, pp. 1012–1016, Rome, Italy, 3-7 Sept. 2018.

[32] U. Scherhag, C. Rathgeb, J. Merkle, and C. Busch, "Deep face representations for differential morphing attack detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3625–3639, 2020.

[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," in *Proceedings of the NIPS*, pp. 2672–2680, Montreal, Quebec, Canada, December 2014.

[34] N. Damer, A. M. Saladie, A. Braun, and A. Kuijper, "MorGAN: recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network," in *Proceedings of the BTAS*, pp. 1–10, Redondo Beach, CA, USA, 22-25 Oct. 2018.

[35] F. Peng, L.-B. Zhang, and M. Long, "FD-GAN: face de-morphing generative adversarial network for restoring accomplice's facial image," *IEEE Access*, vol. 7, pp. 75122–75131, 2019.

[36] L. Qin, F. Peng, S. Venkatesh, R. Ramachandra, M. Long, and C. Busch, "Low visual distortion and robust morphing attacks based on partial face image manipulation," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 3, no. 1, pp. 72–88, 2021.

[37] S. Soleymani, A. Dabouei, F. Taherkhani, J. Dawson, and N. M. Nasrabadi, "Mutual information maximization on disentangled representations for differential morph detection," in *Proceedings of the WACV*, pp. 1730–1740, Waikoloa, HI, USA, 3-8 Jan. 2021.

[38] W. Brendel and M. Bethge, "Approximating CNNs with Bag-of-local-Features models works surprisingly well on ImageNet," in *Proceedings of the ICLR*, New Orleans, LA, USA, May 2019.

[39] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, and Y.-Q. Shi, "A serial image copy-move forgery localization scheme with source/target distinguishment," *IEEE Transactions on Multimedia*, vol. 23, pp. 3506–3517, 2021.

[40] B. Chen, X. Liu, Y. Zheng, G. Zhao, and Y.-Q. Shi, "A robust GAN-generated face detection method based on dual-color spaces and an improved Xception," *IEEE Transactions on Circuits and Systems for Video Technology*, p. 1, 2021.

[41] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu, "Face anti-spoofing using patch and depth-based CNNs," in *Proceedings of the IJCB*, pp. 319–328, Denver, CO, USA, 1-4 Oct. 2017.

[42] P. Zhang, F. Zou, Z. Wu et al., "FeatherNets: convolutional neural networks as light as feather for face anti-spoofing," in *Proceedings of the CVPRW*, pp. 1574–1583, Long Beach, CA, USA, 16-17 June 2019.

[43] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.

[44] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: alexnet-level accuracy with 50x fewer parameters and 1mb model size," in *Proceedings of the ICLR*, Toulon, France, April 2016.

[45] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: inverted residuals and linear bottlenecks," in *Proceedings of the CVPR*, pp. 4510–4520, Salt Lake City, UT, USA, 18-23 June 2018.

[46] A. G. Howard, M. Sandler, G. Chu et al., "Searching for MobileNetV3," in *Proceedings of the ICCV*, pp. 1314–1324, Seoul, Korea, 27 Oct.-2 Nov.

[47] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shuffle-net: an extremely efficient convolutional neural network for mobile devices," in *Proceedings of the CVPR*, Salt Lake City, USA, June 2018.

[48] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: practical guidelines for efficient cnn architecture design," in *Proceedings of the ECCV*, pp. 122–138, Munich, Germany, September 2018.

[49] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the CVPR*, pp. 1800–1807, Honolulu, HI, USA, 21-26 July 2017.

[50] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-net: efficient channel attention for deep convolutional neural networks," in *Proceedings of the CVPR*, pp. 11531–11539, Seattle, WA, USA, 13-19 June 2020.

[51] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image and Vision Computing*, vol. 16, no. 5, pp. 295–306, 1998.

[52] G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft, and K. Q. Weinberger, "Snapshot ensembles: train 1, get M for free," in *Proceedings of the ICLR*, Toulon, France, April 2017.

[53] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the CVPR*, pp. 770–778, Las Vegas, NV, USA, June 2016.