

# Fake face detection via adaptive manipulation traces extraction network

Zhiqing Guo<sup>1</sup> Gaobo Yang<sup>1</sup> Jiyou Chen<sup>1</sup> Xingming Sun<sup>2</sup>

<sup>1</sup>Hunan University <sup>2</sup>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 such as with further compression, blurring, scaling, etc. Due to the relatively fixed structure, convolutional neural network (CNN) tends to learn image content representations. However, CNN should learn subtle manipulation traces for image forensics tasks. Thus, we propose an adaptive manipulation traces extraction network (AMTEN), which serves as pre-processing to suppress image content and highlight manipulation traces. AMTEN exploits an adaptive convolution layer to predict manipulation traces in the image, which are reused in subsequent layers to maximize manipulation artifacts by updating weights during the back-propagation pass. A fake face detector, namely AMTENnet, is constructed by integrating AMTEN with CNN. Experimental results prove that the proposed AMTEN achieves desirable pre-processing. When detecting fake face images generated by various FIM techniques, AMTENnet 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, manipulation traces extraction.

## I. INTRODUCTION

FACE images contain rich and intuitive personal identity information, which make them be commonly used for biometric authentication such as identifying individuals. However, face images also 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 [18], [27], [28] and computer graphics (CG) based methods [48] have made facial image manipulations (FIM) reach a photo-realistic level. This opens the door to a variety of face image applications such as interactive game, movie industry and photography. Nevertheless, FIM might 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<sup>1</sup>. 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.

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 [24], or replacing one face with the other one via FaceSwap [30] or DeepFake [29]. Expression manipulation refers to generating face images with specific expressions [27], or transferring facial expression from the source actor to the target face [48]. For face attribute transfer, it refers to changing the styles of face images, such as age, gender, hair color, etc [6]. In recent years, face identity manipulation has made great progress. The state-of-the-art works such as PGGAN [25] and StyleGAN [26] can synthesize hyper-realistic fake face images with the resolution up to  $1024 \times 1024$ . Recent expression manipulation techniques also generate fake face images without leaving any perceptible artifacts. Several generative models including GANimation [43] and Glow [27] were proposed for expression manipulation with photo-realistic effects. Face2Face, which is a well-known Computer Graphics (CG)-based method, animates well the facial expression of the target video from a source actor [48]. For face attribute transfer, there also exist some generative models such as StarGAN [6], and CycleGAN [57] which change facial attributes. Fig. 1 shows some examples of face images. It is difficult for human eyes to differentiate those fake face images generated by various FIM techniques from real images. As we know, face image is an important form of non-verbal communication, from which we can perceive true information. When face images are maliciously manipulated, it might bring serious influence to people, especially politician and celebrities. 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 various FIM techniques, fake face image detection is lagging far behind. Most existing works were proposed to expose some specific FIM technique [13], [20], [31], [39], just providing binary classification about the trustworthiness of face images. Only a few works attempted to expose multiple FIM forgeries [22]. Some works have studied the influence of post-processing [8], [37], [40], yet they have not fully addressed fake face detection under complex scenarios. Actually, face images are inevitably compressed or resized before spreading over social media, and possible post-processing operations might include JPEG compression (JP), Scaling (SC), Gaussian Blur (GB), Mean Filtering (ME) and Median Filtering (MED). When

<sup>1</sup><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.

the pre-trained detectors are detecting face images with unknown post-processing, 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 image detector.

The conventional image forensics framework is made up of feature extraction and classification [14], [55]. The extracted features are usually hand-crafted, which leads to poor generalization capability. In recent years, Convolutional Neural Network (CNN) has provided us an alternative for feature learning and classification automatically [53]. Instead of learning content representation for image classification tasks, CNN should learn discriminative features from subtle manipulation traces for image forensics [3]. Though CNN has achieved desirable accuracies when detecting image forgeries such as copy-move and JPEG recompression [5], 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 [3], to address this issue. Note that the first convolution layer, which is called the constrained convolution (Constrained-Conv) layer, extracts low-level manipulation traces for image forgery detection. In essence, the Constrained-Conv layer resets specific coefficients of the kernels after each iteration. Actually, since the extracted manipulation traces are fragile, they might be lost after passing through many layers. Though MISLnet provides some insights to improve the CNN model, there remain some open questions. First, is it the best way to reset some specific coefficients in the Constrained-Conv layer after each iteration? Second, can the low-level manipulation traces be reused to improve the performance of the model?

To address these questions, we are motivated to propose an Adaptive Manipulation Traces Extraction Network (AMTEN), which serves as pre-processing to suppress the side effects of image content. For existing image forensics works, the general pipeline is to predict manipulation traces and extract features from them for classification [42]. Instead, AMTEN outputs manipulation traces that can expose face image forgery, but the traces are obtained by subtracting the original image from

the feature map. Note that AMTEN is different from ResNet [21]. ResNet generates an intermediate feature representation map by using the sum of the input feature map and residual block output, without extracting the manipulation traces. However, AMTEN extracts the manipulation traces by using the difference between the input image and the output feature map, which adapts well to capture the manipulation traces for tampering detection tasks. The extracted traces are reused in AMTEN to avoid information loss. A fake face detector, namely AMTENnet, is proposed to detect multiple FIM forgeries. The main works and contributions are summarized as follows.

- A pre-processing module, namely AMTEN, is specially designed for the CNN-based face image forensics. Quite different from the fixed predictors in existing works, AMTEN predicts manipulation traces adaptively during back-propagation. AMTEN provides more discriminative manipulation traces for face forensics tasks. Moreover, it might also serve as the basic manipulation traces predictor, which means that it can be transferred to the CNN-based models to detect other image forgeries.
- By integrating AMTEN with CNN, a robust fake face detector, namely AMTENnet, is constructed to expose the state-of-the-art FIM forgeries under complex scenarios. To the best of our knowledge, this is the first attempt towards the detection of multiple FIM techniques.
- By applying some post-processing operations including lossy compression, blurring and scaling to input images, we simulate practical face image forensics under complex scenarios as real as possible. To prove the effectiveness of the proposed AMTENnet, a series of experiments are conducted. It achieves higher detection accuracy than existing works. In addition, we explore the way to improve the generalization ability.

The remainder of this paper is organized as follows. Section 2 summarizes the related works on face image forensics. Section 3 presents the AMTENnet for fake face detection. Section 4 reports the experimental results and analysis, and we conclude in Section 5.

## II. RELATED WORK

Machine learning has been widely-used in fake face image detection [51]. To expose the face-swapping forgery, Zhang et al. [54] constructed a feature set of bag of words, which provides distinguishable features into SVM for binary classification. To detect the Face2Face reenacted facial expression forgery in videos, Guo et al. [20] exploited both texture-based moment features and optical flow-based motion features. To expose the synthesized face images by GAN, Li et al. [31] defined a similarity index by Chi-square distance to model the disparities in color components. In addition, Agarwal et al. [2] proposed a deepfake face video detection approach by exploiting the correlation between facial expressions and movements. However, these machine learning based works usually have poor generalization capability.

Considering the particularity of face images, some methods exploited the biological inconsistency between real and fake

faces. Li et al. [33] proposed to expose DeepFake videos by detecting the rate of eye blinking. Matern et al. [38] 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. [7] presented a FakeCatcher to detect inauthentic portrait video by exploiting the biological signals of facial areas. Yang et al. [50] 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 also be invalidated.

Some works have addressed face image forensics from new perspectives. Xuan et al. [49] improved the detector’s generalization ability by adding noises in the training stage. Cozzolino et al. [9] addressed the forensics transfer issue among different FIM techniques. Dang et al. [12] considered the issue of imbalanced samples. Yu et al. [52] proposed to discriminate fake face images synthesized by different GANs by using their inherent fingerprints. Considering the time-consuming training of DeepFake, Li et al. [34] simulated the DeepFake-generated negative samples via simple image post-processing such as GB. Li et al. [35] also disrupted AI face synthesis with imperceptible adversarial perturbations. These efforts provide various insights to promote the development of face forensics towards universal forensics.

In recent years, CNNs have been widely used in image forensics tasks due to their superior performance [9]. These CNN models can be divided into three categories: (1) Stacking standard CNN modules for a specific image forensics task [11], [13]; (2) Using hand-crafted residual features extracted by either high pass filter or spatial rich model (SRM) for steganalysis, which are then input into CNN for image forensics [39], [56]; (3) Improving the form of the convolution layer, such as the Constrained-Conv layer [3], to force CNN to directly learn features from tampering traces. However, the existing face image forensics works only exploited either standard CNN or hand-crafted features for forensics. These works have no explicit restrictions on the convolution layer to learn features from tampering traces. For example, Afchar et al. [1] 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. Dang et al. [11] designed a customized CNN to detect face images generated by PGGAN [25] and BEGAN [4]. Li et al. [32] proposed an image representation method, which can decompose the blending boundary of the face image to expose fake face images. Dang et al. [10] utilized the attention mechanism to highlight manipulation region and improve feature representation for face forensics tasks. Mo et al. [39] introduced high pass filter into CNN to identify the PGGAN-generated faces. Gera et al. [19] also proposed a temporal-aware pipeline to expose deepfake videos, which achieves an accuracy up to 97%. In general image forensics, MISLnet is the only work that learns features from tampering traces by limiting convolution layer. Inspired by MISLnet, we design a specific pre-processing module based on the convolution layers, namely AMTEN, to predict manipulation traces. We also propose a robust fake face detector, namely AMTENnet, to learn discriminative features from manipulation traces.

### III. PROPOSED AMTENNET MODEL

#### A. AMTEN

Some existing works such as SRM [15] and SPAM [41] learn features from predicted manipulation traces. They firstly generate a set of predicted pixel value via a fixed predictor  $f(\cdot)$ . Then, the manipulation traces  $T$  is obtained by subtracting the original pixel value from the predicted pixel value. That is,

$$T = f(I) - I \quad (1)$$

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

To mimic this pipeline, AMTEN is specifically designed to automatically learn manipulation traces. Then, CNN is used to learn high-level features from manipulation traces 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.

$$F_j = \sum_{i=1}^m I_i * \omega_{ij} + b_j \quad (2)$$

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 manipulation traces  $F_{mt}$  are obtained by

$$F_{mt} = F_j - I \quad (3)$$

Apparently, the way to obtain manipulation traces in Equation (3) is almost the same as Equation (1). We randomly initialize the coefficients of the Conv 1 layer. Then, the weights are updated by an iterative algorithm during the back-propagation pass. In this paper, we adopt stochastic gradient descent (SGD) to train the model. The rules for iterative updates are defined as follows.

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

$$\omega_{ij}^{(n)} = \omega_{ij}^{(n-1)} - \nabla \omega_{ij}^{(n)} \quad (5)$$

where  $\nabla$  is the gradient,  $\omega_{ij}^{(n)}$  is the weight of the  $i^{th}$  channel of the  $j^{th}$  convolutional kernel in the  $n^{th}$  layer,  $\varepsilon$  is the learning rate and  $E$  is the loss function. We use the momentum  $\theta_1$  and the decay  $\theta_2$  to accelerate model training. In the iterative training process, its goal is to minimize the average loss  $E$  between true label and network output to make it converge. AMTEN iteratively adjusts the weights to obtain better manipulation traces. The average loss  $E$  is defined as

$$E = -\frac{1}{x} \sum_{i=1}^x \sum_{k=1}^p L_i^{(k)} \log(y_i^{(k)}) \quad (6)$$

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  $p$  is the number of neurons in the output layer. In binary classification,  $p = 2$ , which corresponds to real face image and

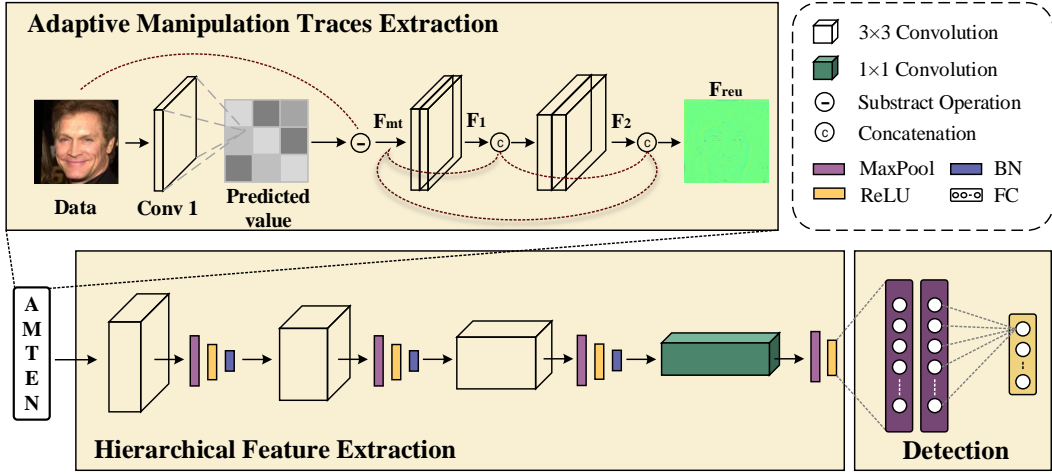


Fig. 2: The proposed AMTENnet architecture. Given an input RGB image, we use the Conv 1 in AMTEN to obtain the feature map of image. Then, the original image is subtracted from the feature map in Conv 1 to extract the low-level manipulation traces  $F_{mt}$ . Furthermore, the stable higher-level manipulation traces, namely  $F_{reu}$ , are obtained by reusing the  $F_{mt}$ . 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.

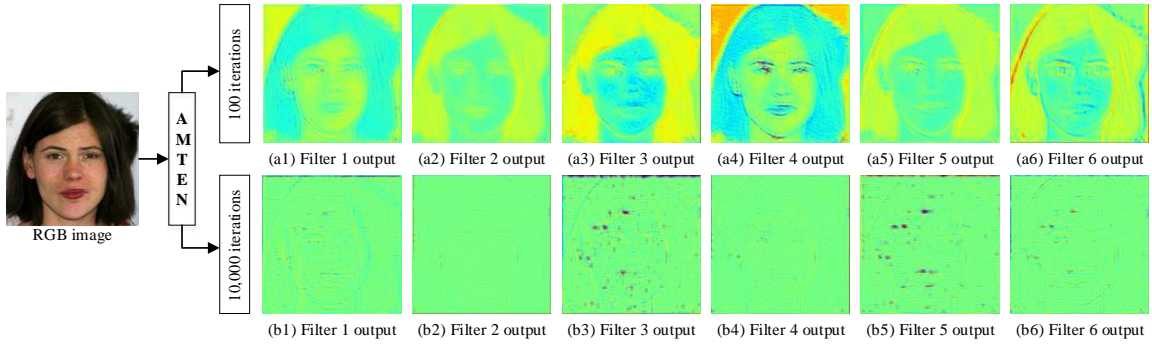


Fig. 3: The output of the six filters in AMTEN at different iterations. As the number of iterations increases, the feature map obtained by the AMTEN gradually suppresses the image content and retains the trace features.

fake face image respectively. In the multiple classification,  $p$  represents the number of categories, which more accurately corresponds to the category to which each type of image belongs.

However, the manipulation traces  $F_{mt}$  extracted by the Conv 1 layer are fragile. If they are used directly, it might still lead to unstable training. To address this issue, we borrow the idea of feature reusing from DenseNet [23]. 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  $[\beta_1, \beta_2, \dots, \beta_n]$  be the concatenation of the  $n$  feature maps. Thus, the obtained traces  $F_{mt}$  is passed into Conv 2 and Conv 3 to obtain the intermediate feature map as follows.

$$F_1 = H_{c_2, c_3}(F_{mt}) \quad (7)$$

Then, the feature map obtained by concatenating  $F_1$  and  $F_{mt}$  is passed to Conv 4 and Conv 5, which can be expressed as

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

Finally, stable manipulation traces are obtained as follows.

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

From AMTEN, stable manipulation traces are obtained by suppressing image content. Fig. 3 compares the manipulation traces obtained by different filters after 100 and 10,000 times of iterations. From it, we can observe that when AMTEN 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, AMTEN adaptively learns manipulation traces, which are more suitable for face image forensics.

### B. Network Architecture

For the proposed AMTENnet, most of the convolution layers adopt  $3 \times 3$  kernels, since it has been claimed that the  $3 \times 3$  convolutional kernel outperforms larger kernels [46]. 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 manipulation traces  $F_{mt}$  are obtained by subtracting the above feature maps from the original image. Let the input image be of size  $128 \times 128$ . As shown in Fig. 2,  $F_1$  and  $F_{mt}$  are firstly concatenated to obtain feature maps, whose dimension is  $128 \times 128 \times 6$ . 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 AMTENnet. In Section 4.3, we will also analyze the influence of the number of convolutional kernels.

AMTEN obtains desirable manipulation traces  $F_{reu}$ . Instead of directly using them as the features for face image forensics, we design a hierarchical feature extraction (HFE) module.  $F_{reu}$  is fed into the HFE module to learn high-level features. Specifically, the HFE module is made up of 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 extract 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. Different from the previous convolution layers, the Conv 9 layer adopts a  $1 \times 1$  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 non-linearity to prevent over-fitting. For the four MaxPool layers, they use the same kernel size of  $3 \times 3$ . 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 [17]. Thus, the AMTENnet model can approximate any nonlinear function. Note that these nonlinear operations including MaxPool and ReLU are not introduced into AMTEN, which prevents the learned manipulation traces from being destroyed by them. To accelerate training, the BN layer is also used in the AMTENnet model to regularize the output of the convolution layers.

Finally, the learned 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 the 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 AMTENnet. “conv”, “maxpool” and “fc” correspond to the variables in Fig. 2

| Layers              | Kernel sizes | Configuration     |         |                  |
|---------------------|--------------|-------------------|---------|------------------|
|                     |              | Kernel quantities | Strides | Output sizes     |
| Conv 1              | $3 \times 3$ | 3                 | 1       | $128 \times 128$ |
| Conv 2              | $3 \times 3$ | 3                 | 1       | $128 \times 128$ |
| Conv 3              | $3 \times 3$ | 3                 | 1       | $128 \times 128$ |
| Conv 4              | $3 \times 3$ | 6                 | 1       | $128 \times 128$ |
| Conv 5              | $3 \times 3$ | 6                 | 1       | $128 \times 128$ |
| Conv 6              | $3 \times 3$ | 24                | 1       | $128 \times 128$ |
| MaxPool             | $3 \times 3$ | \                 | 2       | $64 \times 64$   |
| Conv 7              | $3 \times 3$ | 48                | 1       | $62 \times 62$   |
| MaxPool             | $3 \times 3$ | \                 | 2       | $31 \times 31$   |
| Conv 8              | $3 \times 3$ | 64                | 1       | $29 \times 29$   |
| MaxPool             | $3 \times 3$ | \                 | 2       | $14 \times 14$   |
| Conv 9              | $1 \times 1$ | 128               | 1       | $14 \times 14$   |
| MaxPool             | $3 \times 3$ | \                 | 2       | $7 \times 7$     |
| 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 AMTENnet model. We conduct four groups of experiments. First, AMTENnet is used to detect multiple FIMs. Second, we discuss the design of the AMTENnet model. Third, some comparisons are made among AMTENnet and the state-of-the-art works. Fourth, we explore the way to improve the robustness of AMTENnet in complex scenarios.

**Datasets.** To conduct the above experiments, we firstly build a hybrid fake face (HFF) dataset<sup>2</sup>, 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 [36], high-resolution face images from CelebA-HQ [25], and face video frames from FaceForensics [44], respectively. Thus, real face images under internet scenarios are simulated as real as possible. Then, some most representative FIM techniques, which include PGGAN and StyleGAN for identity manipulation, Face2Face and Glow for face expression manipulation, and StarGAN 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 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 [52]. We mark these images with different attributes as StarGAN-generated. Table II summarizes the details of the HFF dataset. In addition, the open FaceForensics++ (FF++) dataset is used for experiments. There are 1k original video sequences, which are manipulated by four FIM techniques including Deepfakes<sup>3</sup>, Face2Face [48], FaceSwap [30] and NeuralTextures [47]. The tampered videos are further compressed with two quality levels, namely high quality videos (HQ) and low quality videos (LQ). In our experiment, we extract 50k and 10k face images from real video sequences as training sets and testing sets. For four kinds of tampered videos, we extract 12.5k and 2.5k face

<sup>2</sup><https://github.com/EricGzq/Hybrid-Fake-Face-Dataset>

<sup>3</sup><https://github.com/deepfakes/faceswap>.

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      | A generative model based identity manipulation technique.   | 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         |

images from each category as training sets and testing sets. Note that to facilitate the experiment, we only exploit two compressed versions of the FF++ dataset in the experiment. That is, the HQ and LQ datasets contain 100k face images for training and 20k face images for testing, respectively.

**Evaluation Criterion.** For image forgery detectors, they are usually evaluated by classification accuracy. In our 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 AMTEN design selections, the relative error reduction (RER) 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  $RER = (E_1 - E_2)/E_1$ .

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

- Meso-4 [1]: It exploits the mesoscopic properties of face images for facial forgery detection.
- MesoInception-4 [1]: It is an improved version of Meso-4 and has better image forgery detection performance.
- Hand-Crafted-Res [39]: 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 [3]: It exploits the Constrained-Conv layer, which suppresses image content and adaptively learns low-level residual features for universal forensics.
- XceptionNet [45]: For the FF++ dataset, XceptionNet achieved the best performance.
- Model-base: To prove the gains brought by AMTEN, it is removed from the AMTENnet model. The remaining network is called the Model-base here.
- Hand-Crafted-Res-Model-base: We use the hand-crafted feature extractor in [39] to replace the proposed AMTEN.
- Constrained-Conv-Model-base: AMTEN is replaced by the Constrained-Conv in [3].
- SRM-Model-base: AMTEN is replaced by the SRM filter kernels in [56].

**Implementation Details.** The proposed AMTENnet model is implemented under the Caffe framework. We convert all face images in the datasets into the LMDB format and then resize them into 128×128 for use in Caffe. Each forensics model has 10 training epochs. 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.

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 AMTENnet. 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%    |
|           |               |                 | *         | *             | *      | *       | *      | *        | *         |

### B. Detection of multiple FIM forgeries

Firstly, 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 AMTENnet 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

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

where  $\varepsilon_b$  is the basic learning rate,  $N$  is the fixed step size,  $\alpha$  denotes current iteration, and  $\lfloor \cdot \rfloor$  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,  $\varepsilon$  decreases periodically. The batch size is set to 64. Each training epoch requires 1,817 iterations.

The confusion matrixes of Model-base and AMTENnet 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. Note that when AMTEN is added into Model-base, which turns into AMTENnet, the false detection rate is greatly reduced.

Some comparisons are also made among different residual extraction methods, which include Hand-Crafted-Res, SRM, Constrained-Conv and AMTEN. In the experiments, they are followed with the same basic CNN model (Model-base). Table V compares the experimental results achieved by them. From it, AMTENnet achieves the highest accuracy of 98.52%, which proves that AMTEN achieves the best residual extraction. The reasons behind this are summarized as follows. Both SRM and Hand-Crafted-Res use the fixed filter to extract manipulation



TABLE V: The comparison of different residual extraction methods.

| Methods                     | Accuracy      |
|-----------------------------|---------------|
| Hand-Crafted-Res-Model-base | 97.50%        |
| Constrained-Conv-Model-base | 95.24%        |
| SRM-Model-base              | 97.49%        |
| AMTENnet                    | <b>98.52%</b> |

TABLE VI: Identification rate for different CNN models.

| Models     | Description      | Average Accuracy  | RER    |        |
|------------|------------------|---|--------|--------|
| AMTENnet   |                  | <b>98.52%</b>   | -      |        |
| Model-base | Remove the AMTEN | 96.16%  | 61.46% |        |
| Modified   | AMTEN_1          | Image data is used instead of residual features $F_{mt}$  | 97.14% | 48.25% |
|            | AMTEN_2          | $F_{mt}$ is not reused                                    | 97.46% | 41.73% |
|            | AMTEN_3          | Conv 4 = 3 and Conv 5 = 3                                 | 97.42% | 42.64% |
|            | AMTEN_4          | Conv 4 = 12 and Conv 5 = 12                               | 97.88% | 30.19% |
|            | AMTEN_5          | 5×5 convolutional kernel as the predictor                 | 97.81% | 32.42% |
|            | AMTEN_6          | Remove Conv 3 and Conv 5                                  | 98.29% | 13.45% |
| Other      | AMTENnet_7       | All pooling functions are replaced by average pooling     | 96.85% | 53.02% |
|            | AMTENnet_8       | The 1×1 convolutional kernel in Conv 9 is replaced by 3×3 | 98.36% | 9.76%  |

traces, Constrained-Conv resets specific coefficients after each iteration, whereas AMTEN adaptively updates the coefficients during the back-propagation pass to predict manipulation traces. Furthermore, AMTEN introduces feature reusing to improve detection accuracy.

### C. Design Selection of the AMTENnet Model

The CNN model has direct impacts on detection accuracy. For the AMTENnet, its AMTEN design is also important, since AMTEN learns manipulation traces for forensics. Actually, AMTEN is very flexible with the following issues to be further investigated by experiments: (1) Are the residual features  $F_{mt}$  better for forensics than image data itself? (2) Whether reusing  $F_{mt}$  will improve the AMTENnet 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 AMTEN?

To address the above issues, we have made some changes to AMTEN, which are summarized as Fig. 4. For the AMTENnet 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 AMTENnet adopts max pooling for all the pooling layers, it will be replaced with average pooling for experiments. (2) *The 1×1 convolution layer*. Its main purpose is to achieve cross-channel interaction and information integration. 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 4.2 for experiments.

Table VI reports the average detection accuracy and RER for the proposed AMTENnet model with different structures or parameters. From it, we have the following observations. First, AMTEN serves as an effective pre-processing module for the AMTENnet model, since it extracts low-level residual features suitable for 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

TABLE VII: Multiple classification identification rate of different forensics models on two datasets.

| Methods          | HFF dataset   |               |               | Average       | FF++ dataset  |               | Average       |
|------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
|                  | Raw           | JP60          | ME5           |               | HQ            | LQ            |               |
| Meso-4           | 80.76%        | 67.76%        | 62.40%        | 70.31%        | 52.92%        | 50.63%        | 51.78%        |
| MesoInception-4  | 86.40%        | 58.68%        | 77.68%        | 74.25%        | 67.38%        | 45.97%        | 56.68%        |
| Hand-Crafted-Res | 90.54%        | 73.81%        | 74.99%        | 79.78%        | 82.94%        | 64.58%        | 73.76%        |
| MISLnet          | 93.76%        | 86.32%        | 79.06%        | 86.38%        | 82.44%        | 65.61%        | 74.03%        |
| XceptionNet      | 97.17%        | 78.62%        | 90.88%        | 88.89%        | 76.30%        | 71.63%        | 73.97%        |
| AMTENnet         | <b>98.52%</b> | <b>91.02%</b> | <b>92.42%</b> | <b>93.99%</b> | <b>90.11%</b> | <b>72.14%</b> | <b>81.13%</b> |

TABLE VIII: Binary classification identification rate of different forensics models on two datasets.

| Methods          | HFF dataset   |               |               | Average       | FF++ dataset  |               | Average       |
|------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
|                  | Raw           | JP60          | ME5           |               | HQ            | LQ            |               |
| Meso-4           | 76.83%        | 62.40%        | 63.52%        | 67.58%        | 61.59%        | 61.27%        | 61.43%        |
| MesoInception-4  | 94.33%        | 73.63%        | 84.43%        | 84.13%        | 64.77%        | 77.21%        | 70.99%        |
| Hand-Crafted-Res | 89.06%        | 72.97%        | 76.88%        | 79.64%        | 79.23%        | 78.79%        | 79.01%        |
| MISLnet          | 93.71%        | 87.87%        | <b>84.76%</b> | 88.78%        | 83.84%        | 82.85%        | 83.35%        |
| XceptionNet      | 92.82%        | 74.12%        | 76.62%        | 81.19%        | 75.17%        | 75.98%        | 75.58%        |
| AMTENnet         | <b>97.66%</b> | <b>88.91%</b> | 81.36%        | <b>89.31%</b> | <b>85.14%</b> | <b>84.16%</b> | <b>84.65%</b> |

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 AMTENnet, 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

To further verify the performance of the AMTENnet model, multiple and binary classification are conducted for detection, respectively. Multiple classification detection is to identify which technique is used to manipulate the face image, which is a fine-grained recognition. Binary classification<sup>4</sup> is to judge the authenticity of face images, yet it has important advantages for future open-set recognition.

The comparisons are made among AMTENnet and some state-of-the-art works. Note that because Hand-Crafted-Res [39] and MISLnet [3] are designed for other forensics tasks, they can not converge for our forensics task. Thus, we replace the initialization method *Gaussian* with *Xavier* [16] for Hand-Crafted-Res, and we adjust the step size to 1000 for MISLnet.

We use two datasets, namely HFF and FF++ dataset, to conduct the experiments. Among them, to hide the traces left by various FIMs, JP and ME are conducted on HFF 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 and Table VIII report the detection accuracy of multiple and binary classification on two datasets, respectively. We can observe that AMTENnet can achieve the best performance whether it is multiple or binary classification task. Especially in the multiple classification task, AMTENnet outperforms baseline models by a large margin.

<sup>4</sup>In this task, the fake face images generated by all manipulation methods are classified as fake categories.

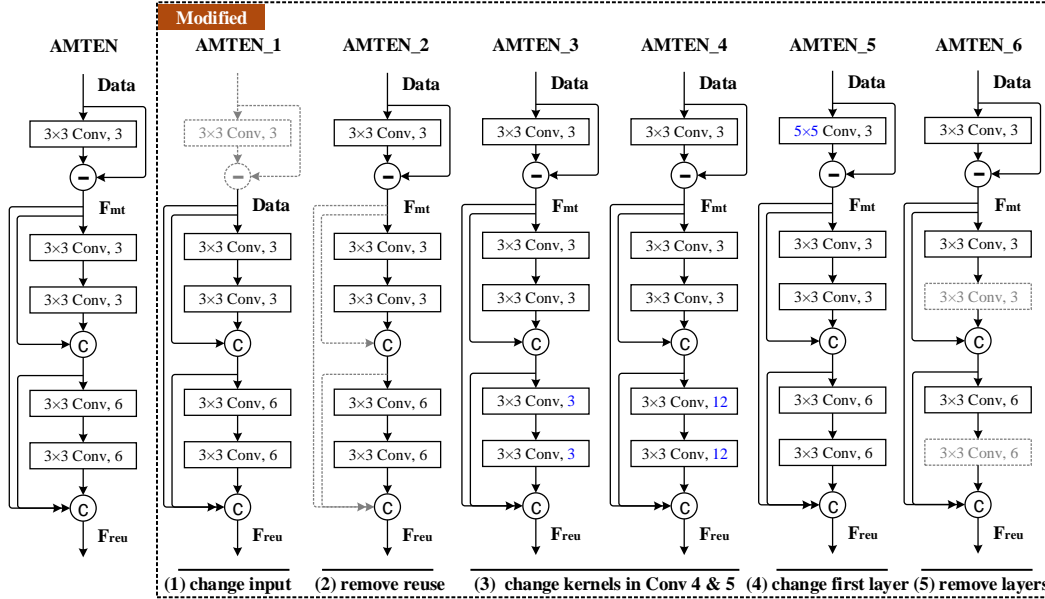


Fig. 4: The proposed AMTEN and the six modified versions. The gray dotted line indicates the removed part, and the blue font indicates the modified parameter.

TABLE IX: Parameter list of image operations.

| Image Operations    |                             | Parameters   |
|---------------------|-----------------------------|--|
| Spatial Filtering   | Mean Filtering (ME)         | kernel size: $3 \times 3$ , $5 \times 5$ , $7 \times 7$  |
|                     | Gaussian Filtering (GB)     | kernel size: $3 \times 3$ , $5 \times 5$ , $7 \times 7$ ; Standard deviation: 0  |
|                     | Median Filtering (MED)      | kernel size: $3 \times 3$ , $5 \times 5$ , $7 \times 7$  |
| 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<br>down-sampling (%): 1, 3, 5, 10, 15, 20, 25, 30, 35, 40, 45 |

TABLE X: Confusion matrix for verifying the generalization ability using AMTENnet.

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

### 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 the detector to learn fake face images under all scenarios. For the AMTENnet, its generalization capability is worthy of further investigation. To further explore the way to improve the detection robustness under complex scenarios, the following two assumptions are made. (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

TABLE XI: Identification rate of AMTENnet 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% |

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 AMTENnet are testing five types of face images. 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 still observe that the image operations with mixed parameters enable the detector to learn more discriminative features, and thus improve the generalization ability.

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 AMTENnet model



TABLE XII: Verify the generalization ability of the AMTEN-net.

| 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% |

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 remains a question: whether the detector trained by the above method can detect face images with other unknown operations? To verify this, the trained AMTENnet 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, AMTENnet 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 have photo-realistic visual qualities, which are quite challenging to be detected. Due to the relatively fixed structure, there are some limitations for the existing CNN-based works. Thus, we proposed a simple yet effective AMTEN module as pre-processing, which exploits the convolution layer to serve as the predictor to obtain image manipulation traces. The weights are updated adaptively during the back-propagation pass. In subsequent layers, the traces are reused to maximize manipulation traces. We also designed a fake face detector, namely AMTENnet, by integrating AMTEN with CNN. The manipulation traces obtained by AMTEN are fed into CNN to learn more discriminative features. A series of experiments were conducted, in which several common post-processing operations are selected to simulate the practical forensics under complex scenarios. The experimental results prove that AMTENnet achieved superior detection accuracy and desirable generalization capability. For the HFF dataset, AMTENnet improved average detection accuracy about 7.61% when compared with MISLnet. Actually, this mainly benefits from AMTEN, since it achieves better residual extraction than Constrained-conv and SRM. We have also explored the way to improve the detector's robustness. It is worthy of note that AMTEN might also serve as a basic residual predictor for other face forensic tasks.

However, the complex scenarios of practical forensics cases are simulated by several common post-processing operations.

Though they do launder the manipulation traces, they are still different from practical cases such as the AI-generated images spreading over social media such as Wechat, Twitter, Facebook and WhatsApp. For future work, we will further improve the robustness of the detector under practical scenarios by gathering real-word samples from social media.

## REFERENCES

- [1] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.
- [2] S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, and H. Li, "Protecting world leaders against deep fakes," in *CVPR Workshops*, 2019, pp. 38–45.
- [3] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2691–2706, 2018.
- [4] D. Berthelot, T. Schumm, and L. Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.
- [5] J. Chen, X. Kang, Y. Liu, and Z. J. Wang, "Median filtering forensics based on convolutional neural networks," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 1849–1853, 2015.
- [6] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.
- [7] U. A. Ciftci and I. Demir, "Fakecatcher: Detection of synthetic portrait videos using biological signals," *arXiv preprint arXiv:1901.02212*, 2019.
- [8] D. Cozzolino, G. Poggi, and L. Verdoliva, "Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection," in *Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security*, 2017, pp. 159–164.
- [9] D. Cozzolino, J. Thies, A. Rössler, C. Riess, M. Nießner, and L. Verdoliva, "Forensictransfer: Weakly-supervised domain adaptation for forgery detection," *arXiv preprint arXiv:1812.02510*, 2018.
- [10] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5781–5790.
- [11] 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," *Applied Sciences*, vol. 8, no. 12, p. 2610, 2018.
- [12] L. M. Dang, S. I. Hassan, S. Im, and H. Moon, "Face image manipulation detection based on a convolutional neural network," *Expert Systems with Applications*, vol. 129, pp. 156–168, 2019.
- [13] N.-T. Do, I.-S. Na, and S.-H. Kim, "Forensics face detection from gans using convolutional neural network," 2018.
- [14] X. Feng, I. J. Cox, and G. Doerr, "Normalized energy density-based forensic detection of resampled images," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 536–545, 2012.
- [15] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.
- [16] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [17] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [19] D. Güera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

- [20] C. Guo, G. Luo, and Y. Zhu, "A detection method for facial expression reenacted forgery in videos," in *Tenth International Conference on Digital Image Processing (ICDIP 2018)*, vol. 10806. International Society for Optics and Photonics, 2018, p. 108061J.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [22] C.-C. Hsu, C.-Y. Lee, and Y.-X. Zhuang, "Learning to detect fake face images in the wild," in *2018 International Symposium on Computer, Consumer and Control (IS3C)*. IEEE, 2018, pp. 388–391.
- [23] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [24] H. Huang, R. He, Z. Sun, T. Tan et al., "Introvae: Introspective variational autoencoders for photographic image synthesis," in *Advances in neural information processing systems*, 2018, pp. 52–63.
- [25] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.
- [26] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.
- [27] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Advances in neural information processing systems*, 2018, pp. 10215–10224.
- [28] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in neural information processing systems*, 2016, pp. 4743–4751.
- [29] P. Korshunov and S. Marcel, "Deepfakes: a new threat to face recognition? assessment and detection," *arXiv preprint arXiv:1812.08685*, 2018.
- [30] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3677–3685.
- [31] H. Li, B. Li, S. Tan, and J. Huang, "Detection of deep network generated images using disparities in color components," *arXiv preprint arXiv:1808.07276*, 2018.
- [32] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face x-ray for more general face forgery detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5001–5010.
- [33] Y. Li, M.-C. Chang, and S. Lyu, "In icu oculi: Exposing ai created fake videos by detecting eye blinking," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.
- [34] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *arXiv preprint arXiv:1811.00656*, 2018.
- [35] Y. Li, X. Yang, B. Wu, and S. Lyu, "Hiding faces in plain sight: Disrupting ai face synthesis with adversarial perturbations," *arXiv preprint arXiv:1906.09288*, 2019.
- [36] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.
- [37] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of gan-generated fake images over social networks," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 2018, pp. 384–389.
- [38] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*. IEEE, 2019, pp. 83–92.
- [39] H. Mo, B. Chen, and W. Luo, "Fake faces identification via convolutional neural network," in *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security*, 2018, pp. 43–47.
- [40] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2307–2311.
- [41] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on information Forensics and Security*, vol. 5, no. 2, pp. 215–224, 2010.
- [42] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Transactions on signal processing*, vol. 53, no. 2, pp. 758–767, 2005.
- [43] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 818–833.
- [44] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics: A large-scale video dataset for forgery detection in human faces," *arXiv preprint arXiv:1803.09179*, 2018.
- [45] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1–11.
- [46] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [47] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–12, 2019.
- [48] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2387–2395.
- [49] X. Xuan, B. Peng, W. Wang, and J. Dong, "On the generalization of gan image forensics," in *Chinese Conference on Biometric Recognition*. Springer, 2019, pp. 134–141.
- [50] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8261–8265.
- [51] X. Yang, Y. Li, H. Qi, and S. Lyu, "Exposing gan-synthesized faces using landmark locations," in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019, pp. 113–118.
- [52] N. Yu, L. S. Davis, and M. Fritz, "Attributing fake images to gans: Learning and analyzing gan fingerprints," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7556–7566.
- [53] W. Yu, X. Sun, K. Yang, Y. Rui, and H. Yao, "Hierarchical semantic image matching using cnn feature pyramid," *Computer Vision and Image Understanding*, vol. 169, pp. 40–51, 2018.
- [54] Y. Zhang, L. Zheng, and V. L. Thing, "Automated face swapping and its detection," in *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*. IEEE, 2017, pp. 15–19.
- [55] X. Zhao, Y. Lin, and J. Heikkilä, "Dynamic texture recognition using volume local binary count patterns with an application to 2d face spoofing detection," *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 552–566, 2017.
- [56] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1053–1061.
- [57] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.