Fake Face Detection via Adaptive Residuals Extraction Network

Zhiqing Guo[1] Gaobo Yang[1] Jiyou Chen[1] Xingming Sun[2]
[1]Hunan University [2]Nanjing University of Information Science and Technology
{guozhiqing, yanggaobo}@hnu.edu.cn cjyhn0302@gmail.com sunnudt@163.com

***Abstract*—With the proliferation of face image manipulation (FIM) techniques such as Face2Face and Deepfake, more fake face images are spreading over the internet, which brings serious challenges to public confidence. Face image forgery detection has made considerable progresses in exposing specific FIM, but it is still in scarcity of a robust fake face detector to expose face image forgeries under complex scenarios. Due to the relatively fixed structure, convolutional neural network (CNN) tends to learn image content representations. However, CNN should learn subtle tampering artifacts for image forensics tasks. We propose an adaptive residuals extraction network (AREN), which serves as pre-processing to suppress image content and highlight tampering artifacts. AREN exploits an adaptive convolution layer to predict image residuals, which are reused in subsequent layers to maximize manipulation artifacts by updating weights during the back-propagation pass. A fake face detector, namely ARENnet, is constructed by integrating AREN with CNN. Experimental results prove that the proposed AREN achieves desirable pre-processing. When detecting fake face images generated by various FIM techniques, ARENnet achieves an average accuracy up to 98.52%, which outperforms the state-of-the-art works. When detecting face images with unknown post-processing operations, the detector also achieves an average accuracy of 95.17%.**

***Index Terms*—facial image manipulation, passive image forensics, convolutional neural network, residuals extraction network.**

I. INTRODUCTION

FACE image is a widely-used biological modality, which contains rich and intuitive personal identity information such as gender, race, age, emotion and health status. However, face images have vulnerability and weak privacy, which makes them easy to be forged. Especially over the last three years, tremendous progresses such as DeepFake, generative models [1]–[3] and computer graphics (CG) based methods [4] have made facial image manipulations (FIM) reach photo-realistic level. This opens the door to many face image applications such as interactive game, movie industry and photography. However, FIM techniques might also be intentionally used for malicious purposes. In June 2019, the MIT Technology Review reported that the rapid spread of a doctored video, in which the White House speaker Nancy Pelosi was drunk, has frightened lawmakers in Washington[1]. Similar AI-enhanced synthetic media are also likely to be used in serious scientific research. Apparently, these FIM techniques bring serious crisis to social security and public confidence.

[1]https://www.technologyreview.com/s/613676/deepfakes-ai-congress-politics-election-facebook-social/.

Fig. 1: Can you identify which face image is fake? (a) Real face images with different resolutions. (b) From left to right, fake face images generated by Glow, StyleGAN, PGGAN, Face2Face, StarGAN, respectively.

Existing FIM techniques can be roughly divided into three categories: identity manipulation, expression manipulation and attribute transfer. Identity manipulation refers to generating fake face images of entirely imaginary people [5], or replacing one face in an image with another face via FaceSwap [6] and DeepFakes [7]. Expression manipulation refers to generating face images with specific expressions [2], or transferring facial expression from the source actor to the target face [4]. For face attribute transfer, it refers to changing the styles of face images, such as age, gender, hair color, etc [8]. In recent three years, face identity manipulation has made great progress [9]. The state-of-the-art methods such as PGGAN [10] and StyleGAN [11] can synthesize hyper-realistic fake face images with the resolution up to 1024×1024. The recently emerging expression manipulation techniques can generate fake face images without leaving any perceptible artifacts. Among them, a few generative models including CDAAE [12], ExprGAN [13], GANimation [14] and Glow [2] were proposed for expression manipulation with photo-realistic effects. Face2Face, which is a well-known CG-based method, animates well the facial expressions of the target video from a source actor [4]. For face attribute transfer, there also exist some generative models such as StarGAN [8], DIAT [15], CycleGAN [16], and IcGAN [17] to change facial attributes. Among them, StarGAN exploits a generator and a discriminator to obtain better visual quality than existing methods. Fig. 1 shows some examples of face images. Apparently, it is difficult for human eyes to differentiate those fake face images generated by FIM from real images. As we know, face image is an

important form of non-verbal communication, from which we can perceive true information. If face images are intentionally manipulated, it might bring serious influence to people, especially politicians and public figures. Thus, face image forgery detection is becoming a key issue to be solved in the community of image forensics.

Compared with the rapid progress of FIM techniques, fake face detection is still lagging far behind. Most existing works were proposed to expose some specific FIM technique [18]–[22], providing binary classification about the trustworthiness of face images. Only a few works expose multiple FIM forgeries [23]. Some works have studied the influence of post-processing [24]–[27], yet they have not fully addressed fake face image detection under complex scenarios. Actually, face images are inevitably compressed or resized before spreading over social media. To hide manipulation traces, manipulated face images usually suffer from some post-processing operations, which might include JPEG compression (JP), Scaling (SC), Gaussian Blur (GB), Mean Filtering (ME) and Median Filtering (MED). When the pre-trained detectors are detecting face images with unknown post-processing operations, there usually exist drastic performance degradations or they can be completely invalidated. Thus, the detection of multiple FIM forgeries under complex scenarios is becoming an urgent task to be solved. We need to develop a more general and robust fake face detector.

The conventional image forensics framework is made up of feature extraction and classification [28], [29]. The extracted features are usually hand-crafted, which leads to poor generalization capability. In recent years, Convolutional Neural Networks (CNN) has provided us an alternative to address the issue of feature selection, since it can learns feature implicitly and completes classification automatically [30]. Note that instead of learning content representation for image classification tasks, CNN should learn features from subtle tampering traces for image forensics [31]. Though CNN has achieved desirable accuracies when detecting copy-move, ME and JPEG recompression [32]–[34], the existing CNN forms still have their own constraints. If we want to further improve detection accuracy, the convolution layer should be forced to learn features from tampering traces by improving its standard form. To the best of our knowledge, there is only one attempt, namely MISLnet [31], to address this issue. The first convolution layer, which is called as the Constrained convolution (Constrained-Conv) layer, extracts low-level residuals for image forgery detection. In essence, the Constrained-Conv layer resets specific coefficients of the kernels after each iteration. The extracted residuals are fragile, which might be lost after passing through many layers. Though MISLnet provides some insights to improve the CNN model, there still remain some open questions. First, is it the best way to reset specific coefficients in the Constrained-Conv layer after each iteration? Second, can the low-level residuals be reused to improve the performance of the model?

To address these questions, we propose an Adaptive Residuals Extraction Network (AREN), which serves as pre-processing to suppress image content. AREN outputs prediction residuals which can characterize manipulation traces, but the residuals are obtained by subtracting the original image from the feature map. Thus, the residuals are different from those in [35], which are input feature map plus residual block output. For existing image forensics approaches, the general pipeline is to predict residuals and extract features from them for classification [36]–[38]. Motivated by this, we propose a fake face detector, namely ARENnet, to detect multiple FIM forgeries. ARENnet is constructed by integrating AREN with CNN. Specifically, the extracted residuals are reused in AREN to avoid information loss. Then, they are fed into the feature extraction network to obtain high-level features. The main works and contributions are summarized as follows.

- A pre-processing module, namely AREN, is specially designed for the CNN-based image forensics. Different from the fixed predictors in existing works, AREN predicts residuals adaptively during back-propagation. Thus, AREN can provide more discriminative residuals for image forensics tasks. AREN might serve as the basic residual predictor, which means that it can be transferred to the CNN-based models to detect other image forgeries.
- A robust fake face detector, namely ARENnet, is constructed by integrating AREN with CNN to expose the state-of-the-art FIM forgeries. To the best of our knowledge, this is the first attempt towards the detection of multiple FIM techniques under complex scenarios.
- We simulate the complex scenarios of practical face image forensics as real as possible. A series of experiments are conducted to prove the effectiveness of the proposed ARENnet. ARENnet achieves higher detection accuracy than existing works. In addition, we also explore the way to improve the generalization ability of the detector.

The remainder of this paper is organized as follows. Section II summarizes the related works on face image forensics. Section III presents the ARENnet for fake face detection. Section IV reports the experimental results and analysis. Conclusion is made in Section V.

II. Related Work

Machine learning have been widely-used in fake face detection [39]. To expose the face-swapping forgery, Zhang et al. [40] constructed a feature set of bag of words, which provides distinguishable information into SVM for binary classification. To detect the Face2Face reenacted facial expression forgery in videos, Guo et al. [18] exploited both texture-based moment features and optical flow-based motion features. To expose the synthesized face images by GAN, Li et al. [21] defined a similarity index by Chi-square distance to model the disparities in color components. In addition, Agarwal et al. [41] proposed a deepfake face video detection approach by exploiting the correlation between facial expressions and movements. However, these machine learning based works have poor generalization capability.

In recent years, many deep learning based works have been proposed to expose AI-generated fake face images [42]–[44]. Afchar et al. [45] proposed a compact CNN model, namely MesoNet, for facial video forgery detection. It achieved an average detection accuracy up to 95% on the FaceForensics

dataset. To detect fake face images by online face-swapping Apps including SwapMe[2] and FaceSwap[3], a two-stream network was proposed [46], in which GoogLeNet is trained to detect tampering artifacts in a face classification stream, and a patch based triplet network is trained to capture local noise residuals and camera characteristics as a second stream. Dang et al. [47] designed a customized CNN to detect face images generated by PGGAN [10] and BEGAN [48]. Nhu et al. [20] added a fully connected layer to VGG-Face [49], which is then fine-tuned to detect fake face images generated by DCGAN [50] and PGGAN [10]. Mo et al. introduced high pass filter into CNN to identify the PGGAN-generated face images [19]. Gera et al. [51] also proposed a temporal-aware pipeline to expose deepfake videos, which achieves an accuracy up to 97%. These deep learning based works need sufficient data and training time. Rössler et al. [52] built a FaceForensics dataset with 500K face images by exploiting the CG-based Face2Face. Later, an expanded FaceForensics++ [53] dataset was built by using four manipulation techniques including Face2Face [4], FaceSwap, DeepFakes [4], and NeuralTextures [54]. Dolhansky et al. [55] also built a DFDC dataset, which contains 5K videos generated by two facial forgery techniques. These three open datasets promote the development of face forensics.

Considering the particularity of face images, some methods exploited the biological inconsistency between real and fake faces. Li et al. [56] proposed to expose DeepFake videos by detecting the rate of eye blinking. Matern et al. [57] exposed fake face images by exploiting some visual artifacts such as the defects of reflection details near eyes, and the imprecise geometry of both nose and teeth. Ciftci et al. [58] presented a FakeCatcher to detect inauthentic portrait video by exploiting the biological signals of facial areas. Yang et al. [59] used 3D head poses to expose AI-generated fake face images. However, when there exist no obvious biological defects in fake face images, these methods might be invalidated as well.

Some works have addressed face forensics from new perspectives [60], [61]. Xuan et al. [62] improved the detector's generalization ability by adding noises in the training stage. Cozzolino et al. [63] addressed the issue of forensics transfer among different FIM techniques. Dang et al. [64] considered the issue of imbalanced samples. Yu et al. [65] proposed to discriminate fake face images synthesized by different GANs by their inherent fingerprints. Considering that the training of DeepFake is time-consuming, Li et al. [66] simulated the DeepFake-generated negative samples via simple image post-processing such as GB. Li et al. [67] also disrupted AI face synthesis with imperceptible adversarial perturbations. These efforts provide various insights to promote the development of face forensics towards universal forensics.

III. Proposed ARENnet Model

The AI-enhanced FIM techniques have enabled fake face images to be visually indistinguishable from real ones. Meanwhile, CNNs have been widely used in various forensics tasks due to their superior performance [63], [68]. The existing CNN models designed for image forensics can be divided into three categories: (1) Stacking standard CNN modules for specific image forensics task [20], [47]; (2) Using hand-crafted feature extractors such as high pass filter and SRM to extract residual features, which are then input into CNN for image forensics [19], [81]; (3) Improving the existing form of the convolution layer, such as Constrained-Conv [31], to directly learn image tampering traces.

Existing face image forensics works exploited either hand-crafted features or standard CNN to learn features. However, the existing CNN-based works have no explicit restrictions on the convolution layer to learn features from tampering traces. In this paper, we specifically design a pre-processing module, namely AREN, to predict residuals. Then, a robust fake face detector, namely ARENnet, is proposed to learn discriminative features from residuals. Fig. 2 is the architecture of the proposed ARENnet model. ARENnet can robustly detect multiple FIM forgeries even with various post-processing.

A. AREN

Some existing works learn features from prediction residuals such as SRM [69], SPAM [70] and median filter residuals [71]. They firstly generate a set of predicted pixel value via a fixed predictor $f(\cdot)$. Then, the prediction residuals R is obtained by subtracting the original pixel value from the predicted pixel value. That is,

$$R = f(I) - I \tag{1}$$

where I is the input image. Then, the prediction residuals are used as low-level features to construct high-level features for image forensics.

To mimic the above pipeline, AREN is specifically designed to automatically learn prediction residuals. Then, CNN is used to learn high-level features from prediction residuals due to its strong feature representation capability. As shown in Fig. 2, the first convolution layer (Conv 1) is used to predict the pixel value as follows.

$$F_j = \sum_{i=1}^{n} I_i * \omega_{ij} + b_j \tag{2}$$

where F_j is the j^{th} feature map which is output by the Conv 1 layer, $I_i * \omega_{ij}$ represents the convolution between the i^{th} channel of the input image I and the i^{th} channel of the j^{th} convolutional kernel in Conv 1, and b_j is the bias term of the j^{th} convolutional kernel. Then, the prediction residuals F_{res} are obtained by

$$F_{res} = F_j - I \tag{3}$$

Apparently, the way to obtain prediction residuals in Equation (3) is almost the same as Equation (1). The only difference is that AREN can adaptively update its weights. Specifically, the initial coefficients of the Conv 1 layer are randomly set. Then, the weights are updated by an iterative algorithm during the back-propagation pass [72]. In this paper, stochastic gradient descent (SGD) is used to train the model. The rules for iterative updates are defined as follows.

$$\bigtriangledown\omega_{ij}^{(n)} = \varepsilon\frac{\partial E}{\partial\omega_{ij}^{(n-1)}} - \theta_1 \cdot \bigtriangledown\omega_{ij}^{(n-1)} + \theta_2 \cdot \varepsilon \cdot \omega_{ij}^{(n-1)} \tag{4}$$

[2]https://itunes.apple.com/us/app/swapme-by-faciometrics/.
[3]https://github.com/MarekKowalski/FaceSwap/.
[4]https://github.com/deepfakes/faceswap.

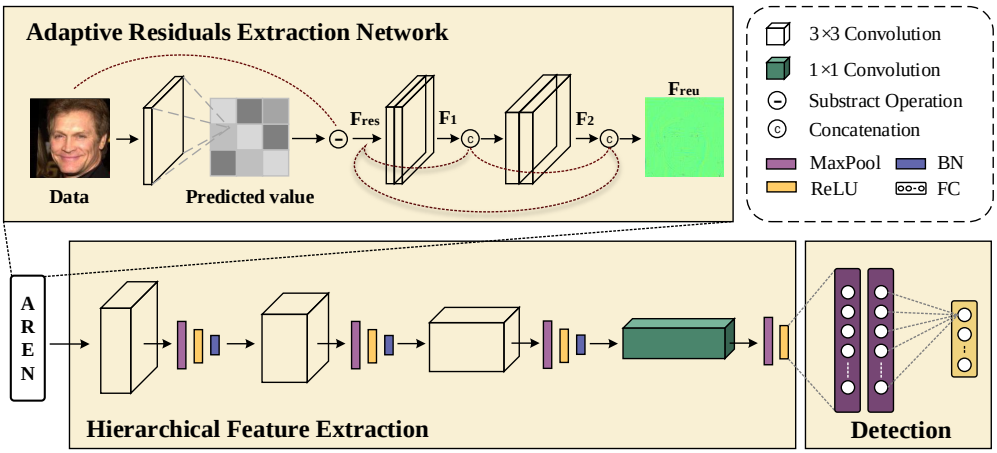

Fig. 2: ARENnet: our proposed ARENnet architecture. Given an input RGB image, we use the Conv 1 in AREN to obtain the feature map of image. Then, the original image is subtracted from the feature map in Conv 1 to extract the low-level prediction residuals F_{res}. Furthermore, the stable higher-level residuals, namely F_{reu}, are obtained by reusing the F_{res}. Next, the F_{reu} are passed to the subsequent convolution layers for hierarchical feature extraction to obtain high-level forensics features. Finally, we use fully connected layers and softmax function to classify the images. MaxPool: Max Pooling Layer; ReLU: Rectified Linear Unit; BN: Batch Normalization; FC: Fully Connected Layer; F_1 and F_2 represent the feature maps of the previous layer, respectively.

$$\omega_{ij}^{(n)} = \omega_{ij}^{(n-1)} - \bigtriangledown\omega_{ij}^{(n)} \tag{5}$$

where $\bigtriangledown$ denotes the gradient, $\omega_{ij}^{(n)}$ is the weight of the i^{th} channel of the j^{th} convolutional kernel in the n^{th} layer, E is the loss function, and ε is the learning rate. Furthermore, we use the momentum θ_1 and the decay θ_2 to accelerate model training [73]. In the iterative training process, its goal is to minimize the average loss E between true label and network output to make it converge [74]. AREN iteratively adjusts the weights to obtain better prediction residuals. The average loss E is defined as

$$E = \frac{1}{x}\sum_{i=1}^{x}\sum_{k=1}^{n} L_i^{(k)} log(y_i^{(k)}) \tag{6}$$

where $L_i^{(k)}$ is the true label of the i^{th} image in the k^{th} class, $y_i^{(k)}$ is the network output, x is the number of training sample, and n is the number of neurons in the output layer.

However, the residuals F_{res} extracted by the Conv 1 layer are fragile. If they are used directly, it might still lead to unstable training. To obtain stable prediction residuals, we borrow the idea of feature reusing by DenseNet [75]. Let c_1 and c_2 be two convolution layers, and $H_{c_1,c_2}(\cdot)$ denote the composite function of c_1 and c_2. Let $[\beta_1, \beta_2, ..., \beta_n]$ be the concatenation of the n feature maps. Thus, the residuals F_{res} is passed into Conv 2 and Conv 3 to obtain the intermediate feature map

$$F_1 = H_{c_2,c_3}(F_{res}) \tag{7}$$

The feature map obtained by concatenating F_1 and F_{res} is passed to Conv 4 and Conv 5, which can be expressed as

$$F_2 = H_{c_4,c_5}([F_1, F_{res}]) = H_{c_4,c_5}([H_{c_2,c_3}(F_{res}), F_{res}]) \tag{8}$$

Finally, we obtain the stable residuals as follows.

$$F_{reu} = [F_2, F_{res}, [H_{c_2,c_3}(F_{res}), F_{res}]] \tag{9}$$

AREN suppresses image content and obtains stable prediction residuals. Fig. 3 compares the prediction residuals obtained by different filters after 100 and 10,000 times of iterations. We can observe that when AREN iterates 100 times, most image contents are not suppressed. However, when the iteration times reach 10,000, most image contents are suppressed whereas keeping well manipulation traces. Note that different from the fixed predictor in existing works, AREN can adaptively learn prediction residuals, which are more suitable for image forensics, via the back-propagation pass.

B. Network Architectures

For the proposed ARENnet, most of the convolution layers adopt 3×3 kernels, since it has been claimed that the 3×3 convolutional kernels outperforms larger kernels [76]. Because there are three color channels in the input images, the Conv 1 layer uses three convolutional kernels to obtain the feature maps, respectively. Then, the prediction residuals F_{res} are obtained by subtracting the above feature maps from the original image. Let the input image be of size 128×128. As shown in Fig. 2, F_1 and F_{res} are firstly concatenated to obtain feature maps, whose dimension is 128×128×6. To fully exploit the features of the previous layer, the number of the convolutional kernels in the successive layer should not be less than the number of channels of the input feature map. Thus, six convolutional kernels are used for Conv 4 and Conv 5. Table I summarizes the parameters of ARENnet. In Section IV-C, we also analyze the influence of the number of convolutional kernels.

AREN obtains desirable prediction residuals F_{reu}. Instead of directly using them as the features for image forensics, we design a hierarchical feature extraction (HFE) module. F_{reu} is fed into the HFE module to learn high-level features for image forensics. Specifically, the HFE module is made up of

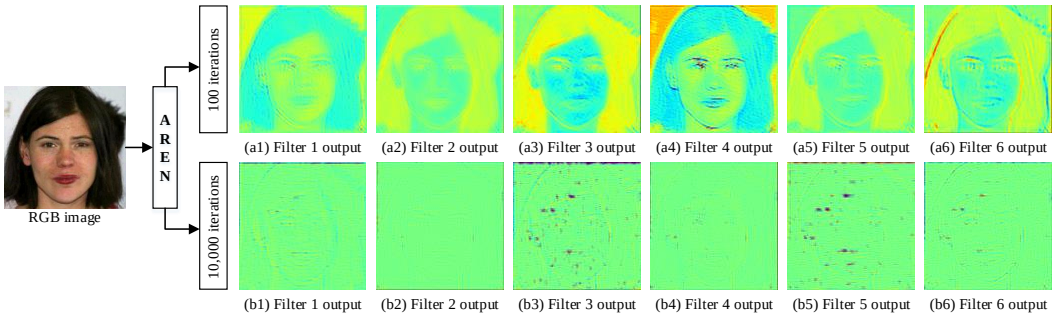

Fig. 3: The output of the six filters in AREN at different iterations. As the number of iterations increases, the feature map obtained by the AREN gradually suppresses the image content and retains the trace features.

four convolution layers, four max-pooling layers (MaxPool), four ReLU activation functions (ReLU), and three batch normalization (BN) layers.

For the four convolution layers, we gradually increase the number of the convolutional kernels. That is, Conv 6=24, Conv 7=48, Conv 8=64, and Conv 9=128. For the convolutional kernels, small stride can extracts more abundant features than large stride. Thus, the stride of each convolution layer is set to 1. Before feeding the feature maps into the classification module, another convolution layer, namely Conv 9, is introduced to achieve cross-channel interaction [77]. Different from the previous convolution layers, the Conv 9 layer adopts a 1×1 kernel. It learns the linear combination of those features located in the same location but different channels.

Each convolution layer is followed with other types of layers, which include MaxPool, ReLU and BN. The MaxPool layer retains the most representative information (i.e., the maximum value) within the sliding window. It also reduces the dimension of feature maps, and introduces network nonlinearity to prevent over-fitting. For the four MaxPool layers, they use the same kernel size of 3×3. To reduce the dimension of feature maps, the stride of each MaxPool layer is set to 2. The ReLU layer is introduced to increase network nonlinearity and overcome gradient vanishing. Thus, the ARENnet model can approximate any nonlinear function. Note that these nonlinear operations including MaxPool and ReLU are not introduced into AREN, which prevents the learned residuals from being destroyed by them. To accelerate training [78], the BN layer is also used in the ARENnet model to regularize the output of the convolution layers.

Finally, the learned deep features are passed into the classification module, which is made up of three fully connected layers. The first two fully connected layers, which learn the associations among deep features, have 300 neurons, respectively. The neurons in the last fully connected layer, whose outputs correspond to the real face image and possible face image manipulations.

TABLE I

SPECIFICATION OF THE ARENnet. "CONV", "MAXPOOL" AND "FC" CORRESPOND TO THE VARIABLES IN FIG. 2

Configuration				
Layers	Kernel sizes	Kernel quantities	Strides	Output sizes
Conv 1	3×3	3	1	128×128
Conv 2	3×3	3	1	128×128
Conv 3	3×3	3	1	128×128
Conv 4	3×3	6	1	128×128
Conv 5	3×3	6	1	128×128
Conv 6	3×3	24	1	128×128
MaxPool	3×3	\	2	64×64
Conv 7	3×3	48	1	62×62
MaxPool	3×3	\	2	31×31
Conv 8	3×3	64	1	29×29
MaxPool	3×3	\	2	14×14
Conv 9	1×1	128	1	14×14
MaxPool	3×3	\	2	7×7
FC 1: 300-dimension				
FC 2: 300-dimension				
Softmax Function				

IV. Experimental Results and Analysis

A. Experimental Settings

Most existing works provide only binary classification about the trustworthiness of face images, without considering more complex scenarios in practical forensics. In the experiments, we simulate some complex scenarios as real as possible to verify the effectiveness of the proposed ARENnet model. We conduct four groups of experiments. First, ARENnet is used to detect multiple FIMs. Second, we discuss the design of the ARENnet model. Third, some comparisons are made among ARENnet and the state-of-the-art works. Fourth, we explore the way to improve the robustness of ARENnet in complex scenarios.

Datasets. To conduct the above experiments, we firstly build a hybrid fake face (HFF) dataset [5], which contains eight types of face images. For real face images, three types of face images are randomly selected from three open datasets. They are low-resolution face images from CelebA [79], high-resolution face images from CelebA-HQ [10], and face video frames from FaceForensics [52], respectively. Thus, real face images under internet scenarios are simulated as real as possible. Then, some most representative FIM techniques, which include PGGAN [10] and StyleGAN [11] for identity manipulation, the CG-based Face2Face [4] and Glow [2] for face expression manipulation, and StarGAN [8] for face attribute transfer, are selected to produce fake face images. Note that since StarGAN can transfer facial attributes such as hair color (black, blond, brown) gender (male or female) and age (young or old) to

[5]https://github.com/EricGzq/Hybrid-Fake-Face-Dataset

TABLE II

THE DETAILS OF HFF DATASETS

	Data Type	Description	Image Size	Corpus Size
Real Face Images	CelebA	Low-resolution face images	178×218	25k
	CelebA-HQ	High-resolution face images	1024×1024	10k
	YouTube-Frame	Face video frames	Random size	25k
Fake Face Images	PGGAN	A generative model based identity manipulation technique.	1024×1024	10k
	StyleGAN		1024×1024	10k
	Glow	A generative model based expression manipulation technique.	256×256	25k
	Face2Face	A CG-based expression manipulation technique.	Random size	25k
	StarGAN	A generative model based attribute transfer technique.	256×256	25k

other domains, five types of face attributes are manipulated via StarGAN. It has been claimed that face images with different attributes share the same artifacts or fingerprints when they are generated by the same GAN [65]. We mark these images with different attributes as StarGAN-generated. Table II summarizes the details of the HFF dataset. In addition, the open FaceForensics++ [53] dataset is used for experiments. There are 1k original video sequences, which are manipulated by four FIM techniques including Deepfakes, Face2Face, FaceSwap and NeuralTextures. The tampered videos are further compressed with two quality levels, namely high quality videos (HQ) and low quality videos (LQ). For each compressed video, ten face frames are extracted. Thus, there are totally 40k face images for training and 10k face images for testing.

Evaluation Criterion. Image forgery detectors are usually evaluated by classification accuracy. In our multi-classification tasks, since the distribution of data is roughly balanced, we also use classification accuracy for performance evaluation. To further evaluate the performance gains brought by different AREN design selections, the relative error reduction (RER) [31] is also used as performance evaluation metrics. Let E_1 and E_2 be the numbers of errors for two detectors ($E_1 > E_2$). RER is defined as $\mathrm{RER} = (E_1 - E_2)/E_1$.

Baseline Models. We choose some state-of-the-art works as the baselines for comparisons. They are summarized below.

- Meso-4 [45]: It exploits the mesoscopic properties of face images for facial forgery detection.
- Hand-Crafted-Res [19]: Three high pass filters are used as pre-processing to extract hand-crafted features. The parameters with the best performance are used for comparisons.
- MISLnet [31]: It uses the Constrained-Conv layer, which can suppress image content and adaptively learn low-level residual features for universal forensics.
- XceptionNet [80]: For the FaceForensics++ dataset, XceptionNet achieved the best performance [53].
- Model-base: To prove the gains brought by AREN, it is removed from the ARENnet model. The rest network is called the Model-base here.
- Hand-Crafted-Res-Model-base: We use the hand-crafted feature extractor in [19] to replace AREN.
- Constrained-Conv-Model-base: AREN is replaced by the Constrained-Conv in [31].
- SRM-Model-base: AREN is replaced by the SRM filter kernels in [81].

Implementation Details. The ARENnet model is implemented under the Caffe framework [82]. We convert all face images in the datasets into the LMDB format and then resize them into 128×128 for use in Caffe. When training on the HFF dataset, each forensics model has 10 training epochs. Since fewer frames are extracted from the FaceForensics++ dataset, each model has 20 epochs when trained on this dataset. We record the detection accuracies on the testing set after every 1000 iterations. Two Nvidia GeForce GTX 1080 Ti GPUs are used to train the model.

B. Detection of multiple FIM forgeries

Since it is relatively easy to detect fake face images generated by some specific FIM technique, the proposed detector is used to expose multiple FIM techniques simultaneously. That is, each type of face images are randomly selected from the HFF dataset and divided into three sub-datasets for training (75%), validation (5%), and testing (20%), respectively. Note that test images have never appeared in the training set and the validation set. In the experiments, there are about 116k face images for training, which include real images with different resolutions and five types of fake images. When training the ARENnet model, SGD is used for iterative optimization, and we set the momentum $\theta_1 = 0.95$ and the decay $\theta_2 = 0.005$. The learning rate is defined as

$$\varepsilon = \varepsilon_b \times \gamma^{\lfloor (\frac{\alpha}{N}) \rfloor} \tag{10}$$

where ε_b is the basic learning rate, N is the fixed step size, α denotes current iteration, and $\lfloor \rfloor$ denotes rounding down. Their initial values are as follows: $\varepsilon_b = 0.001$, $\gamma = 0.5$, $N = 1000$. With the increment of the iteration times, ε decreases periodically. The batch size is set to 64. Each training epoch requires 1,817 iterations.

The confusion matrixes of Model-base and ARENnet are reported in Table III and Table IV, respectively. Their average detection accuracies are 96.16% and 98.52%, respectively. We can observe from Table III that the false detection rate between PGGAN and CelebA-HQ is high. Actually, two types of face images share similar textures. They are difficult to be detected, especially when they are resized into 128×128. For Face2Face and YouTube-Frame, there also exist the same phenomenon.

TABLE III

CONFUSION MATRIX FOR IDENTIFYING VARIOUS TYPES OF MANIPULATIONS USING MODEL-BASE

THE ASTERISKS "*" REPRESENTS THE VALUE ARE BELOW 1%

		Predicted class							
		CelebA	**CelebA-HQ**	**YouTube-Frame**	**Glow**	**StarGAN**	**PGGAN**	**StyleGAN**	**Face2Face**
The class	**CelebA**	**99.48%**	*	*	*	*	*	*	*
	CelebA-HQ	*	**87.05%**	*	*	*	11.90%	*	*
	YouTube-Frame	*	*	**92.28%**	*	*	*	*	7.62%
	Glow	*	*	*	**99.90%**	*	*	*	*
	StarGAN	*	*	*	*	**99.74%**	*	*	*
	PGGAN	*	19.35%	*	*	*	**80.25%**	*	*
	StyleGAN	*	*	*	*	*	*	**99.80%**	*
	Face2Face	*	*	1.98%	*	*	*	*	**97.98%**

TABLE IV

CONFUSION MATRIX FOR IDENTIFYING VARIOUS TYPES OF MANIPULATIONS USING ARENNET

THE ASTERISKS "*" REPRESENTS THE VALUE ARE BELOW 1%

		Predicted class							
		CelebA	**CelebA-HQ**	**YouTube-Frame**	**Glow**	**StarGAN**	**PGGAN**	**StyleGAN**	**Face2Face**
The class	**CelebA**	**99.56%**	*	*	*	*	*	*	*
	CelebA-HQ	*	**95.30%**	*	*	*	3.45%	*	*
	YouTube-Frame	*	*	**97.68%**	*	*	*	*	2.10%
	Glow	*	*	*	**99.92%**	*	*	*	*
	StarGAN	*	*	*	*	**99.66%**	*	*	*
	PGGAN	*	8.50%	*	*	*	**91.45%**	*	*
	StyleGAN	*	*	*	*	*	*	**99.85%**	*
	Face2Face	*	*	*	*	*	*	*	**99.38%**

TABLE V

THE COMPARISON OF DIFFERENT RESIDUAL

EXTRACTION METHODS

Methods	Accuracy
Hand-Crafted-Res-Model-base [19]	97.50%
Constrained-Conv-Model-base [31]	95.24%
SRM-Model-base [81]	97.49%
ARENnet	**98.52%**

Note that when AREN is added into Model-base, which turns into ARENnet, the false detection rate is greatly reduced.

We also makes comparisons among different residual extraction methods including Hand-Crafted-Res, Constrained-Conv, SRM and AREN. They are followed with the same basic CNN model (Model-base) for experiments. Table V compares the experimental results achieved by different residual extraction methods. ARENnet achieves the highest accuracy of 98.52%, which proves that AREN achieves the best residual extraction capability. The reasons behind this are summarized as follows. Both Hand-Crafted-Res and SRM use the fixed filter to extract residuals, Constrained-Conv resets specific coefficients after each iteration, whereas AREN adaptively updates the coefficients during the back-propagation pass to predict residuals. AREN also introduces feature reusing to improve detection accuracy.

C. Design Selection of the ARENnet Model

The CNN model has direct impacts on detection performance. For the ARENnet model, its AREN design is also important, since AREN learns prediction residuals for forensics. Actually, AREN is very flexible with the following issues to be further investigated by experiments: (1) Are the residual features F_{res} better for forensics than image data itself? (2) Whether reusing F_{res} will improve the ARENnet or not? (3) How many kernels are appropriate for Conv 4 and Conv 5? (4) Is 3×3 convolutional kernel in the first layer better than 5×5 convolutional kernel? (5) What is the effect of the convolution layer on AREN?

To address the above issues, we have made some changes to AREN, which are summarized as Fig. 4. For the ARENnet model itself, we further discuss two issues: (1) *The pooling layer*. As we know, there are two common pooling strategies, namely max pooling and average pooling. Since ARENnet adopts max pooling for all the pooling layers, it will be

TABLE VI

IDENTIFICATION RATE FOR DIFFERENT CNN MODELS

Models		Description	Average Accuracy	RER
ARENnet		\	**98.52%**	-
Model-base		Remove the AREN	96.16%	61.46%
Modified	AREN_1	Image data is used instead of residual features F_{res}	97.14%	48.25%
	AREN_2	F_{res} is not reused	97.46%	41.73%
	AREN_3	Conv 4 = 3 and Conv 5 = 3	97.42%	42.64%
	AREN_4	Conv 4 = 12 and Conv 5 = 12	97.88%	30.19%
	AREN_5	5×5 convolutional kernel as the predictor	97.81%	32.42%
	AREN_6	Remove Conv 3 and Conv 5	98.29%	13.45%
Other	Net_7	All pooling functions are replaced by average pooling	96.85%	53.02%
	Net_8	The 1×1 convolutional kernel in Conv 9 is replaced by 3×3	98.36%	9.76%

TABLE VII

IDENTIFICATION RATE FOR DIFFERENT FORENSICS MODELS

ON HFF DATASET

Methods	Raw	JP60	ME5	Average
Meso-4 [45]	80.76%	67.76%	62.40%	70.31%
Hand-Crafted-Res [19]	90.54%	73.81%	74.99%	79.78%
MISLnet [31]	93.76%	86.32%	79.06%	86.38%
XceptionNet [80]	97.17%	78.62%	90.88%	88.89%
ARENnet	**98.52%**	**91.02%**	**92.42%**	**93.99%**

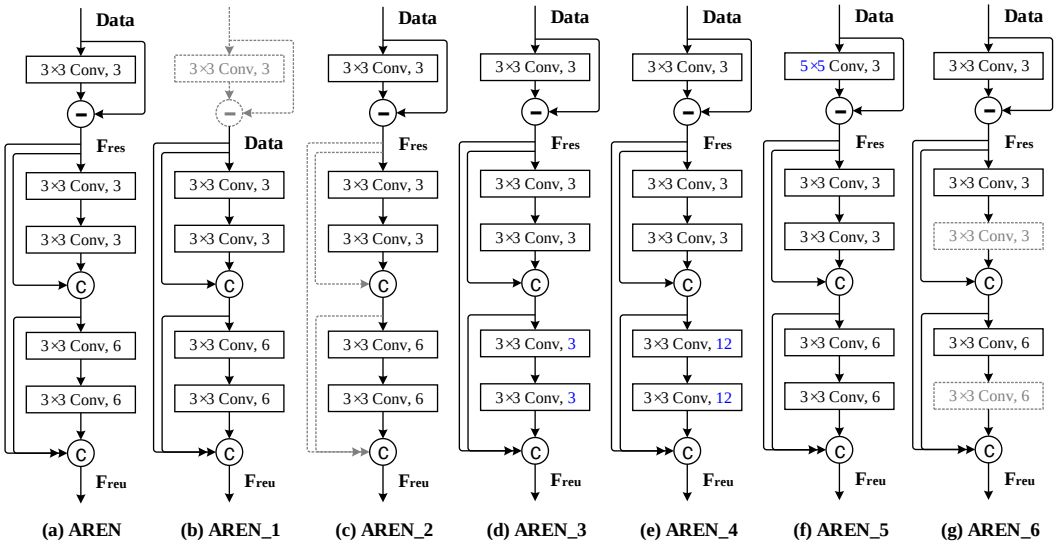

Fig. 4: The proposed AREN and the six modified versions. The gray dotted line indicates the removed part, and the blue font indicates the modified parameter.

replaced with average pooling for experiments. (2) *The 1×1 convolution layer*. Its main purpose is to achieve cross-channel interaction and information integration [77]. To verify the 1×1 convolutional kernel in Conv 9, it will be replaced with the 3×3 convolutional kernels. To make fair comparisons, We use the same datasets described in Section IV-B for experiments. The ARENnet model after each modification is trained for 10 epochs. The batch size is 64, and each epoch requires 1,817 iterations.

Table VI reports the average detection accuracy and RER for the proposed ARENnet model with different structures or parameters. From it, we have the following observations. First, AREN serves as an effective pre-processing module for the ARENnet model, since it extracts low-level residual features suitable for image forensics. If image data is directly used for feature learning, the detection accuracy will decrease 1.38%. Second, for the convolutional kernels in the first layer, the size of 3×3 is better than the size of 5×5, which improves the detection accuracy about 0.71%. A bigger receptive field does not lead to better detection accuracy, and 3×3 convolutional kernels are sufficient for excellent feature extraction. Third, it is a nice choice to use two convolution layers between two concatenation operations. The experimental results prove that two convolution layers are more stable than one convolution layer. Fourth, max pooling is more preferable than average pooling for the ARENnet, simply because it improves the detection accuracy about 1.67%. Finally, the 1×1 convolutional kernel in the Conv 9 layer improves 0.16% detection accuracy than the 3×3 convolutional kernels, which benefits from the cross-channel interaction and information integration.

D. Comparisons with State-of-the-art works

Comparisons are made among ARENnet and some state-of-the-art works. Note that because Hand-Crafted-Res [19] and MISLnet [31] are designed for other forensics tasks, they can not converge for our forensics task. Thus, we replace the initialization method *Gaussian* with *Xavier* for Hand-Crafted-Res, and we adjust the step size to 1000 for MISLnet. We use the same dataset described in Section IV-B for experiments. To hide the traces left by various FIMs, JP and ME are conducted on this dataset as post-processing, respectively. The quality factor of JP is set to 60 (JP60), and the kernel size of ME is set to 5×5 (ME5). Table VII reports the detection accuracies under three scenarios. We can observe that for most detectors, both JP60 and ME5 greatly degrades the detection accuracies. However, ARENnet achieves much better accuracies under three scenarios.

TABLE VIII

IDENTIFICATION RATE FOR DIFFERENT FORENSICS MODELS ON FACEFORENSICS++ DATASET

Methods	HQ	LQ	Average
Meso-4 [45]	42.30%	35.67%	38.99%
Hand-Crafted-Res [19]	66.16%	40.65%	53.41%
MISLnet [31]	69.92%	55.28%	62.60%
XceptionNet [80]	48.82%	42.37%	45.60%
ARENnet	**79.05%**	**56.31%**	**67.68%**

Another experiment is also conducted on the FaceForensics++ dataset. For each real face image, there are four forged face images. Since these images are obtained from compressed videos, they have poor qualities. We detect five types of face images, and the experimental results are reported in Table VIII. ARENnet still achieves the best detection accuracies

E. Detection Robustness

When face images are spreading over the internet, they usually suffer from some image manipulations with various parameters. It is almost impossible for a face image forgery detector to learn from fake face images under various scenarios. For the ARENnet model, its generalization capability is worthy of further investigation. To further explore the way to improve the detection robustness under complex scenarios, we

TABLE IX

PARAMETER LIST OF IMAGE OPERATIONS

Image Operations		Parameters
Spatial Filtering	Mean Filtering (ME)	kernel size: 3×3, 5×5, 7×7
	Gaussian Filtering (GB)	kernel size: 3×3, 5×5, 7×7; Standard deviation: 0
	Median Filtering (MED)	kernel size: 3×3, 5×5, 7×7
Spatial Enhancement	Gamma Correction (GC)	gamma: 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0
Lossy Compression	JPEG Compression (JP)	quality factor: 60-90
	JPEG Compression 2000 (JP2)	compression ratio: 2.0-8.0
Resampling	Scaling (SC)	up-sampling (%): 1, 3, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90 down-sampling (%): 1, 3, 5, 10, 15, 20, 25, 30, 35, 40, 45

TABLE X

CONFUSION MATRIX FOR VERIFYING THE GENERALIZATION ABILITY USING ARENNET.

		Testing Set					
		Raw	JP60	JP-mix	ME5	ME-mix	Average
Training Set	Raw	**98.52%**	76.41%	76.41%	75.37%	77.11%	81.54%
	JP60	90.19%	**91.02%**	87.43%	54.61%	59.40%	76.53%
	JP-mix	93.67%	88.78%	**90.33%**	69.81%	73.29%	83.18%
	ME5	74.66%	47.57%	55.52%	**92.42%**	87.87%	71.61%
	ME-mix	91.67%	65.53%	67.45%	90.07%	**90.74%**	81.09%

TABLE XI

IDENTIFICATION RATE OF ARENNET TRAINING ON MIXED DATA.

	Training on mixed data		
	Small	Middle	Large
Raw	95.03%	95.36%	96.32%
JP60	85.81%	86.46%	88.52%
JP-mix	86.35%	86.87%	89.31%
ME5	87.66%	91.92%	91.35%
ME-mix	88.52%	92.05%	92.82%
Average	88.67%	90.53%	91.66%

made the following two assumptions. (1) By applying some post-processing operations to face image data, can the detector learn essential differences among various FIMs to improve its generalization capability? (2) Compared with single parameter, can the post-processing operation with mixed parameters improve the generalization capability of the detector.

To address the above assumptions, some widely-used image operations are performed on the HFF dataset to simulate face images spreading over the internet. Table IX summarizes the parameters of these image operations. Note that if the dataset has been suffered from a specific image operation with mixed parameters, it is denoted as 'image operation' plus '-mix', such as JP-mix, ME-mix, etc. We select two representative image operations, namely JP and ME, for experiments. Lossy compression can easily confuse the judgment of the detector by reducing image quality. Spatial filtering can hide image details such as manipulation artifacts by blurring the image. They are selected to destroy the traces left in face images by different FIM forgeries. There are five types of face images, which include original face images (Raw), JP60 compressed images (JP60), JP-mix compressed images (JP-mix), ME5 filtered images (ME5), and ME-mix filtered images (ME-mix), respectively. In the experiments, the detector is firstly trained with one type of face images, and then the pre-trained model is tested with the other four types of face images.

Table X reports the confusion matrixes when ARENnet are testing five types of face images. From it, we can observe that the detector trained on JP is also effective when detecting Raw and JP, while the detector trained on ME achieves desirable accuracy when detecting Raw and ME. Furthermore, though JP and ME are two distinct image operations to manipulate images, we can still observe that the image operations with mixed parameters enable the detector to learn more discriminative features, and thus improve the generalization ability.

TABLE XII

VERIFY THE GENERALIZATION ABILITY OF THE ARENNET.

Operation type		Training on mixed data		
		Small	Middle	Large
Spatial filtering	GB-mix	92.07%	94.57%	96.06%
	MED-mix	92.22%	94.75%	95.05%
Spatial enhancement	GC-mix	88.01%	91.89%	93.52%
Lossy compression	JP2-mix	94.64%	94.89%	95.90%
Resampling	SC-mix	94.09%	94.82%	95.34%
Average		92.21%	94.18%	95.17%

We also select face images from Raw, JP-mix and ME-mix datasets with the same proportion to construct Small, Middle and Large mixed datasets, respectively. The ARENnet model is trained on the mixed training datasets of 124k, 165k, and 372k face images, respectively. Then, the trained detector is used to identify Raw, JP, JP-mix, ME, and ME-mix, respectively. Table XI reports the experimental results. The detection accuracies also increase with the increase of training data. Their average accuracies are 88.67%, 90.53%, and 91.66%, respectively.

For the generalization capability, there is still a question left: whether the detector trained by the above method can detect face images with other unknown operations? To verify this, the trained ARENnet is tested to detect some other types of face images, such as GB-mix, MED-mix, GC-mix, JP2-mix, and SC-mix. The experimental results are reported in Table XII. The average accuracy is 95.17%. That is, ARENnet achieves desirable generalization capability, especially when it is trained on the large dataset. This proves that training the detector with those face images after image operations with mixed parameters is an effective strategy to enhance detection robustness, since the detector can learn more discriminative features from them.

V. Conclusion

The latest AI-enhanced fake face images can achieve photo-realistic visual qualities, which are quite challenging to be detected. In this work, we addressed fake face image detection under complex scenarios. Due to the relatively fixed structure, there are some limitations for the existing CNN-based works. We proposed a simple yet effective AREN module as pre-processing. AREN exploits the convolution

layer to serve as a predictor to obtain image residuals. The weights are updated adaptively during the back-propagation pass. In subsequent layers, the prediction residuals are reused to maximize manipulation traces. We also designed a fake face detector, namely ARENnet, by integrating AREN with CNN. That is, the prediction residuals obtained by AREN are fed into CNN to learn more discriminative features. The proposed ARENnet model was systematically validated by a series of experiments. The experimental results showed that ARENnet achieved superior detection accuracy and desirable generalization capability. Compared with MISLnet, ARENnet improved detection accuracy about 4.76% on average when testing on the HFF dataset. This mainly benefits from the AREN, which is much better for residual extraction than Constrained-conv and SRM. Some common post-processing operations are selected to simulate the practical forensics under complex scenarios. We have also explored the way to improve the detector's robustness. It is worthy of mention that AREN might serve as a basic residual predictor for other image forensic tasks. For future work, we will further improve the robustness of the detector under complex scenarios.

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio. "Generative adversarial nets," in *Proc. NIPS.*, Dec. 2014, pp. 2672-2680.

[2] D. P. Kingma, P. Dhariwal. "Glow: Generative flow with invertible 1×1 convolutions," in *Proc. NIPS.*, Dec. 2018, pp. 10215–10224.

[3] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen et al, I. Sutskever, and M. Welling. "Improved variational inference with inverse autoregressive flow," in *Proc. NIPS.*, Dec. 2016, pp. 4743-4751.

[4] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. "Face2face: Real-time face capture and reenactment of RGB videos," in *Proc. CVPR.*, Jun. 2016, pp. 2387-2395.

[5] H. Huang, Z. Li, R. He, Z. Sun, and T. Tan. "Introvae: Introspective variational autoencoders for photographic image synthesis," in *Proc. NIPS.*, Dec. 2018, pp. 52-63.

[6] I. Korshunova, W. Shi, J. Dambre, and L. Theis. "Fast Face-Swap Using Convolutional Neural Networks," in *Proc. ICCV.*, Oct. 2017, pp. 3677-3685.

[7] P. Korshunov, and S. Marcel. (Dec. 2018). "DeepFakes: a New Threat to Face Recognition? Assessment and Detection." [Online]. Available: https://arxiv.org/abs/1812.08685

[8] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, and J. Choo. "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. CVPR.*, Jun. 2018, pp. 8789-8797.

[9] L. Liang, and X. Zhang, "Adaptive Label Propagation for Facial Appearance Transfer," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3068-3082, Dec. 2019.

[10] T. Karras, T. Aila, S. Laine, and J. Lehtinen. "Progressive Growing of GANs for Improved Quality, Stability, and Variation," in *Proc. ICLR.*, Apr. 2018, pp. 1-26.

[11] T. Karras, S. Laine, and T. Aila. "A Style-Based Generator Architecture for Generative Adversarial Networks," in *Proc. CVPR.*, Jun. 2019, pp. 4401-4410.

[12] Y. Zhou, B. E. Shi. "Photorealistic facial expression synthesis by the conditional difference adversarial autoencoder," in *Proc. Int. Conf. Affective Comput. Intell. Interaction*, Oct. 2017, pp. 370-376.

[13] H. Ding, K. Sricharan, and R. Chellappa. "Exprgan: Facial expression editing with controllable expression intensity," in *Proc. AAAI.*, Feb. 2018, pp. 6781-6788.

[14] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer. "Ganimation: Anatomically-aware facial animation from a single image," in *Proc. ECCV.*, Sep. 2018, pp. 818-833.

[15] M. Li, W. Zuo, and D. Zhang. (Oct. 2016). "Deep identity-aware transfer of facial attributes." [Online]. Available: https://arxiv.org/abs/1610.05586

[16] J. Zhu, T. Park, P. Isola, and A. A. Efros. "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. ICCV.*, Oct. 2017, pp. 2223-2232.

[17] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez. (Nov. 2016). "Invertible conditional gans for image editing." [Online]. Available: https://arxiv.org/abs/1611.06355

[18] C. Guo, G. Luo, and Y. Zhu. "A detection method for facial expression reenacted forgery in videos," in *Proc. 10th Int. Conf. Digit. Image Process.*, Aug. 2018, pp. 414-422.

[19] H. Mo, B. Chen, and W. Luo. "Fake faces identification via convolutional neural network," in *Proc. 6th ACM Workshop Inf. Hid. Multimedia Security*, Jun. 2018, pp. 43-47.

[20] T. D. Nhu, I. S. Na, and S. H. Kim. "Forensics Face Detection From GANs Using Convolutional Neural Network," in *Proc. Int. Symp. Inf. Technol. Convergence*, Oct. 2018, pp. 376-379.

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

[22] L. Nataraj, T. M. Mohammed, S. Chandrasekaran, A. Flenner, J. H. Bappy, A. K. Roy-Chowdhury, and B. S. Manjunath. (Mar 2019). "Detecting GAN generated Fake Images using Co-occurrence Matrices." [Online]. Available: https://arxiv.org/abs/1903.06836

[23] C. Hsu, C. Lee and Y. Zhuang. "Learning to Detect Fake Face Images in the Wild," in *Proc. Int. Symp. Comput. Consum. Control*, Dec. 2018, pp. 388-391.

[24] F. Marra, D. Gragnaniello, D. Cozzolino and L. Verdoliva. "Detection of GAN-Generated Fake Images over Social Networks," in *Proc. IEEE Conf. Multimedia Inf. Process. Retrieval*, Apr. 2018, pp. 384-389.

[25] H. H. Nguyen, J. Yamagishi and I. Echizen. "Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May. 2019, pp. 2307-2311.

[26] P. He, H. Li, and H. Wang. "Detection of Fake Images Via The Ensemble of Deep Representations from Multi Color Spaces," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2019, pp. 2299-2303.

[27] D. Cozzolino, G. Poggi, and L. Verdoliva. "Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection," in *Proc. 5th ACM Workshop Inf. Hid. Multimedia Security*, Jun. 2017, pp. 159-164.

[28] X. Feng, I. J. Cox, and G. Doerr, "Normalized Energy Density-Based Forensic Detection of Resampled Images," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 536-545, Jun. 2012.

[29] X. Zhao, Y. Lin, and J. Heikkil, "Dynamic Texture Recognition Using Volume Local Binary Count Patterns With an Application to 2D Face Spoofing Detection," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 552-566, Mar. 2018.

[30] S. Wu, S. Zhong, and Y. Liu, "A Novel Convolutional Neural Network for Image Steganalysis With Shared Normalization," *IEEE Trans. Multimedia*, vol. 22, no. 1, pp. 256-270, Jan. 2020.

[31] B. Bayar, and M. C. Stamm. "Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2691-2706, Apr. 2018.

[32] J. Chen, X. Kang, Y. Liu, and Z. J. Wang. "Median Filtering Forensics Based on Convolutional Neural Networks," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 1849-1853, Nov. 2015.

[33] Y. Rao, and J. Ni. "A deep learning approach to detection of splicing and copy-move forgeries in images," in *Proc. IEEE Int. Workshop Inf. Forensics Security*, Dec. 2016, pp. 1-6.

[34] Q. Wang, and R. Zhang. "Double JPEG compression forensics based on a convolutional neural network," *EURASIP J. Info. Security*, vol. 2016, no. 1, pp. 1-12, Oct. 2016.

[35] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition," in *Proc. CVPR.*, Jun. 2016, pp. 770-778.

[36] A. C. Popescu, and H. Farid. "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758-767, Feb. 2005.

[37] X. Qiu, H. Li, W. Luo, and J. Huang. "A universal image forensic strategy based on steganalytic model," in *Proc. 2th ACM Workshop Inf. Hid. Multimedia Security*, Jun. 2014, pp. 165-170.

[38] M. Kirchner. "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *Proc. 10th ACM Workshop Multimedia Security*, New York, NY, USA, 2008, pp. 11-20.

[39] X. Yang, Y. Li, H. Qi, and S. Lyu. (Mar. 2019). "Exposing GAN-synthesized Faces Using Landmark Locations." [Online]. Available: https://arxiv.org/abs/1904.00167

[40] Y. Zhang, L. Zheng and V. L. L. Thing. "Automated face swapping and its detection," in *Proc. IEEE 2nd Int. Conf. Signal Image Process.*, Aug. 2017, pp. 15-19.

[41] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, and H. Li. "Protecting World Leaders Against Deep Fakes," in *Proc. CVPR. Workshops*, Jul. 2019, pp. 38-45.

[42] A. Jain, R. Singh and M. Vatsa. "On Detecting GANs and Retouching based Synthetic Alterations," in *Proc. IEEE 9th Int. Conf. Biometrics Theory, Appl., Syst.*, Oct. 2018, pp. 1-7.

[43] S. Tariq, S. Lee, H. Kim, Y. Shin, and S. S. Woo. "GAN is a friend or foe?: a framework to detect various fake face images," in *Proc. 34th ACM/SIGAPP Symp. Appl. Comp.*, Apr. 2019, pp. 1296-1303.

[44] Y. Zhuang and C. Hsu. "Detecting Generated Image Based on a Coupled Network with Two-Step Pairwise Learning," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2019, pp. 3212-3216.

[45] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen. "Mesonet: a compact facial video forgery detection network," in *Proc. IEEE Int. Workshop Inf. Forensics Security*, Dec. 2018, pp. 1-7.

[46] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. "Two-stream neural networks for tampered face detection," in *Proc. CVPR. Workshops*, Jul. 2017, pp. 1831-1839.

[47] L. M. Dang, S. I. Hassan, S. Im, J. Lee, S. Lee, and H. Moon. "Deep learning based computer generated face identification using convolutional neural network," *Appl. Sci.*, vol. 8, no. 12, pp. 2610-2628, Dec. 2018.

[48] D. Berthelot, T. Schumm, and L. Metz. (May. 2017). "Began: Boundary equilibrium generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1703.10717

[49] O. M. Parkhi, A. Vedaldi, and A. Zisserman. "Deep face recognition," in *Proc. BMVC.*, Sep. 2015, pp. 1-12.

[50] A. Radford, L. Metz, and S. Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. ICLR.*, May. 2016, pp. 1-16.

[51] D. Gera and E. J. Delp. "Deepfake Video Detection Using Recurrent Neural Networks," in *Proc. IEEE 15th Int. Conf. Advanced Video Signal Based Surveillance*, Nov. 2018, pp. 1-6.

[52] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. (Mar. 2018). "FaceForensics: A large-scale video dataset for forgery detection in human faces." [Online]. Available: https://arxiv.org/abs/1803.09179

[53] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. (Apr. 2019). "FaceForensics++: Learning to Detect Manipulated Facial Images." [Online]. Available: https://arxiv.org/abs/1901.08971

[54] J. Thies, M. Zollhöfer, and M. Nießner. "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics*, 2019.

[55] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer. (Oct. 2019) "The Deepfake Detection Challenge (DFDC) Preview Dataset." [Online]. Available: https://arxiv.org/abs/1910.08854

[56] Y. Li, M. Chang and S. Lyu. "In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking," in *Proc. IEEE Int. Workshop Inf. Forensics Security*, Dec. 2018, pp. 1-7.

[57] F. Matern, C. Riess and M. Stamminger. "Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations," in *Proc. IEEE Wint. Appl. Comput. Vis. Workshops*, Jan. 2019, pp. 83-92.

[58] U. A. Ciftci, and I. Demir. (Jan. 2019). "FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals." [Online]. Available: https://arxiv.org/abs/1901.02212

[59] X. Yang, Y. Li and S. Lyu. "Exposing Deep Fakes Using Inconsistent Head Poses," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May. 2019, pp. 8261-8265.

[60] H. Jeon, Y. Bang, and S. S. Woo. "FakeTalkerDetect: Effective and Practical Realistic Neural Talking Head Detection with a Highly Unbalanced Dataset," in *Proc. ICCV.*, Oct. 2019.

[61] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi. "Do GANs Leave Artificial Fingerprints," in *Proc. IEEE Conf. Multimedia Inf. Process. Retrieval*, Mar. 2019, pp. 506-511.

[62] X. Xuan, B. Peng, W. Wang, and J. Dong. "On the generalization of GAN image forensics," in *Proc. Chinese Conf. Biometric Recognition*, Oct. 2019, pp. 134-141.

[63] D. Cozzolino, J. Thies, A. Rssler, C. Riess, M. Niener, and L. Verdoliva. (Dec. 2018). "ForensicTransfer: Weakly-supervised Domain Adaptation for Forgery Detection." [Online]. Available: https://arxiv.org/abs/1812.02510

[64] L. M. Dang, S. I. Hassan, S. Im, and H. Moon. "Face image manipulation detection based on a convolutional neural network," *Expert Syst. Appl.*, vol. 129, no. 1, pp. 156-168, Sep. 2019.

[65] N. Yu, L. Davis, and M. Fritz. "Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints," in *Proc. ICCV.*, Oct. 2019.

[66] Y. Li, and S. Lyu. "Exposing deepfake videos by detecting face warping artifacts," in *Proc. CVPR. Workshops*, Jun. 2018, pp. 46-52.

[67] Y. Li, X. Yang, B. Wu, and S. Lyu. (Jun. 2019). "Hiding Faces in Plain Sight: Disrupting AI Face Synthesis with Adversarial Perturbations." [Online]. Available: https://arxiv.org/abs/1906.09288

[68] X. Lin, J. Liu, and X. Kang, "Audio Recapture Detection With Convolutional Neural Networks," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1480-1487, Aug. 2016.

[69] J. Fridrich, and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868-882, Jun. 2012.

[70] T. Pevny, P. Bas and J. Fridrich. "Steganalysis by Subtractive Pixel Adjacency Matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215-224, Jun. 2010.

[71] X. Kang, M. C. Stamm, A. Peng and K. J. R. Liu. "Robust Median Filtering Forensics Using an Autoregressive Model," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 9, pp. 1456-1468, Sep. 2013.

[72] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. "Backpropagation applied to handwritten zip code recognition," *Neural comput.*, vol. 1, no. 4, pp. 541-551, 1989.

[73] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. "Efficient backprop," in *Neural Networks: Tricks of the Trade.* New York, NY, USA: Springer, 2012, pp. 9-48.

[74] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121-2159, Feb. 2011.

[75] G. Huang, Z. Liu, L. v. d. Maaten and K. Q. Weinberger. "Densely Connected Convolutional Networks," in *Proc. CVPR.*, Jul. 2017, pp. 2261-2269.

[76] K. Simonyan, and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *Proc. ICLR.*, May. 2015, pp. 1-14.

[77] M. Lin, Q. Chen, and S. Yan. "Network in network," in *Proc. ICLR.*, Apr. 2014, pp. 1-10.

[78] S. Ioffe, and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proc. ICML.*, Jul. 2015, pp. 448-456.

[79] Z. Liu, P. Luo, X. Wang, and X. Tang. "Deep learning face attributes in the wild," in *Proc. ICCV.*, Dec. 2015, pp. 3730-3738.

[80] F. Chollet. "Xception: Deep Learning With Depthwise Separable Convolutions," in *Proc. CVPR.*, Jul. 2017, pp. 1251-1258.

[81] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. "Learning Rich Features for Image Manipulation Detection," in *Proc. CVPR.*, Jun. 2018, pp. 1053-1061.

[82] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 675-678.