Multi-Task SE-Network for Image Splicing Localization

Yulan Zhang, Guopu Zhu, *Senior Member, IEEE*, Ligang Wu, *Fellow, IEEE*, Sam Kwong, *Fellow, IEEE*, Hongli Zhang, *Member, IEEE*, and Yicong Zhou, *Senior Member, IEEE*

***Abstract*— Image splicing can be easily used for illegal activities such as falsifying propaganda for political purposes and reporting false news, which may result in negative impacts on society. Hence, it is highly required to detect spliced images and localize the spliced regions. In this work, we propose a multi-task squeeze and excitation network (SE-Network) for splicing localization. The proposed network consists of two streams, namely label mask stream and edge-guided stream, both of which adopt convolutional encoder-decoder architecture. The information from the edge-guided stream is transmitted to the label mask stream for enhancing the discrimination of features between the spliced and host regions. This work has three main contributions. First, image edges, along with label masks and mask edges, are exploited to supply more comprehensive supervision for the localization of spliced regions. Second, the low-level feature maps extracted from shallow layers are fused with the high-level feature maps from deep layers to provide more reliable feature for splicing localization. Finally, several squeeze and excitation attention modules are incorporated into the network to recalibrate the fused features to enhance the feature expression. Extensive experiments show that the proposed multi-task SE-Network outperforms existing splicing localization methods evidently on two synthetic splicing datasets and four benchmark splicing datasets.**

***Index Terms*— Image forensics, image splicing localization, multi-task learning, squeeze and excitation attention module, low-level feature fusion.**

Manuscript received July 5, 2021; revised September 4, 2021; accepted October 18, 2021. Date of publication October 27, 2021; date of current version July 5, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB1406902; in part by the National Natural Science Foundation of China under Grant 62172402, Grant 62033005, and Grant 61872350; in part by the University of Macau under Grant MYRG2018-00136-FST; in part by the Natural Science Foundation of Heilongjiang Province under Grant ZD2021F001; in part by the Tip-Top Scientific and Technical Innovative Youth Talents of Guangdong Special Support Program under Grant 2019TQ05X696; and in part by the Basic Research Program of Shenzhen under Grant JCYJ20170818163403748. This article was recommended by Associate Editor Y. Wu. *(Corresponding author: Guopu Zhu.)*

Yulan Zhang is with the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: yl.zhang@siat.ac.cn).

Guopu Zhu is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China, and also with the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: guopu.zhu@hit.edu.cn).

Ligang Wu is with the Department of Control Science and Engineering, Harbin Institute of Technology, Harbin 150001, China (e-mail: ligangwu@hit.edu.cn).

Sam Kwong is with the Department of Computer Science, City University of Hong Kong, Hong Kong, China (e-mail: cssamk@cityu.edu.hk).

Hongli Zhang is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: zhanghongli@hit.edu.cn).

Yicong Zhou is with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yicongzhou@um.edu.mo).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3123829.

Digital Object Identifier 10.1109/TCSVT.2021.3123829

I. Introduction

With the rapid development of the Internet and digital devices, images on the web have shown an explosive growth. However, digital images can be easily forged by nonprofessional users with the user-friendly editing software (including GIMP, Photoshop, Meitu, *etc.*) [1]–[3]. Image splicing is one of the most common types of image forgery, it copies a particular region of one image (called donor image) and pastes the region to another image (called host image). Before pasting, the particular region may be geometrically transformed by rotation and scaling to match with the host image. Image splicing can be potentially used for malicious intents, such as falsifying evidence in court, forging the bank electronic bills and reporting false news to the public, which may lead to serious security issues.

As mentioned above, the integrity and authenticity of images are greatly challenged by image splicing. Hence, the issue of splicing detection and localization has received considerable interest during the past few years. Image splicing detection [4]–[9] is an image-level classification problem which aims at determining whether an image has undergone splicing or not. Nevertheless, image splicing localization is a more challenging and important task than splicing detection since it can identify the spliced region in pixel level. In recent years, identifying the spliced region has attracted more and more attention. A comprehensive review on traditional methods of image splicing localization can be found in [10]. Existing splicing localization methods mainly exploit sensor pattern noise (SPN) [11]–[16], interpolation patterns of Color Filter Array (CFA) [17], [18], and JPEG compression traces [19]–[25] as forensic features. These traditional localization methods focus on one specified image attribute, which is not applicable for all splicing cases. For example, if the devices of the donor image and the host image are the same, the SPN-based methods will become invalid; besides, the JPEG compression trace-based methods can only detect the spliced images in JPEG format.

Since deep neural network has excellent performance in multimedia understanding and computer vision, such as

saliency detection and visual tracking, numerous studies have focused on splicing localization based on convolutional neural network (CNN) [26]–[32]. Salloum *et al*. [26] proposed a multi-task fully convolutional network (MFCN), which is designed for learning the spliced mask and the mask edges. But this method overlooks both the low-level features in shallow layers of the network and the supervision from the edges of the whole image. Bappy *et al*. [33] proposed a unified architecture composed of a convolutional network and a long short-term memory network (CNN-LSTM) for detecting different types of image forgeries. CNN-LSTM model tries to capture the discriminative features in boundaries shared by tampered and un-tampered regions. After observing that the performance of splicing localization depends not only on local features (*i.e.*, the edges between the spliced and host regions) but also on global features (*i.e.*, semantic information and illumination, *etc*.), Cun and Pun [27] proposed a semi-global network for splicing localization, and the localization results were post-processed by fully connected conditional random fields (CRF). It should be pointed that the methods in [33] and [27] localize spliced images in patches, thus their localization maps show obvious block artifacts. Bappy *et al*. [34] proposed a hybrid LSTM network to exploit both the frequency domain features and the spatial context information for splicing localization. Bi *et al*. [35] proposed an end-to-end ringed residual U-Net (RRU-Net) for splicing localization by exploiting residual-propagation and residual-feedback. To deal with the realistic image forgery, Wu *et al*. proposed a unified ManTra-Net [36], which can detect and localize 385 types of image manipulations. To address the problem of constrained image splicing detection and localization (CISDL), Liu *et al*. proposed an adversarial learning network [28] and an attention-aware encoder-decoder deep matching network (AttentionDM) [29]. Whereas, these two methods are studied on image pairs, which are not suitable for the splicing localization of single image. Liu and Pun [30] proposed a deep fusion network that concentrates on learning low-level features for realistic splicing localization. They made two widely used hypotheses, one is that a sensor noise difference exists between the spliced and host regions, and the other is that the JPEG compression qualities are also different between these two regions. Hence, when the host image and the donor image are the same one, this method will not perform well. Xiao *et al*. [32] proposed a splicing localization method by combining coarse to refined network (C2RNet) and adaptive clustering. C2RNet is utilized to identify the suspicious forgery regions, and adaptive clustering is applied as a post-processing technique to obtain the final localization results. But this method requires post-processing to refine the localization results. Recently, Zhuang *et al*. [37] proposed a dense fully convolutional network (DenseFCN) to detect the image manipulations performed by Photoshop tools. This method obtained an unsatisfactory performance on existing datasets such as NIST-2016 dataset.

None of the existing methods take the edge of images into consideration for splicing localization, and the localization results of the spliced edges are far from satisfactory. In addition, most of the existing methods [26], [28], [29], [33] only focus on the high-level features in deep layers and neglect the low-level features in shallow layers. To solve the above problems encountered by existing CNN-based methods, we propose a new multi-task learning network with squeeze and excitation attention modules (SEAMs) for image splicing localization. In the proposed method, the edges of spliced images, along with the ground-truth masks and the edges of masks, are exploited to guide the learning of label mask, which can take advantage of the global edge information. Next, the low-level local features generated from the shallow layers are fused with the high-level global features from the up-sampled layers in decoder. Moreover, SEAMs are incorporated into the network to automatically recalibrate the fused features. The weights of the importance are exploited to guide the network to concentrate on the useful feature and suppress the less useful feature for splicing localization.

In summary, the contributions of this work are as follows:

1) A new multi-task squeeze and excitation network (SE-Network) consisting of two encoder-decoder streams is proposed for image splicing localization. Both the image edges and the mask edges are exploited to aid the learning of spliced mask, which is especially beneficial to the spliced edge localization.
2) To extract more discriminative feature maps and improve the representational power of the proposed network, the low-level feature maps from shallow layers of the network are fused with the high-level feature maps in the decoder. The fused feature maps can improve the localization performance on spliced regions.
3) SEAMs are incorporated to recalibrate the fused feature, which can be seen as a self-attention mechanism on channels. The abundant ablation experiments have verified the effectiveness of each component of the proposed multi-task SE-Network. The experiments show that the proposed network achieves much better performance than state-of-the-art methods on two synthetic splicing datasets and four benchmark splicing datasets.

The remainder of this paper proceeds as follows. Section II introduces the preliminary knowledge of splicing localization briefly. Section III elaborates the proposed multi-task SE-Network for splicing localization. Section IV reports and analyzes the experimental results. Finally, a few concluding remarks are provided in Section V.

II. Preliminaries

In this section, we introduce the preliminary knowledge of image splicing, the localization of image splicing, and the evaluation metrics for splicing localization.

A. Image Splicing

Image splicing is one of the most popular image forgery operations in practice. We use an example, as shown in Fig. 1, to illustrate the procedure of image splicing. In this example, the bird in the donor image is first cut, and geometrically transformed by scaling, and then is pasted to the host image to generate a composite image. Post-processing operations, such

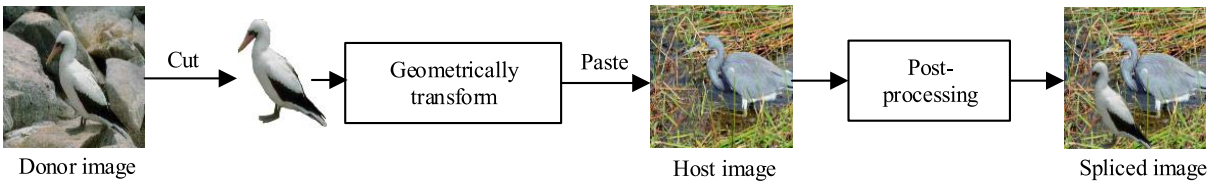

Fig. 1. Procedure of image splicing.

as Gaussian filtering and image enhancing, may be performed on the composite image to ensure that the spliced region is consistent with the host image. The post-processing on the edge of the spliced region makes the splicing localization more challenging.

B. Detection and Localization of Image Splicing

Image splicing detection is an image-level binary classification problem which can only identify whether a query image is spliced. Although splicing detection has been well addressed by many works [2], [4], [6]–[9], it is not convincing in practical application since it cannot supply the explicit forgery traces. Image splicing localization is a more challenging task than splicing detection since it aims to identify the spliced regions. During the recent years, the techniques of image splicing localization [16], [18], [26]–[30], [35], [38] have made great progress. Despite existing methods can localize spliced regions roughly, there are still several shortcomings to be overcome. For example, the edges of the image, which are valuable for the splicing edge localization, are neglected. This results in unsatisfactory localization maps on the edges of spliced regions. Besides, existing methods ignore the low-level feature in the shallow layers, and the extracted features are not discriminative enough.

C. Performance Metrics

Image splicing localization is a pixel-level binary classification problem, thus F1-score and intersection over union (IoU) are used for its performance evaluation. F1-score and IoU can be calculated, respectively, by

$$\mathrm{F1} = \frac{2TP}{2TP + FN + FP}, \tag{1}$$

and

$$\mathrm{IoU} = \frac{TP}{TP + FN + FP}, \tag{2}$$

where TP is the number of correctly detected spliced pixels, FN is the number of wrongly detected spliced pixels, and FP is the number of wrongly detected un-spliced pixels. The values of F1-score and IoU are both range from 0 to 1, and a larger value corresponds to a better performance.

III. Proposed Method

In this section, we elaborate the proposed multi-task SE-Network for image splicing localization. The architecture of the proposed network, as shown in Fig. 2, will be described as follows.

A. Multi-Task Learning Network

Multi-task learning (MTL) can improve the learning capability of deep networks by sharing the representations between multiple related tasks. This paper proposes a novel MTL network that can learn the splicing masks, the mask edges, and the image edges simultaneously. The proposed network consists of two streams, one is the edge-guided stream, and the other is the label mask stream. In the following, we will describe these two streams in detail.

1) Edge-Guided Stream: As shown in Fig. 2, the edge-guided stream is a convolutional encoder-decoder network, which has an end-to-end learning architecture. The encoder extracts discriminative feature maps from the input images, and the decoder further processes the feature maps and generates the final pixel-wise predictions. In this paper, we adopt U-Net architecture [39] as our encoder-decoder network, because U-Net is empirically found to be quite effective for image manipulation localization [40]. It is seen from Fig. 2 that the edges of the spliced images are utilized to supervise the training of the edge-guided network.

The encoder of the edge-guided stream includes four pairs of convolutional and down-sampling layers. The convolutional layers have the same kernel size of 3×3 and stride of 1, and each convolutional layer is followed by a BN layer and a Rectified Linear Unit (ReLU) layer. Inspired by [41], down-sampling is implemented by a convolutional layer with kernel size of 4×4 and stride of 2 instead of a max pooling layer. This is due to that the convolutional layer can adaptively update the parameters rather than fix it. By our experience, the convolutional layers can obtain better performance than the max pooling layers. The decoder of the edge-guided stream is composed of four pairs of up-sampling and convolutional layers. The up-sampling layers upsample the feature maps with bilinear interpolation. Hence, the size of the feature map is doubled in both width and height after up-sampling. Then an empirical convolutional layer with kernel size of 1×1 and stride of 1, which is followed by a BN layer and a ReLU layer, is added to refine the up-sampled feature maps. Nevertheless, the features may suffer from distortion after up-sampling. In order to make up for the feature loss caused by upsampling, a skip connection is built between the feature maps that have the same spatial size.

2) Label Mask Stream: The label mask stream has a similar structure to the edge-guided stream, it is also a convolutional encoder-decoder network. From Fig. 2, it is seen that both the label mask and the edges of the label mask are utilized to supervise the training of the label mask network. The

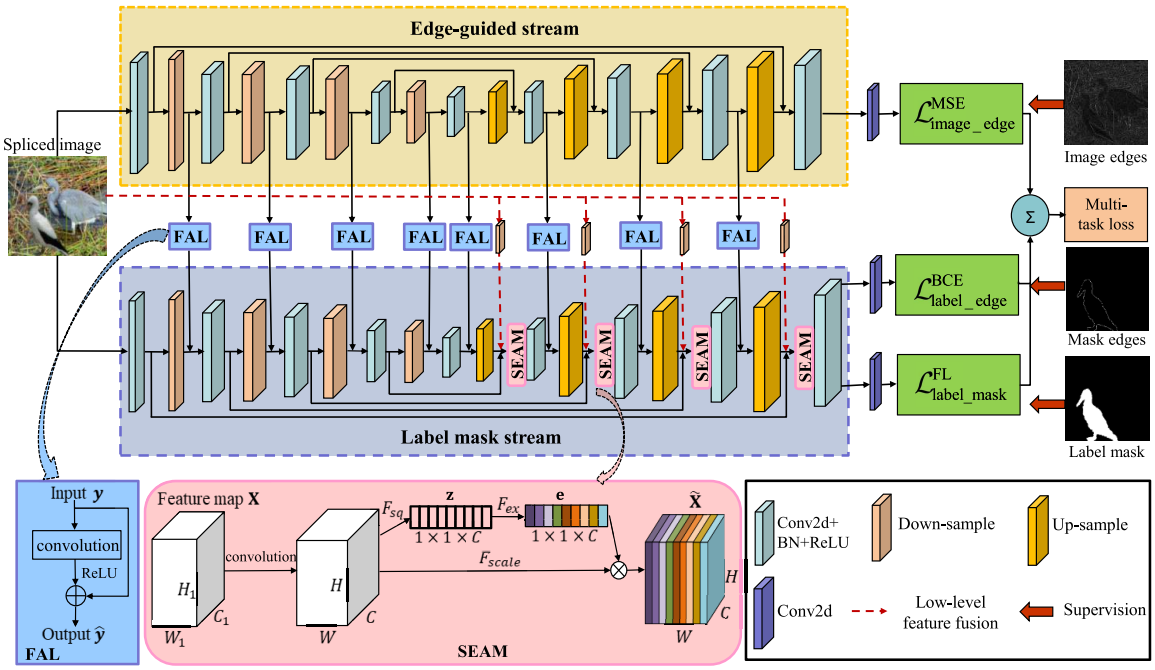

Fig. 2. Architecture of the proposed multi-task SE-Network for splicing localization.

label mask stream differs from the edge-guided stream in the following three aspects.

First, feature adaption layers (FALs) are adopted to filter out the interference of the local edge features output from the edge-guided stream. Then the filtered features are fused with the feature maps from the same level in the label mask stream. The FAL consists of a Res-Net block, which is shown in the left bottom corner of Fig. 2. We can see that the FAL is made up of an identity path and a learnable convolutional layer followed by a ReLU layer. Note that the convolutional layer has a kernel size of 1×1 and a kernel stride of 1. In IV-B.1, we will validate the use of Res-Net block. Let $\mathbf{y}$ denote the input feature map of an FAL. Then, the output of the FAL can be obtained by

$$\hat{\mathbf{y}} = \mathbf{y} \oplus \mathrm{ReLU}\left(\mathrm{C}_{1\times1}(\mathbf{y})\right), \tag{3}$$

where $\oplus$ represents the pixel-addition, $\mathrm{C}_{1\times1}$ denotes a normalized 1×1 convolution. To reduce the feature loss in fusion, concatenation operation is chosen to fuse the output of the FAL, *i.e.*, $\hat{\mathbf{y}}$, and the feature maps from the label mask stream. In this way, the features in the edge-guided stream are integrated in the mask learning process.

Second, the low-level feature maps from the shallow layers of the network are fused with the high-level feature maps from the up-sampling layers in the decoder network. The low-level feature fusion is marked with dashed red lines in Fig. 2. The fusion is introduced to take full advantage of discriminative low-level features. Generally speaking, low-level features, which are highly related to local features, refer to image details, such as the edges, corners and gradients in images. Most of the existing methods ignore these low-level features. In this paper, the low-level features are extracted directly from images with four parallel down-sampling layers. From left to right, the four down-sampling layers that perform $8\times$, $4\times$, $2\times$, and $1\times$ down-sampling are implemented by the convolutional layers with kernel size/stride of 8/8, 4/4, 2/2, and 1/1, respectively. Hence, the feature maps output from the shallow layers are of the same size with those output from the up-sampling layers in the decoder. In this way, the low-level and high-level features are integrated to enhance the high resolution representations. The effectiveness of the low-level feature fusion is investigated in IV-B.1 specifically.

Third, the fused features are coarse for splicing localization. In order to solve this problem, the fused feature maps are transmitted to an attention mechanism for recalibration, rather than are directly put into the next convolutional layer. The commonly used attention mechanisms mainly include convolutional block attention module (CBAM) [42] and squeeze-excitation attention module (SEAM) [43]. SEAM is a simple channel attention mechanism for feature maps, while CBAM is a more complex attention mechanism that utilizes both the channel and spatial attentions. Despite CBAM has shown better performance than SEAM on object detection [42], the spatial attention in CBAM may degrade the discriminability of the feature maps to some degree. On the contrary, SEAM can adaptively recalibrate the discriminative features and improve the discriminability of the fused feature maps. Thus, we adopt SEAM as the attention mechanism. SEAMs will be described in the next subsection.

B. Squeeze and Excitation Attention Module

Squeeze and excitation attention module [43] was proposed very recently. Inspired by the promising performance of SEAMs in recalibrating the discriminative features, we incorporate SEAMs into the multi-task network to reweight the fused feature maps. The architecture of SEAM is shown in

the bottom middle of Fig. 2, and will be described in the following.

First, let $\mathbf{X} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{C_1}] \in \Re^{H_1 \times W_1 \times C_1}$ denote the input feature map and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_C]$ denote the filter kernel. The input feature $\mathbf{X}$ is transformed by $\mathbf{v}_c$ ($c \in [1, C]$) as follows:

$$\mathbf{u}_c = \mathbf{X} * \mathbf{v}_c = \sum\nolimits_{i=1}^{C_1} \mathbf{x}^i * v_c^i, \tag{4}$$

where $*$ denotes convolution, $\mathbf{v}_c = [v_c^1, v_c^2, \ldots, v_c^{C_1}]$, $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C] \in \Re^{H \times W \times C}$ is the intermediate feature map. Since the result of the convolution (*i.e.*, $\mathbf{U}$) is inherently implicit and local, it is expected to provide the network with access to global information and recalibrate the filter response by utilizing squeeze and excitation.

Second, the squeeze operation F_{sq}, implemented with a global average pooling layer, is used for global information embedding. By shrinking the intermediate feature $\mathbf{U}$ through the spatial dimension $H \times W$, a statistic $\mathbf{z} = [z_1, z_2, \ldots, z_C] \in \Re^C$ is generated, *i.e.*,

$$z_c = \mathrm{F}_{sq}(\mathbf{u}_c) = \frac{1}{H \times W} \sum_{m=1}^{H} \sum_{n=1}^{W} \mathbf{u}_c(m, n). \tag{5}$$

Third, the excitation operation, denoted as F_{ex}, is utilized for adaptively recalibrating the responses of each channel. A gating mechanism with a sigmoid activation is used to implemented the squeeze operation due to two reasons: 1) the excitation operation should be flexible and be able to learn a non-linear interaction between channels; 2) in order to emphasize multiple channels, the excitation operation must learn a non-mutually-exclusive relationship. Thus, the excitation function can be formulated as

$$\mathbf{e} = \mathrm{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma\left(g(\mathbf{z}, \mathbf{W})\right) = \sigma\left(\mathbf{W}_2 \mathrm{ReLU}(\mathbf{W}_1 \mathbf{z})\right), \tag{6}$$

where σ denotes the sigmoid activation, g denotes a gating mechanism implemented with ReLU, $\mathbf{W}_1 \in \Re^{\frac{C}{r} \times C}$, $\mathbf{W}_2 \in \Re^{C \times \frac{C}{r}}$, and r denotes the dimensionality reduction ratio. In order to limit the complexity and improve the generalizability of trained model, we utilize two fully-connected (FC) layers around the ReLU layer to form a gating mechanism. The first and second FC layers are used for dimension reduction and expansion, respectively.

Finally, the output of the SEAM, denoted as $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}^1, \tilde{\mathbf{x}}^2, \ldots, \tilde{\mathbf{x}}^C]$, is obtained by rescaling $\mathbf{U}$ with the activation $\mathbf{e}$, *i.e.*,

$$\tilde{\mathbf{x}}^c = \mathrm{F}_{\mathrm{scale}}(\mathbf{u}_c, e_c) = e_c \mathbf{u}_c, \tag{7}$$

where $\mathbf{e} = [e_1, e_2, \ldots, e_C]$, $\mathrm{F}_{\mathrm{scale}}(\mathbf{u}_c, e_c)$ represents channel-wise multiplication between e_c and $\mathbf{u}_c$. By mapping the input-specific descriptor $\mathbf{z}$ to a set of channel weights $\mathbf{e}$, SEAMs intrinsically introduce dynamics condition on the input features. Hence, it can also be regarded as a self-attention mechanism that pay more attention to the useful features and suppress the less useful features according to the splicing localization task. The feature map $\tilde{\mathbf{X}}$ output from SEAMs discriminate the spliced region better than the original fused feature map $\mathbf{X}$. The effectiveness of SEAMs on splicing localization will be studied in IV-B.1 briefly.

C. Loss Function

In this subsection, a multi-task loss function is designed to evaluate the fitting degree of the network. The proposed loss consists of three parts, one loss is for the label mask, one is for the mask edges, and the other is for the edges of the whole image.

The spliced region accounts for a small part of the whole image. This will lead to imbalance in positive and negative regions. Since the easy negative regions contribute little useful learning information, the imbalance will decrease the efficiency of training. Besides, the easy negative samples may overwhelm training and degrade the models. To address the class imbalance problem, we employ focal loss (FL) [44] to reweight the loss of negative samples and positive samples. In addition, the focal loss reduces the weight of easy samples and pays more attention to the hard positives during training. For the label mask, the FL is written as

$$\begin{aligned}\mathcal{L}_{\text{label\_mask}}(\mathbf{Y}, \mathbf{P}) = &- \sum_{j=1}^{W} \sum_{i=1}^{H} \alpha (1 - P_{i,j})^{\gamma} Y_{i,j} log(P_{i,j}) \\ &+ (1 - \alpha) P_{i,j}^{\gamma} (1 - Y_{i,j}) \log(1 - P_{i,j}),\end{aligned} \tag{8}$$

where $\mathbf{Y} = [Y_{i,j}]$, $\mathbf{P} = [P_{i,j}]$; $Y_{i,j}$ and $P_{i,j}$ denote the predicted spliced label and the probability of being spliced at coordinate (i, j), respectively; α is a parameter used to balance the positive and negative regions; γ is used to balance the easy and hard classified samples. In our experiments, we empirically set $\alpha = 0.25$, $\gamma = 2$.

For the loss of mask edge, we adopt the binary cross-entropy (BCE) loss, which is expressed by

$$\begin{aligned}\mathcal{L}_{\text{label\_edge}}(\mathbf{E}, \mathbf{Q}) = &- \sum_{j=1}^{W} \sum_{i=1}^{H} E_{i,j} log(Q_{i,j}) \\ &+ (1 - E_{i,j}) \log(1 - Q_{i,j}),\end{aligned} \tag{9}$$

where $\mathbf{E} = [E_{i,j}]$, $\mathbf{Q} = [Q_{i,j}]$; $E_{i,j}$ and $Q_{i,j}$ denote the predicted label of spliced edge and the probability of being spliced edge at coordinate (i, j), respectively. Note that to deal with the imbalance between the positive (spliced) regions and the negative (un-spliced) regions, we use the focal loss as the label mask loss to down-weight the loss assigned to easily-classified samples. Whereas, the main goal of the label edge loss is to detect the label edges, and is just an auxiliary task for splicing localization. It is relatively easy for the network to obtain the label edges. Thus, the general binary cross entropy loss is suitable to be used as the label edge loss.

The minimum square error (MSE) loss is utilized for the edge prediction of whole image. The MSE loss is given as

$$\mathcal{L}_{\text{image\_edge}}(\mathbf{S}, \hat{\mathbf{S}}) = \frac{1}{H \times W} \sum_{j=1}^{W} \sum_{i=1}^{H} \left(S_{i,j} - \hat{S}_{i,j}\right)^2, \tag{10}$$

where $\mathbf{S} = [S_{i,j}]$, $\hat{\mathbf{S}} = [\hat{S}_{i,j}]$; $S_{i,j}$ and $\hat{S}_{i,j}$ denote the ground-truth label of image edge and the predicted label of image edge at coordinate (i, j), respectively. The ground-truth labels of image edges are the edges of the whole image. These image edge labels are obtained with mathematical

morphology [45]. Specifically, given an image I, its edges can be calculated by $S(I) = I - (I \ominus B)$, where $S(I)$ denotes the edge of image I, $\ominus$ denotes erosion operation, and B is a suitable structuring element. In this paper, we set B as [0 1 0; 1 1 1; 0 1 0].

Finally, the total loss of the multi-task SE-Network for splicing localization can be written as

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{label\_mask}} + \lambda_1 \mathcal{L}_{\text{label\_edge}} + \lambda_2 \mathcal{L}_{\text{image\_edge}}. \quad (11)$$

Each part of the total loss plays an important role in splicing localization, which will be validated in the ablation experiments in Section IV-B.1. Adjusting parameters are sometimes used to balance the influence of different losses in computer vision [46], [47]. The objective of our network training is to minimize the total loss $\mathcal{L}_{\text{total}}$, which, theoretically speaking, is equivalent to minimizing $\mathcal{L}_{\text{label\_mask}}$, $\mathcal{L}_{\text{label\_edge}}$, and $\mathcal{L}_{\text{image\_edge}}$, separately. Hence, the setting of the adjusting parameters λ_1 and λ_2 may not make a significant impact on our splicing localization. Furthermore, our experimental comparisons on the adjusting parameters have shown that the setting of the adjusting parameters has little impact on the localization performance, and the setting $\lambda_1 = \lambda_2 = 1$ obtains the best splicing localization performance compared with the other settings. For simplicity, we set $\lambda_1 = \lambda_2 = 1$ in our experiment.

IV. Experiments

A. Experimental Setup

The experiments are performed on Ubuntu 16.04 with GeForce GTX 1080 Ti GPU. The multi-task SE-Network for image splicing localization is implemented with the Pytorch framework [48]. The adaptive moment estimation (Adam) optimizer [49] is applied for network training. The learning rate is initially set to 1×10^{-3}, and then set to 1×10^{-4} after 30 epochs. The edge-guided and the label mask streams are trained at the same time. The two streams are interacted with each other through FALs during training. We fix the batch size to 8 and train the network for 300 epochs. Finally, the model that achieves the highest F1-score on test dataset is selected as the final model for evaluation.

1) Data Preparation: We evaluate the tested methods on four public splicing datasets, including the CASIA1.0 dataset [50], the CASIA2.0 dataset [50], the Carvalho dataset [2], and the Columbia dataset [51]. The samples in the above-mentioned datasets are all called the benchmark datasets, and the number of the benchmark samples is 5,864. Additionally, as done by the composite method in [34], the objects extracted from Microsoft coco dataset [52] are spliced onto the Dresden dataset [53] and the non-manipulated NIST'16 dataset,[1] separately. The samples generated from Dresden dataset and the NIST'16 dataset are called the synthetic datasets in this paper, and the number of the synthetic samples is 49,182. The setting of different datasets on training, testing, and validation are listed in Table I. For convenience, images from all the datasets are resized into 384×384. Some of the spliced samples and their corresponding ground-truth masks are shown in Fig. 3, where the white masks represent the spliced regions.

To evaluate the robustness against post-processing on splicing localization, the test images are post-processed by JPEG compression, image scaling, Gaussian filtering, and image sharpening, separately. The quality factor (QF) of JPEG compression is set as $QF \in \{80, 85, 90, 95, 100\}$, the scale factor (SF) for image scaling is set as $SF \in \{0.7, 0.8, 0.9, 1.2, 1.4, 1.6\}$, the standard deviation δ of Gaussian filter is set as $\delta \in \{0.5, 1.0, 1.5, 2.0\}$, and the strength parameter λ is set as $\lambda \in \{0.5, 1.0, 1.5, 2.0\}$.

2) Compared Methods: The proposed method is compared with several deep learning-based methods for a fair comparison, which are listed as follows:

1) *MFCN [26]:* An image splicing localization method based on multi-task fully convolutional network;
2) *CNN_LSTM [33]:* An image manipulated localization method based on CNN_LSTM network;
3) *SGN_CRF [27]:* A splicing localization method based on semi-global network and fully connected CRF;
4) *RRU-Net [35]:* A splicing localization method based on ringed residual U-Net;
5) *ManTra-Net [36]:* A forgery localization method based on manipulation tracing network;
6) *HLED [34]:* A forgery localization method based on hybrid LSTM and encoder-decoder network;
7) *DenseFCN [37]:* An image tampering localization method based on dense fully convolutional network.

We re-implemented MFCN according to Ref. [26]. We adopted the code released in SGN_CRF [27][2] for CNN_LSTM [33] and SGN_CRF [27]. Besides, the released code of RRU-Net [35],[3] HLED [34],[4] and DenseFCN [37][5] are also used for training and testing. In addition, the released pre-trained weights and inference code are utilized for ManTra-Net.[6]

B. Experimental Results

In this subsection, we first validate the effectiveness of each component in the proposed network. Then, the splicing localization results of the proposed method are compared with those of the compared methods. Finally, the robustness against several image post-processing operations is investigated.

1) Ablation Study: We validate the effectiveness of several important components of the proposed network, including the low-level feature fusion, the multi-task loss, and SEAMs. To verify the effectiveness of each component on splicing localization, five schemes are designed. The localization results are shown in Table II, the "Average" denotes the average localization results of all the test datasets.

Both Scheme 1 and Scheme 2 are trained with $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}} + \mathcal{L}_{\text{image\_edge}}$ and without SEAMs. The difference

[1] https://www.nist.gov/itl/iad/mig/open-media-forensics-challenge

[2] https://github.com/vinthony/image-splicing-localization
[3] https://github.com/yelusaleng/RRU-Net
[4] https://github.com/jawadbappy/forgery_localization_HLED
[5] https://github.com/ZhuangPeiyu/Dense-FCN-for-tampering-localization
[6] https://github.com/ISICV/ManTraNet

TABLE I
SETTING OF DIFFERENT DATASETS ON TRAINING, TESTING, AND VALIDATION

Dataset		Train	Validation	Test	Total_test
Benchmark datasets	Columbia	125	5	50	671
	CASIA1	341	20	100	
	CASIA2	4500	122	501	
	Carvalho	70	10	20	
Synthetic datasets	Spliced_NIST	9534	674	3367	12295
	Spliced_Dresden	24998	1786	8928	

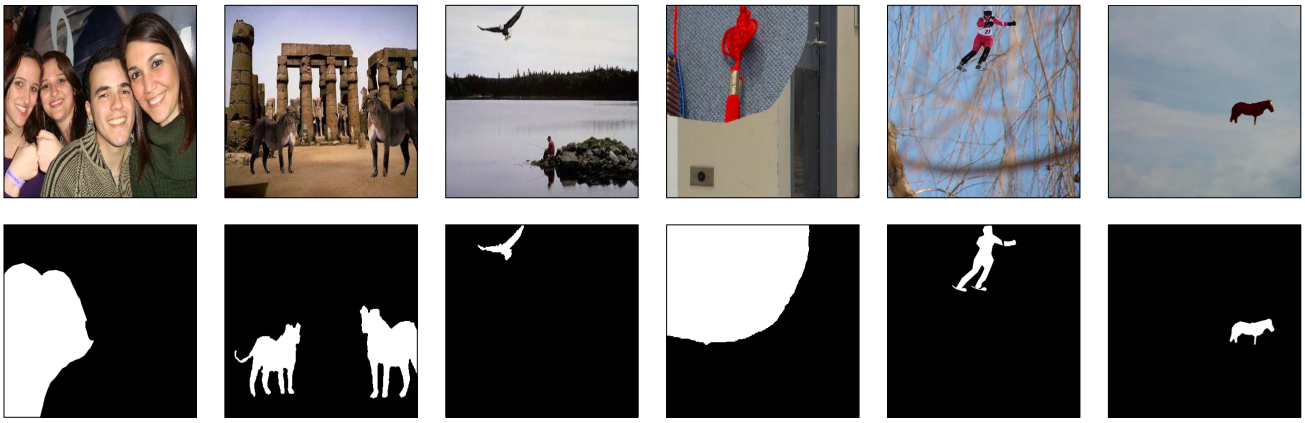

Fig. 3. Spliced samples (the first row) and the corresponding ground-truth masks (the second row) in different datasets.

TABLE II
DIFFERENT CONFIGURATIONS OF THE ABLATION STUDY AND THE CORRESPONDING LOCALIZATION RESULTS ON DIFFERENT DATASETS (IOU/F1-SCORE (%)); THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

		Scheme 1	Scheme 2	Scheme 3	Scheme 4	Proposed
Low-level feature fusion	Without	✓				
	With		✓	✓	✓	✓
Multi-task loss	$\mathcal{L}_{\text{label\_mask}}$	✓	✓	✓	✓	✓
	$\mathcal{L}_{\text{label\_edge}}$	✓	✓		✓	✓
	$\mathcal{L}_{\text{image\_edge}}$	✓	✓			✓
SEAMs	Without	✓	✓	✓	✓	
	With					✓
Dataset	Benchmark datasets	51.25/58.28	62.03/68.72	58.28/64.74	61.18/67.49	**69.34/74.91**
	Synthetic datasets	91.03/94.84	94.09/96.71	92.03/95.38	93.25/96.11	**94.64/97.09**
	Average	88.97/92.94	92.43/95.26	90.28/93.79	91.58/94.63	**93.32/95.94**

between these two schemes is that Scheme 1 is trained without low-level feature fusion, while Scheme 2 is trained with it. To validate the effect of the multi-task loss on splicing localization, Scheme 2, Scheme 3, and Scheme 4 are designed. All the three schemes are performed with low-level feature fusion and without SEAMs. Scheme 3, Scheme 4, and Scheme 2 are performed with $\mathcal{L}_{\text{label\_mask}}$, $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}}$, and $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}} + \mathcal{L}_{\text{image\_edge}}$, respectively. Finally, the proposed network is performed with low-level feature fusion, the multi-task loss $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}} + \mathcal{L}_{\text{image\_edge}}$ and SEAMs.

From Table II, the following observations can be obtained:

- *Low-Level Feature Fusion:* From the results of Scheme 1 and Scheme 2, it can be found that the use of low-level feature fusion obtains a significant improvement on localization performance. This may be due to that the low-level feature can supply more discriminative features for splicing localization. With low-level feature fusion, the average IoU/F1-score of localization results obtain an improvement of 10.78%/10.44% and 3.06%/1.87% on the benchmark datasets and the synthetic datasets, respectively. It is surprising that low-level feature fusion improves the localization results more significantly on the benchmark datasets than on the synthetic datasets. The reason for this can be explained as follows. On one hand, the number of the benchmark datasets is too small to extract discriminative feature of splicing. The low-level feature may supply more useful information for discriminating splicing to make up for the shortcomings

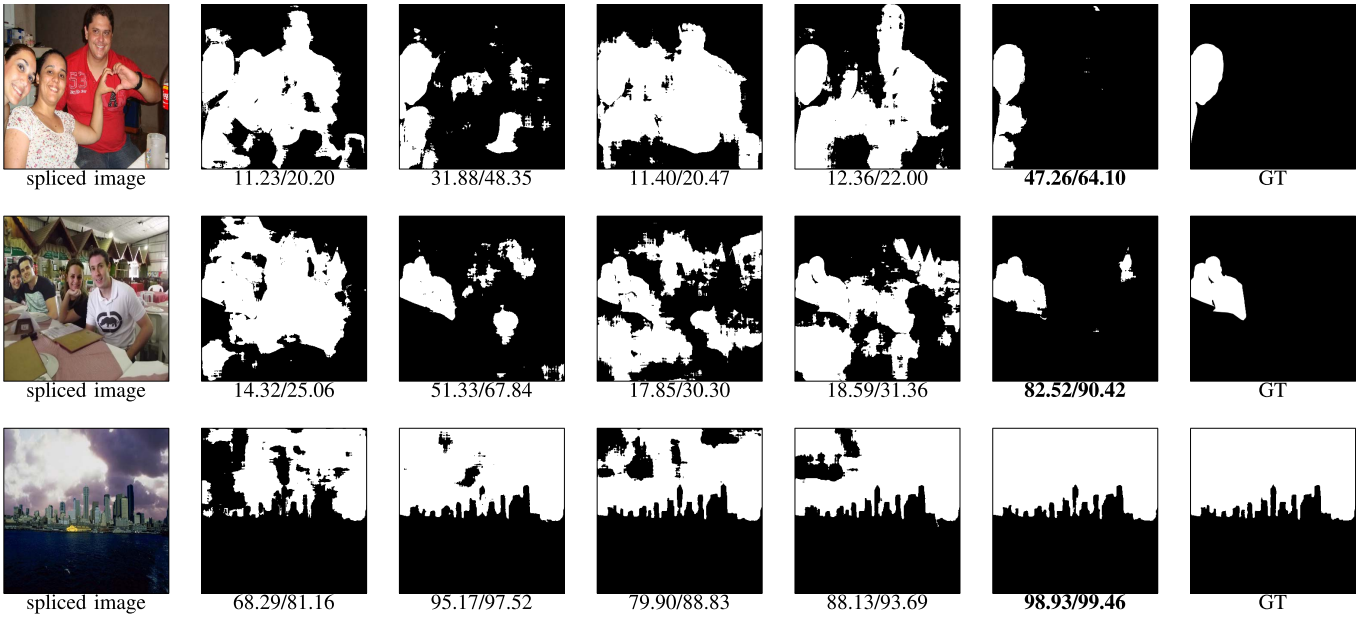

Fig. 4. Localization results of different schemes (IoU/F1-score (%)). The first column represents spliced images; the second to the sixth columns represent the localization maps obtained by Scheme 1, Scheme 2, Scheme 3, Scheme 4 and the proposed method, respectively; the last column is the corresponding ground-truth mask of the first column. The best results are highlighted in bold.

of insufficient samples in benchmark datasets. On the other hand, for the synthetic datasests, the number of synthetic datasets is more abundant, the model can get a relatively high localization rate, so the low-level feature fusion contributes less for the synthetic datasets than for the benchmark datasets.

- *Multi-Task Loss Function:* Comparing Scheme 4 with Scheme 3, it is evident that the loss $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}}$ outperforms the loss $\mathcal{L}_{\text{label\_mask}}$ by 2.9%/2.75% and 1.22%/0.73% in IoU/F1-score on the benchmark datasets and the synthetic datasets, respectively. Comparing Scheme 2 with Scheme 4, we can see that loss $\mathcal{L}_{\text{label\_mask}} + \mathcal{L}_{\text{label\_edge}} + \mathcal{L}_{\text{image\_edge}}$ can further improve the localization performance by taking the edges of images into account. Scheme 2 outperforms Scheme 4 by 0.85%/1.23% and 0.84%/0.60% in IoU/F1-score for the benchmark datasets and the synthetic datasets, respectively. The effectiveness on the localization results of the multi-task loss can be interpretated in two aspects: first, the mask edges (*i.e.*, the edges between the spliced and host regions) can be a remarkable characteristic of the spliced region, hence they can facilitate the localization of spliced regions; second, the edges of the whole image can supervise the network to learn more accurate image edges and further improve the performance of the mask edge localization.
- *Squeeze and Excitation Attention Module:* Compared with Scheme 2, the proposed method adds SEAMs after the low level-feature fusion, and obtains the best performance. The IoU/F1-score with SEAMs outperforms Scheme 2 by 7.31%/6.19% and 0.55%/0.38% for the benchmark datasets and the synthetic datasets, respectively. This result can be explained below. First, the feature maps obtained by fusing spatial low-level feature and the channel-wise up-sampled feature are coarse for localizing the spliced regions. So the fused maps are not suitable to transmit to the next layer directly. Second, SEAMs guide the network to pay more attention to the spliced regions under the supervision of the given ground-truth. Last but not the least, an SEAM is comprised of a lightweight gating mechanism, which can enhance the feature expression by explicitly modeling the interdependencies between channels.

Fig. 4 presents some localization maps of the above-mentioned schemes. Comparing the second column with the third column, we find that the low-level feature fusion can significantly improve the localization results. The comparisons among the fourth, the fifth and the third columns reveal that both the mask edges and the image edges provide stronger supervision for the splicing localization task. Furthermore, comparing the sixth column with the third column, it is seen that the scheme with SEAMs achieves evident improvements over the scheme without SEAMs. As depicted in the third and the sixth columns in Fig. 4, the results (the sixth column) obtained with SEAMs are of lower false positive rates than those (the third column) obtained without SEAMs.

To investigate the effectiveness of different attention mechanisms and different types of FAL on splicing localization, we make an experimental comparison, which is reported in Table III. In the "FAL" row of Table III, "Without" denotes that the feature maps from the edge-guided stream are directly concatenated with those from the label mask stream without using FAL; "Conv" denotes the FAL implemented with a single 1×1 convolution unit; and "Res_block" denotes the FAL implemented with a Res-Net block. Note that all these experiments are performed with low-level feature fusion and multi-task loss.

TABLE III

EFFECT OF DIFFERENT ATTENTION MECHANISMS AND DIFFERENT TYPES OF FAL ON SPLICING LOCALIZATION (IOU/F1-SCORE (%)); THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Attention mechanism	CBAM	✓			
	SEAM		✓	✓	✓
FAL	Without		✓		
	Conv			✓	
	Res_block	✓			✓
Dataset	Benchmark datasets	61.20/67.84	60.11/68.29	65.70/71.53	**69.34/74.91**
	Synthetic datasets	94.20/96.79	93.71/96.53	94.51/97.01	**94.64/97.09**
	Average	92.49/95.29	92.02/95.07	93.02/95.69	**93.32/95.94**

TABLE IV

LOCALIZATION RESULTS OF COMPARED METHODS ON DIFFERENT DATASETS (IOU/F1-SCORE (%)); THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Dataset \ Methods		MFCN [26]	CNN_LSTM [33]	SGN_CRF [27]	RRU-Net [35]	ManTra-Net [36]	HLED [34]	DenseFCN [37]	Proposed
Benchmark datasets	Columbia	52.90/62.22	15.36/27.48	17.22/25.60	66.93/73.14	32.80/47.18	17.52/27.03	23.12/32.29	**86.70/91.00**
	CASIA1	34.02/42.27	30.91/43.12	9.75/16.64	42.64/48.14	13.87/22.59	18.61/26.93	12.54/17.35	**59.24/66.69**
	CASIA2	30.43/37.55	10.56/16.66	9.09/14.50	47.52/53.33	12.61/20.09	13.35/19.40	6.88/9.83	**71.39/76.53**
	Carvalho	21.50/32.11	16.87/27.78	13.56/23.37	17.96/24.74	20.17/32.51	9.32/16.08	16.38/27.11	**25.13/35.26**
	Average_Benchmark	32.37/39.93	14.15/21.77	9.93/15.91	47.36/53.18	15.62/19.37	14.32/20.99	9.91/10.75	**69.34/74.91**
Synthetic datasets	Spliced_NIST	73.54/83.18	28.98/40.93	30.91/43.12	85.93/91.11	33.33/46.08	61.34/73.33	67.40/75.13	**93.80/96.56**
	Spliced_Dresden	77.91/86.71	37.54/51.34	38.97/52.85	88.34/93.20	33.75/47.16	66.28/77.98	77.04/83.78	**94.95/97.29**
	Average_Synthetic	76.71/85.74	35.20/48.49	36.76/50.19	87.68/92.63	33.63/46.86	64.93/76.71	74.40/81.41	**94.64/97.09**
Average		74.42/83.37	34.11/47.11	35.37/48.41	85.59/90.59	32.70/45.44	62.31/73.82	71.06/77.75	**93.32/95.94**

From the third and sixth columns in Table III, it can be seen that the network with SEAM outperforms that with CBAM by 8.14%/7.07% and 0.44%/0.30% in terms of IoU/F1-score on the benchmark and the synthetic datasets, respectively. This may be due to that the spatial attention in CBAM may degrade the discriminability of the feature maps to some degree. On the contrary, SEAM can adaptively recalibrate the discriminative features and improve the discriminability of the fused feature maps. Comparing the fourth, fifth, and sixth columns, we see that the FAL implemented with Res-block achieves the best localization results in all test cases. The underlying reason for this result can be explained as follows. First, the feature maps from the edge-guided stream are of some interference to the label mask stream. Thus, these feature maps are not suitable to be directed concatenated with those from the label mask stream. Second, the FAL implemented with a single convolutional unit can filter out some interference and improve the splicing performance to some extent. However, the convolution based FAL will cause more information loss. Third, the Res-block based FAL could filter out the interference through its convolution unit path and retain the important information of the features through its identity path.

Based on the above observations from our experimental comparisons, it can be concluded that the four components designed in the proposed network (*i.e.*, the low-level feature fusion, the multi-task loss learning, SEAMs, and the Res-block based FALs) can indeed improve the performance of splicing localization.

2) Comparison With Existing Methods: The proposed method is compared with existing splicing localization methods introduced in IV-A.2. The splicing localization results on different datasets are shown in Table IV, the "Average_Benchmark" and the "Average_Synthetic" represent the average results on the benchmark datasets and the synthetic datasets, respectively. It can be observed from Table IV that the proposed method achieves an average IoU/F1-score of 93.32%/95.94%, which is 7.73%/5.35% higher than the second-best method, *i.e.*, RRU-Net.

Fig. 5 shows some localization maps of the compared methods. From Fig. 5, we can see that MFCN [26] can roughly localize the spliced regions but with lower IoU and F1-score. The localization results of CNN_LSTM [33] and SGN_CRF [27] show apparent block artifacts due to that both the two methods process images in blocks. The localization maps of ManTra-Net [36] are very coarse with a high false negative rate. HLED [34] can only successfully identify a small part of spliced regions. DenseFCN [37] wrongly localize extra objects as spliced regions. RRU-Net [35] detects the spliced regions quite well on the whole, whereas, the edges of the spliced regions are not well localized. The proposed multi-task SE-Network can accurately localize almost all the spliced regions, it outperforms all the compared methods in both IoU and F1-score. Especially, the proposed method obtains the most accurate spliced edges among all the compared methods.

3) Robustness Against Post-Processing: To conceal the traces of splicing, the forged images may be manipulated by some post-processing operations. Next, we will investigate the

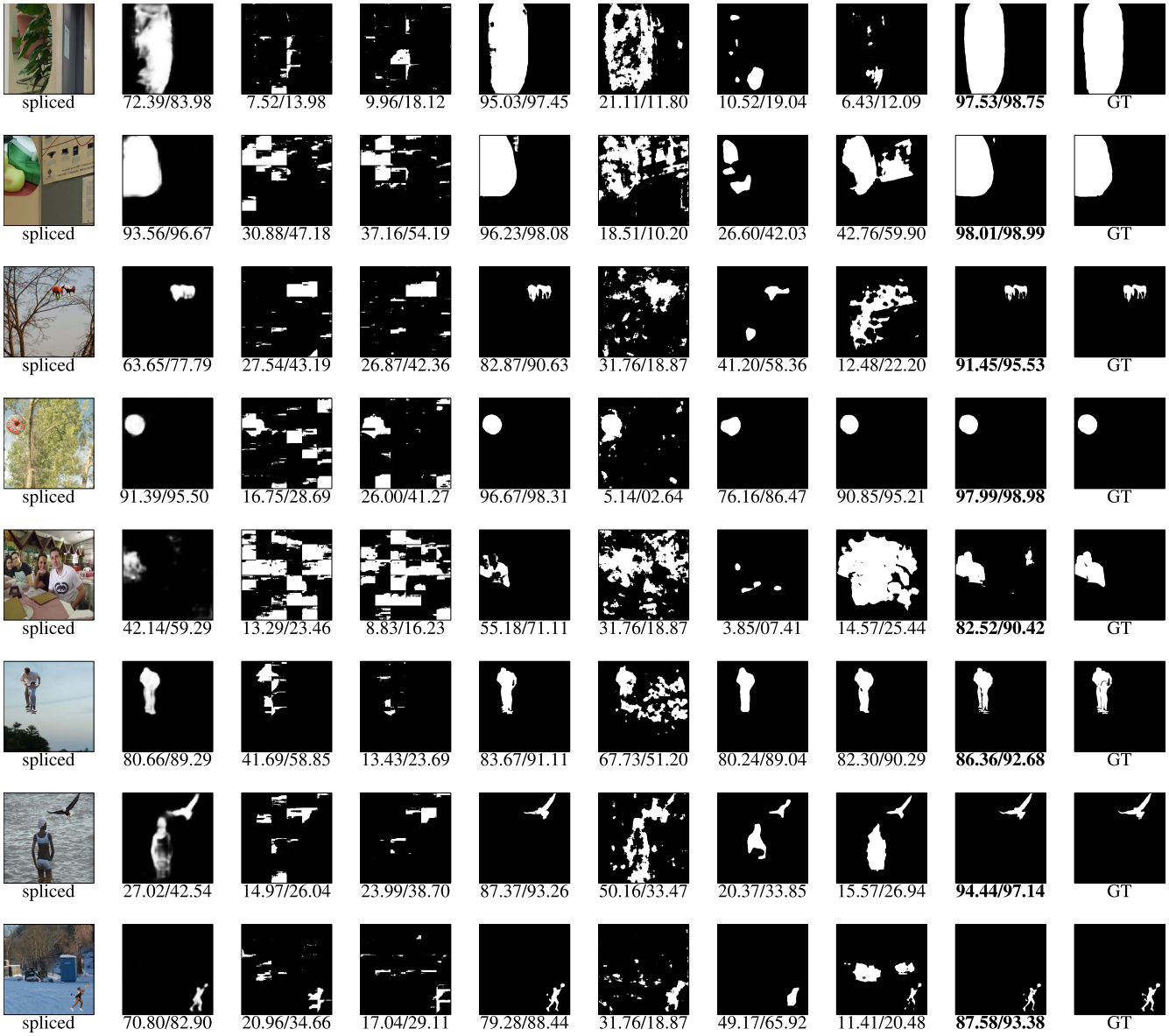

Fig. 5. Localization maps of different compared methods (IoU/F1-score (%)). The first column is the spliced images; the second to the ninth columns represent the localization maps obtained by MFCN [26], CNN_LSTM [33], SGN_CRF [27], RRU-Net [35], ManTra-Net [36], HLED [34], DenseFCN [37] and the proposed method, respectively; the last column is the corresponding ground-truth mask of the first column. The best results are highlighted in bold.

robustness of the proposed method against several common post-processing operations. The spliced images for testing are separately JPEG compressed, scaled, Gaussian filtered and sharpened with the parameters introduced in IV-A.1, and then tested with the models trained using the spliced images without post-processing. In the experiments, we adopt MFCN [26], DenseFCN [37] and RRU-Net [35]) for comparison, since these methods are the top three in terms of localization accuracy among the compared methods introduced in IV-B.2. The F1-scores obtained with different methods against several post-processing operations are shown in Fig. 6, from which the following observations can be obtained:

- *Robustness Against JPEG Compression:* Spliced images may be processed by JPEG compression to alleviate the inconsistencies between the spliced regions and the untampered regions. Fig. 6(a) illustrates that DenseFCN [37] is less sensitive to JPEG compression, but its overall localization results are not satisfactory. With QFs decreasing, the localization results of the other three tested methods degrade to some extent. While the proposed method obtains the best localization results among all the compared methods.
- *Robustness Against Image Scaling*: In practice, spliced images may undergo image scaling. Fig. 6(b) shows the F1-scores obtained with different methods with the variation of SFs. It is seen that these four methods can resist image scaling to some extent, but the proposed method outperforms the other methods in all the test cases. It can also be seen that when $SF < 1$, the localization results degrade obviously with the decreasing of SFs; when $SF > 1$, the localization results change little with the increasing of SFs. The reason for this may be due to that scaling down degrade the quality of images

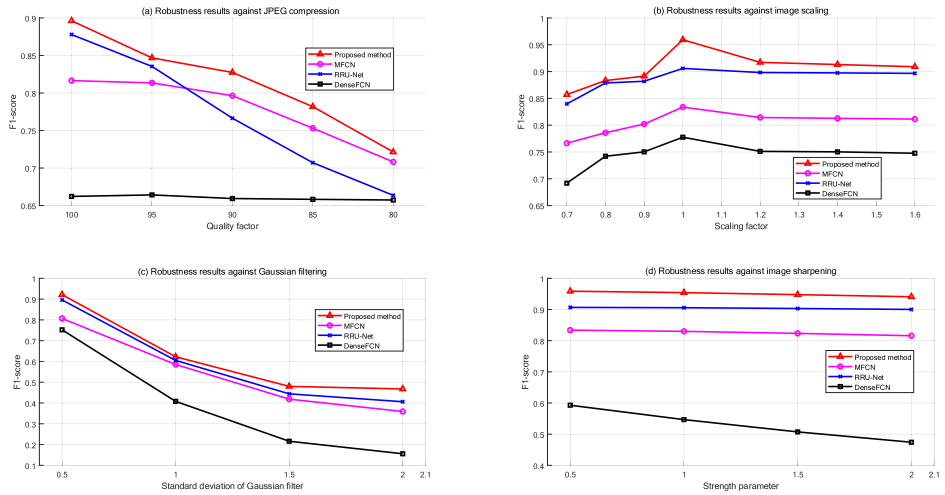

Fig. 6. F1-scores by different methods against post-processing. (a) Robustness results against JPEG compression; (b) Robustness results against image scaling; (c) Robustness results against Gaussian filtering; (d) Robustness results against image sharpening.

seriously, while scaling up can retain the splicing traces in images.

- *Robustness Against Gaussian Filtering:* Gaussian filtering is commonly used to smooth the edges of spliced images. Fig. 6(c) demonstrates that the results of the four methods are sensitive to Gaussian filtering. With the increase of the standard deviation of Gaussian filter, the localization results obtained by the proposed method degrade more slowly than those by the other three methods. Hence, it can be concluded that the proposed method is more robust against Gaussian filtering.
- *Robustness Against Image Sharpening:* Image sharpening is also commonly used for post-processing. Fig. 6(d) shows that, except for DenseFCN, the localization results of the other three methods vary little with the increase of the strength parameter. Moreover, it is obvious that the proposed method outperforms all the compared methods significantly.

According to the above experimental results, we can see that the proposed method is more robust against image scaling and image sharpening, and less robust against JPEG compression and Gaussian filtering. A possible explanation for this is that, for image post-processing operations like image scaling and image sharpening, the image contents will not change too much, so the splicing traces are well retained. On the contrary, JPEG compression and Gaussian filtering degenerate image details too much, hence the splicing traces are weakened. Whereas, the proposed method achieves the best localization results among all the compared methods. In conclusion, the proposed method is robust against these common post-processing operations.

V. Conclusion

In this paper, a novel multi-task SE-Network consisting of an edge-guided stream and a label mask stream has been proposed for splicing localization. First, a multi-task loss is designed by taking label masks, mask edges, and image edges into consideration to improve the localization rate of the spliced edges. The mask edges and the image edges can supply abundant supervision on the spliced region. Second, to provide more reliable feature of splicing, the low-level feature maps generated from shallow layers are fused with the high-level features from deep layers in the decoder. Finally, SEAMs are incorporated into the multi-task network to recalibrate the fused feature maps. The feature maps in different channels are reweighted to make the model focus on the spliced traces. The ablation studies have verified the effectiveness of each component of the network. Experimental results on two synthetic splicing datasets and four benchmark datasets have shown that the proposed multi-task SE-Network achieves the best performance among the compared splicing localization methods. Moreover, the proposed method has turned out to be robust against various post-processing operations.

Although the proposed method outperforms existing splicing localization methods, its generalization ability needs to be further improved. This limitation results in that the performance of splicing localization on small datasets, such as Carvalho dataset, is not satisfactory. To overcome the limitation, improvements should be made in the following two aspects in our future work. First, by means of few-shot learning, we train a transformer that can learn the variations among samples, and then implement data augmentation with the transformer. Second, we use embedding learning to map the samples to a low dimensional space to reduce the hypothesis space, and then obtain the approximate solution of the model under the hypothesis space with a small number of samples. In addition, to further enhance the localization accuracy, we will investigate some other effective losses that can provide more useful supervision information.

References

[1] H. Farid, "Image forgery detection," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 16–25, Mar. 2009.

[2] T. J. de Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. Rocha, "Exposing digital image forgeries by illumination color classification," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 7, pp. 1182–1194, Jul. 2013.

[3] X. Liao, K. Li, X. Zhu, and K. J. R. Liu, "Robust detection of image operator chain with two-stream convolutional neural network," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 5, pp. 955–968, Aug. 2020.

[4] Z. He, W. Lu, W. Sun, and J. Huang, "Digital image splicing detection based on Markov features in DCT and DWT domain," *Pattern Recognit.*, vol. 45, no. 12, pp. 4292–4299, 2012.

[5] X. Zhao, S. Wang, S. Li, and J. Li, "Passive image-splicing detection by a 2-D noncausal Markov model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 2, pp. 185–199, Feb. 2015.

[6] E.-S.-M. El-Alfy and M. A. Qureshi, "Combining spatial and DCT based Markov features for enhanced blind detection of image splicing," *Pattern Anal. Appl.*, vol. 18, no. 3, pp. 713–723, Aug. 2015.

[7] M. Kaur and S. Gupta, "A passive blind approach for image splicing detection based on DWT and LBP histograms," in *Proc. Int. Symp. Secur. Comput. Commun.* Singapore: Springer, 2016, pp. 318–327.

[8] J. Wang, Q. Ni, G. Liu, X. Luo, and S. K. Jha, "Image splicing detection based on convolutional neural network with weight combination strategy," *J. Inf. Secur. Appl.*, vol. 54, Oct. 2020, Art. no. 102523.

[9] N. Kanwal, A. Girdhar, L. Kaur, and J. S. Bhullar, "Digital image splicing detection technique using optimal threshold based local ternary pattern," *Multimedia Tools Appl.*, vol. 79, nos. 19–20, pp. 12829–12846, May 2020.

[10] M. Zampoglou, S. Papadopoulos, and Y. Kompatsiaris, "Large-scale evaluation of splicing localization algorithms for web images," *Multimedia Tools Appl.*, vol. 76, no. 4, pp. 4801–4834, Feb. 2017.

[11] B. Mahdian and S. Saic, "Using noise inconsistencies for blind image forensics," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1497–1503, Sep. 2009.

[12] C.-T. Li and Y. Li, "Color-decoupled photo response non-uniformity for digital image forensics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 2, pp. 260–271, Feb. 2012.

[13] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva, "A Bayesian-MRF approach for PRNU-based image forgery detection," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 4, pp. 554–567, Apr. 2014.

[14] S. Lyu, X. Pan, and X. Zhang, "Exposing region splicing forgeries with blind local noise estimation," *Int. J. Comput. Vis.*, vol. 110, no. 2, pp. 202–221, Nov. 2014.

[15] D. Cozzolino, G. Poggi, and L. Verdoliva, "Splicebuster: A new blind image splicing detector," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Nov. 2015, pp. 1–6.

[16] P. Korus and J. Huang, "Multi-scale analysis strategies in PRNU-based tampering localization," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 809–824, Apr. 2016.

[17] A. E. Dirik and N. Memon, "Image tamper detection based on demosaicing artifacts," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 1497–1500.

[18] P. Ferrara, T. Bianchi, A. D. Rosa, and A. Piva, "Image forgery localization via fine-grained analysis of CFA artifacts," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 5, pp. 1566–1577, Oct. 2012.

[19] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *Proc. 6th Int. Workshop Inf. Hiding*. Berlin, Germany: Springer-Verlag, 2004, pp. 128–147.

[20] Z. Lin, J. He, X. Tang, and C.-K. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognit.*, vol. 42, no. 11, pp. 2492–2501, 2009.

[21] W. Luo, J. Huang, and G. Qiu, "JPEG error analysis and its applications to digital image forensics," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 480–491, Sep. 2010.

[22] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1003–1017, Jun. 2012.

[23] J. Yang, J. Xie, G. Zhu, S. Kwong, and Y.-Q. Shi, "An effective method for detecting double JPEG compression with the same quantization matrix," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 11, pp. 1933–1942, Nov. 2014.

[24] J. Wang, H. Wanga, J. Li, X. Luo, Y.-Q. Shi, and S. K. Jha, "Detecting double JPEG compressed color images with the same quantization matrix in spherical coordinates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2736–2749, Aug. 2020.

[25] J. Yang, Y. Zhang, G. Zhu, and S. Kwong, "A clustering-based framework for improving the performance of JPEG quantization step estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 4, pp. 1661–1672, Apr. 2021.

[26] R. Salloum, Y. Ren, and C.-C. J. Kuo, "Image splicing localization using a multi-task fully convolutional network (MFCN)," *J. Vis. Commun. Image Represent.*, vol. 51, pp. 201–209, Feb. 2018.

[27] X. Cun and C.-M. Pun, "Image splicing localization via semi-global network and fully connected conditional random fields," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 252–266.

[28] Y. Liu, X. Zhu, X. Zhao, and Y. Cao, "Adversarial learning for constrained image splicing detection and localization based on atrous convolution," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 10, pp. 2551–2566, Oct. 2019.

[29] Y. Liu and X. Zhao, "Constrained image splicing detection and localization with attention-aware encoder-decoder and atrous convolution," *IEEE Access*, vol. 8, pp. 6729–6741, 2020.

[30] B. Liu and C.-M. Pun, "Exposing splicing forgery in realistic scenes using deep fusion network," *Inf. Sci.*, vol. 526, pp. 133–150, Jul. 2020.

[31] F. Ding, G. Zhu, M. Alazab, X. Li, and K. Yu, "Deep-learning-empowered digital forensics for edge consumer electronics in 5G HetNets," *IEEE Consum. Electron. Mag.*, early access, Dec. 28, 2020, doi: 10.1109/MCE.2020.3047606.

[32] B. Xiao, Y. Wei, X. Bi, W. Li, and J. Ma, "Image splicing forgery detection combining coarse to refined convolutional neural network and adaptive clustering," *Inf. Sci.*, vol. 511, pp. 172–191, Feb. 2020.

[33] J. H. Bappy, A. K. Roy-Chowdhury, J. Bunk, L. Nataraj, and B. S. Manjunath, "Exploiting spatial structure for localizing manipulated image regions," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4970–4979.

[34] J. H. Bappy, C. Simons, L. Nataraj, B. S. Manjunath, and A. K. Roy-Chowdhury, "Hybrid LSTM and encoder–decoder architecture for detection of image forgeries," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3286–3300, Jul. 2019.

[35] X. Bi, Y. Wei, B. Xiao, and W. Li, "RRU-Net: The ringed residual U-Net for image splicing forgery detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 30–39.

[36] Y. Wu, W. Abdalmageed, and P. Natarajan, "ManTra-Net: Manipulation tracing network for detection and localization of image forgeries with anomalous features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9543–9552.

[37] P. Zhuang, H. Li, S. Tan, B. Li, and J. Huang, "Image tampering localization using a dense fully convolutional network," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 2986–2999, 2021.

[38] K. Bahrami, A. C. Kot, L. Li, and H. Li, "Blurred image splicing localization by exposing blur type inconsistency," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 5, pp. 999–1009, May 2015.

[39] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[40] Y. Zhang, F. Ding, S. Kwong, and G. Zhu, "Feature pyramid network for diffusion-based image inpainting detection," *Inf. Sci.*, vol. 572, pp. 29–42, Sep. 2021.

[41] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014, *arXiv:1412.6806*.

[42] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[43] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[44] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[45] R. Gonzalez and R. Woods, *Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 2006.

[46] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, "Gated-SCNN: Gated shape CNNs for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5229–5238.

[47] W. Zhu *et al.*, "AnatomyNet: Deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy," *Med. Phys.*, vol. 46, no. 2, pp. 576–589, Feb. 2018.

[48] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[50] J. Dong, W. Wang, and T. Tan, "CASIA image tampering detection evaluation database," in *Proc. IEEE China Summit Int. Conf. Signal Inf. Process.*, Jul. 2013, pp. 422–426.

[51] T.-T. Ng, J. Hsu, and S.-F. Chang, "Columbia image splicing detection evaluation dataset," CalPhotos Digit. Library, DVMM Lab. Columbia Univ., New York, NY, USA, Tech. Rep. #203-2004-3, 2009.

[52] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2014, pp. 740–755.
[53] T. Gloe and R. Böhme, "The 'Dresden Image Database' for benchmarking digital image forensics," in *Proc. ACM Symp. Appl. Comput.*, 2010, pp. 1584–1590.

Yulan Zhang received the M.S. degree from Wuyi University, Jiangmen, China, in 2017. She is currently pursuing the Ph.D. degree with the Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences. Her research interests mainly include multimedia security, deep learning, and image processing.

Guopu Zhu (Senior Member, IEEE) received the B.S. degree in transportation from Jilin University, China, in 2002, and the M.S. and Ph.D. degrees in control science and engineering from the Harbin Institute of Technology, China, in 2004 and 2007, respectively. He is currently a Professor with the Harbin Institute of Technology. He has authored or coauthored more than 40 articles in peer-reviewed international journals. His main research areas are multimedia security, image processing, and control theory.

Ligang Wu (Fellow, IEEE) received the B.S. degree in automation from the Harbin University of Science and Technology, China, in 2001, and the M.E. degree in navigation guidance and control, and the Ph.D. degree in control theory and control engineering from the Harbin Institute of Technology, China, in 2003 and 2006, respectively.

He was a Research Associate/a Senior Research Associate with The University of Hong Kong, the City University of Hong Kong, and Imperial College London. He is currently a Professor with the Harbin Institute of Technology. He has published seven research monographs and more than 200 research articles in internationally refereed journals. His current research interests include analysis and design for cyber-physical systems, robotic and autonomous systems, intelligent systems, and power electronic systems. His awards and recognitions include the National Science Fund for Distinguished Young Scholar, the China Young Five-Four Medal, the Distinguished Professor of Chang Jiang Scholar, and the Highly Cited Researcher since 2015. He also serves as an Associate Editor for a number of journals, including IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE/ASME TRANSACTIONS ON MECHATRONICS, and *IET Control Theory & Applications*.

Sam Kwong (Fellow, IEEE) received the B.Sc. degree from the State University of New York at Buffalo, the M.A.Sc. degree in electrical engineering from the University of Waterloo, Canada, and the Ph.D. degree from FernUniversität in Hagen, Germany.

Before working with academia, he was a Diagnostic Engineer with Control Data Canada, where he was responsible for designing diagnostic software to detect the manufacturing faults of the VLSI chips in the Cyber 430 machine. He later joined Bell Northern Research as a Member of Scientific Staff working on the Integrated Services Digital Network (ISDN) Project. In 1996, he was responsible for designing the first handheld GSM mobile phone consultancy project at the City University of Hong Kong, one of the largest at the time. He also served as the Head and a Professor with the Department of Computer Science, City University of Hong Kong, from 2012 to 2018, where he is currently the Chair Professor. He has been actively engaged in knowledge exchange between academia and industry. He has coauthored three research books, eight book chapters, and over 300 technical articles.

Dr. Kwong was elevated to an IEEE Fellow for his contributions to optimization techniques in cybernetics and video coding in 2014. His involvement in the multiple facets of IEEE throughout the years has been extensive and committed. With respect to the IEEE Systems, Man and Cybernetics Society (SMCS), he serves as the Hong Kong SMCS Chapter Chairperson, a Board Member, a Conference Coordinator, a Membership Coordinator, and a member of the Long Range Planning and Finance Committee, the Vice President of Conferences and Meetings, and the Vice President of Cybernetics. He led the IEEE SMC Hong Kong Chapter to win the Best Chapter Award in 2011 and awarded the Outstanding Contribution Awards for his contributions to SMC in 2015. He is also the President-Elect of the IEEE SMC Society. He also frequently delivers keynote speeches in IEEE-supported conferences. He is also an Associate Editor of leading IEEE transaction journals, such as IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and IEEE TRANSACTIONS ON CYBERNETICS.

Hongli Zhang (Member, IEEE) received the B.Sc. degree in computer science from Sichuan University, Chengdu, China, in 1994, and the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999. She is currently a Professor with the School of Cyberspace Science, Harbin Institute of Technology. Her research interests include network and computer security, network modeling, and parallel processing.

Yicong Zhou (Senior Member, IEEE) received the B.S. degree in electrical engineering from Hunan University, Changsha, China, and the M.S. and Ph.D. degrees in electrical engineering from Tufts University, Medford, MA, USA.

He is currently a Professor with the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include image processing, computer vision, machine learning, and multimedia security.

Dr. Zhou is a fellow of the Society of Photo-Optical Instrumentation Engineers (SPIE) and was recognized as one of World's Top 2% Scientists and one of the Highly Cited Researchers in 2020. He received the Third Price of Macao Natural Science Award as a Sole Winner in 2020 and a co-recipient in 2014. He has been the Leading Co-Chair of Technical Committee on Cognitive Computing in the IEEE Systems, Man, and Cybernetics Society, since 2015. He serves as an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and four other journals.