Low Resolution Facial Manipulation Detection

Xiao Han[1], Zhongyi Ji[1], and Wenmin Wang[2]
[1]School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen, China
[2]School of International Institute of Next Generation Internet, Macau University of Science and Technology, Macau, China
hanxiao18@pku.edu.cn, jizhongyi@pku.edu.cn, wmwang@must.edu.mo

***Abstract*—Detecting manipulated images and videos is an important aspect of digital media forensics. Due to severe discriminative information loss caused by resolution degradation, the performance of most existing methods is significantly reduced on low resolution manipulated images. To address this issue, we propose an Artifacts-Focus Super-Resolution (AFSR) module and a Two-stream Feature Extractor (TFE). The AFSR recovers facial cues and manipulation artifact details using an autoencoder learned with an artifacts focus training loss. The TFE adopts a two-stream feature extractor with key points-based fusion pooling to learn discriminative facial representations. These two complementary modules are jointly trained to recover and capture distinctive manipulation artifacts in low resolution images. Extensive experiments on two benchmarks including FaceForensics++ and DeepfakeTIMIT, evidence the favorable performance of our method against other state-of-the-art methods.**

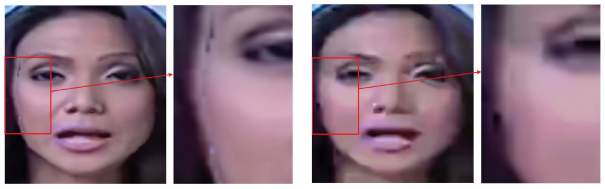
***Index Terms*—Manipulation Detection, Low Resolution, Autoencoder, Key points-based Fusion Pooling**

I. Introduction

Deepfake phenomenon has been a major concern in digital image forensics. Several available methods can be used to translate face or facial movement in real time [1], and with the rapid advances in deep learning, it has enabled non-professional users to easily create fake videos of people or to generate realistic facial image forgeries from a real video. In order to prevent such techniques from being used for improper purposes, detecting these AI-manipulated facial images or videos has become an problem in need of an urgent solution.

Traditional detection approaches tend to employ hand-crafted descriptors, such as LBP, HOG or SIFT, to extract the distinguishing features (e.g., texture information, noise artifacts, and so on) of the real and fake facial images and then use binary classifiers to decide whether an image has been manipulated or not [2].

Being able to learn more powerful and high-level features than hand-crafted features, convolutional neural networks (CNNs) have been widely applied to detect manipulated facial images. Some approaches [3] leverage the pre-trained CNNs as feature extractors and also employ various types of auxiliary information to improve detection performance, such as face depth maps [4] and remote photoplethysmography signals (rPPG) [5]. Other approaches focus on developing well-designed networks or customized components. [6] proposed a two-stream network to extract global and local facial features, respectively. [7] designed a Y-shaped autoencoder that used the multi-task learning to simultaneously detect manipulated images and locate the manipulated regions. [8] proposed a sequence learning approach and employed Long Short-Term Memory (LSTM) for detecting eye blinking, which was not reproduced well in fake videos. With such great progress, CNNs-based methods have achieved remarkable performance on some public benchmarks and outperformed their human counterpart's performance.



(a) A high resolution image. (b) A low resolution image.

Fig. 1. Example fake images from Face2Face dataset [1] show that manipulation artifacts (facial edges) tend to disappear due to resolution degradation.

However, there are still some limitations for existing detection methods, such as exhibiting poor performance on low resolution images. Several previous works [1] have demonstrated that resolution degradation has a serious negative impact on facial manipulation detection task. On the one hand, resolution degradation leads to a poor visual appearance and discriminative information loss and as a result, the global features extracted by CNNs tend to be less discriminative and representative for potential manipulation detection. On the other hand, some manipulation methods only partially manipulate real facial images, such as mouth reenactment. As shown in Fig.1, tiny manipulation artifacts tend to disappear as resolution decreases, leading to difficulties for learning the difference between real and fake facial images.

To address these problems, we propose a deep network consisting of two modules as shown in Fig.2. Our contributions can be summarized as follows:

- *Artifacts-Focus Super-Resolution:* The AFSR module upscales the resolution of input images using an autoencoder with residual connections. Different to the general SR module, the AFSR is trained with the artifacts focus loss, which focuses more on recovering manipulation artifact details.
- *Two-stream Feature Extractor:* To capture tiny manipulation artifacts, error images are used as auxiliary information, which are generated by an Error Level Analysis (ELA). The TFE adopts a two-steam network to extract

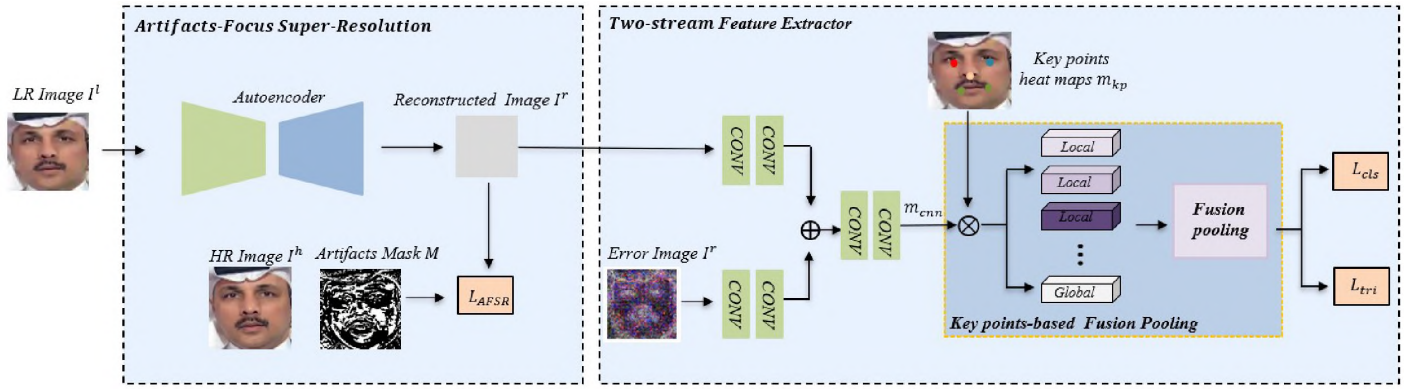


Fig. 2. The architecture of our network, which consists of two modules: Artifacts-Focus Super-Resolution and Two-stream Feature Extractor.

and fuse the features of reconstructed images and their corresponding error images, respectively.

- *Key points-based Fusion Pooling:* We extract semantic features of facial key-point regions and aggregate them with global features by fusion pooling to obtain discriminative information. Experiments show that key points-based fusion pooling is better than normal global average pooling (GAP) or global max pooling (GMP).

II. Proposed Methods

In this section, we will introduce our proposed method, including ASFR and TFE modules.

A. Artifacts-Focus Super-Resolution

The ASFR module is built to apply super-resolution to low resolution inputs, and recover lost information due to resolution degradation, e.g., facial cues and manipulation artifacts. The ASFR is implemented based on the autoencoder architecture. We chose the ResNet18 as the encoder, followed by a decoder which consists of five residual connected decoder blocks. In each residual-connected decoder block, the feature map from the previous layer is upsampled by a nearest-neighbor interpolation, following which we add two 3×3 convolution, batch norm and ReLU layers. Residual connections can preserve the visual cues in the previous layers, and hence help to enhance the quality of reconstructed images.

Pixel-wise Mean Square Error (MSE) is commonly used for SR model training. Simply minimizing the MSE may not be optimal for manipulation detection, because it does not differentiate between manipulation artifacts and normal facial features. To recover more manipulation artifacts on facial images, we propose the artifacts focus loss L_{ASFR}, i.e.,

$$L_{MSE} = \frac{1}{N}\sum_i \left\| I_i^r - I_i^h \right\|_2^2 \quad (1)$$

$$L_{AFSR} = \frac{1}{N}\sum_i \left\| M \otimes \left(I_i^r - I_i^h \right) \right\|_2^2 + \lambda_1 L_{MSE} \quad (2)$$

where $\otimes$ denotes element-wise multiply and M is a manipulation mask, I_i^l, I_i^r, I_i^h denote a low resolution image, a reconstructed image and a high resolution image, respectively.

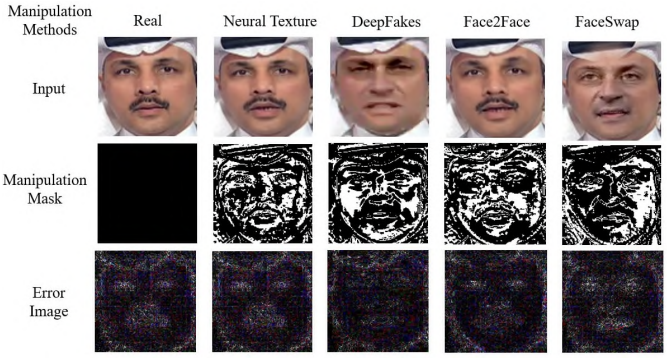


Fig. 3. Error images and manipulation mask of a real image and its corresponding four type of manipulated images.

In the binary manipulation mask M, 1 and 0 denotes manipulated and non-manipulated regions, respectively. Zero-maps is used as M for real facial images. For manipulated images, the absolute pixel-wise difference in the RGB channels are firstly computed between fake images and their corresponding real images. Then the difference images are converted into gray scale, and then a constant threshold is used for obtaining the binary manipulation mask as M, as shown in Fig.3.

B. Two-stream Feature Extractor

Two-stream feature extractor firstly applies two streams of convolution layers to extract feature maps for the reconstructed images I^r and their corresponding error images I^e, respectively. Then the two feature maps are fused through subsequent convolution layers to generate final discriminative output m_{cnn}. Error images I^e, which are generated by error level analysis, can show distinctive response on manipulated regions. The two-stream design of the feature extractor enables networks to share and aggregate information of two inputs and thereby alleviates the problem of insufficient texture features in low resolution images.

It has shown in previous works that some manipulation methods in particular can leave distinct artifacts on apecific key points regions, such as the mouth and eyes regions. Therefore, instead of using global average pooling (GAP)

and global max pooling (GMP) to obtain global features, we employ key points-based fusion pooling on m_{cnn}.

Key points-based fusion pooling extracts semantic features of facial key points regions and fuses them to capture manipulation artifacts in specific key point regions. Specifically, given feature maps m_{cnn} and key points heat maps m_{kp}, we can obtain a group of semantic local features of key points regions $\{p_k\}$ and a global feature q_1 through a element-wise multiply $\otimes$ and a global average pooling $g(\cdot)$. These procedures can be formulated in Eq.(3) and Eq.(4), where K is the key-point number.

$$\{p_k\}_{k=1}^{K} = g\left(m_{cnn} \otimes m_{kp}\right) \tag{3}$$

$$q_1 = g\left(m_{cnn}\right) \tag{4}$$

In order to capture tiny manipulation artifacts, fusion pooling is applied to fuse above local and global features. As shown in Fig.4, we firstly perform average and max pooling with all part-level features $\{p_k\}$, denoted by p_{avg}, p_{max}, respectively. Then we compute contrastive features $p_{\text{avg\_cont}}$, $p_{\text{max\_cont}}$ by subtracting p_{avg}, p_{max} and q_1, respectively. The bottleneck layers are applied to reduce the number of channels of $p_{\text{avg\_cont}}$ and $p_{\text{max\_cont}}$ from C to $\frac{C}{2}$, denoted by $\hat{p}_{\text{avg\_cont}}$, $\hat{p}_{\text{max\_cont}}$. Formally, we obtain a discriminate global feature $\hat{q}$ as follows:

$$\hat{q} = q_1 + C(\hat{p}_{\text{avg\_cont}}, \hat{p}_{\text{max\_cont}}) \tag{5}$$

where $C(\cdot)$ denotes concatenation operation. The features $\hat{q}$ is based on global features q_1, and aggregates the complementary information from the contrastive feature $\hat{p}_{\text{avg\_cont}}$, $\hat{p}_{\text{avg\_cont}}$. Finally a fully convolution (FC) layer is trained on $\hat{q}$ to predict the probability of the image being manipulated.

In TFE module, we utilize classification and triplet losses as in Eq.(6). The cross entropy loss is applied as the classification loss and the triplet loss is designed as an implicit supervision to promote intra-class compactness and inter-class separability at the feature level. Here, y_i is the probability of $\hat{q}_i$ be classified as fake, m is a margin, $d_{\hat{q}_i,\hat{q}_j}$ is the distance between a L_2 normalizated positive pair $(\hat{q}_i, \hat{q}_j)$ from the same category, and the negative pair $(\hat{q}_i, \hat{q}_k)$ is from different category.

$$\begin{aligned} L_{TFE} &= \frac{1}{N}\sum_i L_{cls}(\hat{q}_i) + L_{tri}(\hat{q}_i) \\ &= \frac{1}{N}\sum_i -logy_i + \max\left(d_{\hat{q}_i,\hat{q}_j} - d_{\hat{q}_i,\hat{q}_k} + m, 0\right) \end{aligned} \tag{6}$$

III. Experiments

A. Databases

FaceForensics++ [9]: This dataset contains 1000 real videos and 4000 fake videos which are generated by four types of manipulation methods: Face2Face (facial transferring), FaceSwap (graphics-based manipulation), DeepFakes (deep-learning-based manipulation) and Neural Textures (face reconstruction). Each type of dataset is split into 720 videos for training, 140 for validation, and 140 for testing. Three levels of compression based on the H.264 codec are used: no compression, easy compression (quantization = 23), and strong compression (quantization = 40), corresponding to raw images (Raw), high resolution images (HR), and low resolution images (LR). We employ Raw images and their corresponding manipulation masks as the supervision of AFSR during training phase.

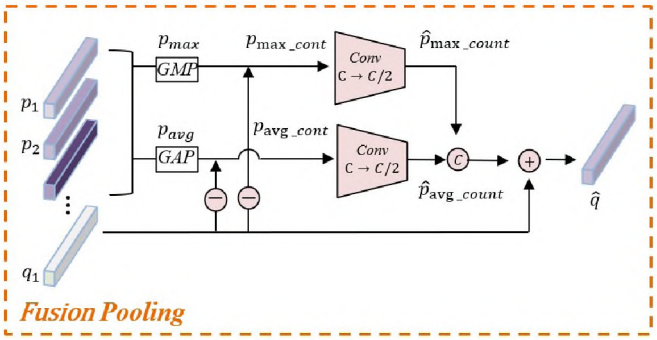


Fig. 4. The architecture of fusion pooling. The GAP and GMP denote global average pooling and global max pooling operation respectively.

DeepfakeTIMIT [10]: This dataset contains two sets of fake videos which are made using a lower quality (LQ) model and higher quality(HQ) model, respectively. Each fake video set has 32 subjects, where each subject has 10 videos with faces swapped. The original videos of their corresponding 32 subjects are from the VidTIMIT dataset. We select subsets of each subject from the VidTIMIT and all fake videos from DeepfakeTIMIT for validation (10537 original images and 34023 fake images for each quality set). We employ HQ images and their corresponding manipulation masks as the supervision of AFSR during training phase.

B. Implementation details

Model Architectures. For the AFSR module, we utilize ResNet18 as the encoder, which followed by a residual-connected decoder. At the end of the decoder, a Tanh activation layer is applied to reconstruct images. For the TFE module, we also employ a two-stream resnet18 consisting of five convolution blocks as the backbone by removing last FC layer. Specifically, the weights of the latter three convolution blocks are shared. For the face key points model, we use a MTCNN to estimate five human key points including the left eye, right eye, nose, left of the mouth, right of the mouth.

Training Details. 200 frames of each training video are used for training, and 10 frames of each validation and testing video are used for validation and testing, respectively. Facial regions are cropped and resized to 225 × 225 pixels before being fed into networks. Optimization is performed using the ADAM optimizer set to the default parameters with a batch size of 64. The initial learning rate is 10^{-4} and is divided by 10 after every 10 epochs. λ_1 in L_{AFSR} is set to 0.5. m in L_{TFE} is set 0.5. The total loss is the sum of the L_{AFSR} and the L_{TFE}.

TABLE I
AUC PERFORMANCE (%) ON FACEFORENSICS++ AND DEEPFAKETIMIT DATASETS.

Methods	DeepFakeTIMIT		FaceForencis++			
	LQ	HQ	Raw	HR	LR	total
Artifacts-NN-resnet101 [11]	97.6	86.9	-	-	-	-
MesoNet [12]	87.8	68.4	95.2	83.1	70.5	83.6
XceptionNet	88.2	74.2	99.3	95.7	81.0	93.1
Bayar and Stamm [13]	-	-	98.7	83.0	66.8	87.8
Multi-task AE [7]	62.2	55.3	-	-	-	-
Ours-GAP	92.1	80.6	98.2	91.2	86.8	92.1
Ours-GMP	93.2	81.0	98.3	94.0	89.8	94.0
Ours-FP	**99.2**	**88.3**	**99.6**	**97.6**	**91.4**	**97.0**

TABLE II
MULTI-CLASS AUC PERFORMANCE (%) FOR LQ FACEFORENSICS++ DATASET.

Dataset	MesoNet [12]	FF++ [1]	RNN-CNN [14]	ours
FaceSwap	83.0	93.0	**96.3**	95.3
DeepFake	90.0	94.0	**96.9**	94.2
Face2Face	83.2	91.0	94.3	**95.2**
NeuralTexture	75.0	81.0	-	**85.3**

C. Experiments Results

Results on benchmarks. We compare the Area Under Curve (AUC) performance of our method with other state-of-the-art methods. As the results show in TABLE I, our model outperforms most listed state-of-the-art methods with 91.4% and 88.3% on both low resolution FaceForensics++ and DeepfakeTIMIT dataset. Specifically, our method has a notable improvement over the other methods on the low resolution dataset and shows a more stable performance when the resolution of images/videos changes.

Simultaneously, we also briefly compare the performance of different pooling methods in our model in TABLE I, where GAP, GMP and FP denote the global average pooling, global max pooling, key points-based fusion pooling respectively. Experiments show that key points-based fusion pooling effectively improves the performance of our model.

Results on various attacks. The results on four types of manipulated datasets are shown in TABLE II, which include Face2Face, FaceSwap, DeepFakes and Neural Textures. All experiments are employed on low resolution dataset of FaceForensics++ and our approach achieves an excellent performance on all types of manipulated video sets, showing its strong adaptability and robustness on various manipulation methods.

Results on cross-compression datasets. We also evaluate our model on cross-compression datasets. As the results show in TABLE III, the performance of the model trained on low compression drops significantly for the input of higher compression. On the contrary, the model trained on low-resolution images can generalize well to higher resolution ones. Inspired by this phenomenon, we can find a new method for training to improve the robustness of networks on various resolutions, which is worthy of future research.

TABLE III
AUC PERFORMANCE (%) ON CROSS-COMPRESSION DATASET.

Dataset	Train	Network tested on		
		Raw	HR	LR
FaceForensics++	Raw	99.64	68.34	63.41
	HR	98.14	97.56	70.32
	LR	97.24	94.10	91.36

IV. CONCLUSION

In this paper, we proposed an Artifacts-Focus Super-Resolution module and a Two-stream Feature Extractor. Two modules are jointly trained to recover and capture facial cues and manipulation artifacts on low resolution images. Experiments have demonstrated that two modules can improve the performance of low resolution facial manipulation detection. We also show the robustness of our method on detecting various types of manipulated images with different resolutions.

V. ACKNOWLEDGEMENTS

The work was supported by the National Natural Science Foundation of China and Guangdong Province Scientific Research on Big Data (No. U1611461).

REFERENCES

[1] Rossler, Andreas and Cozzolino, et al., "Faceforensics++: Learning to detect manipulated facial images," in Proceedings of the IEEE International Conference on Computer Vision, pp. 1–11, 2019.
[2] Patel, Keyurkumar and Han, et al., "Secure face unlock: Spoof detection on smartphones, " IEEE. IEEE transactions on information forensics and security, vol. 11, pp. 2268–2283, 2016.
[3] Yang, Jianwei and Lei, et al., Learn convolutional neural network for face anti-spoofing, arXiv preprint arXiv:1408.5601, 2014.
[4] Atoum, Yousef and Liu, et al., "Face anti-spoofing using patch and depth-based CNNs," in 2017 IEEE International Joint Conference on Biometrics (IJCB), 2017, pp. 319–328.
[5] Liu, Yaojie and Jourabloo, et al., "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 389–398.
[6] Zhou, Peng and Han, et al., "Two-stream neural networks for tampered face detection," in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 1831–1839.
[7] H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, Multi-task Learning For Detecting and Segmenting Manipulated Facial Images and Videos, arXiv preprint arXiv:1906.06876, 2019.
[8] Stehouwer, Joel and Dang, Hao and Liu, Feng and Liu, Xiaoming and Jain, Anil. On the Detection of Digital Face Manipulation, arXiv preprint arXiv:1910.01717.
[9] A. Rssler, D. Cozzolino and L. Verdoliva, et al., "FaceForensics++: Learning to Detect Manipulated Facial Images, " in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 1-11.
[10] Korshunov, Pavel and Marcel, Sebastien, DeepFakes: a New Threat to Face Recognition? Assessment and Detection, arXiv: Computer Vision and Pattern Recognition, 2018.
[11] Li, Yuezun and Lyu, Siwei, Exposing DeepFake Videos By Detecting Face Warping Artifacts, arXiv: Computer Vision and Pattern Recognition, 2018.
[12] Afchar, Darius and Nozick, et al., "Mesonet: a compact facial video forgery detection network," in 2018 IEEE International Workshop on Information Forensics and Security (WIFS), 2018, pp. 1–7.
[13] Bayar, Belhassen and Stamm, Matthew C, "A deep learning approach to universal image manipulation detection using a new convolutional layer," in Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016, pp. 5–10.
[14] Sabir, Ekraam and Cheng, et al., Recurrent Convolutional Strategies for Face Manipulation Detection in Videos, pp. 80–87, 2019.