This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3204753

SNIS: A Signal Noise Separation-based Network for Post-processed Image Forgery Detection

Jiaxin Chen, Xin Liao*, Wei Wang, Zhenxing Qian, Zheng Qin, and Yaonan Wang

***Abstract*—Image forgery detection has aroused widespread research interest in both academia and industry because of its potential security threats. Existing forgery detection methods achieve excellent tampered regions localization performance when forged images have not undergone post-processing, which can be detected by observing changes in the statistical features of images. However, forged images may be carefully post-processed to conceal forgery boundaries in a particular scenario. It becomes tough challenging to these methods. In this paper, we perform an analogous analysis between image forgery detection and blind signal separation, and formulate the post-processed image forgery detection problem into a signal noise separation problem. We also propose a signal noise separation-based (SNIS) network to solve the problem of detecting post-processed image forgery. Specifically, we first adopt the signal noise separation module to separate tampered region from the complex background region with post-processing noise, which weakens or even eliminates the negative impact of post-processing on forgery detection. Then, the multi-scale feature learning module uses a parallel atrous convolution architecture to learn high-level global features from multiple perspectives. Besides, a feature fusion module is utilized to enhance the discriminability of tampered regions and real regions by strengthening the boundary information. Finally, the prediction module is designed to predict the tampered region and classify the type of tampering operation. Extensive experiments show that the proposed SNIS is not only effective for forgery detection on forged images without post-processing, but also promising in robustness against multiple post-processing attacks. Furthermore, SNIS is robust in detecting forged images from unknown sources.**

***Index Terms*—Image forgery detection, tampered region localization, signal noise separation, post-processed images.**

I. Introduction

NOWADAYS, digital images are often used as electronic evidence to strengthen or refute a certain claim in a news report, new media marketing, forensic testimony, or criminal investigation so that people have a more intuitive and powerful understanding of the reported event. However, all this only holds under the condition that the content depicted in the digital image is true.

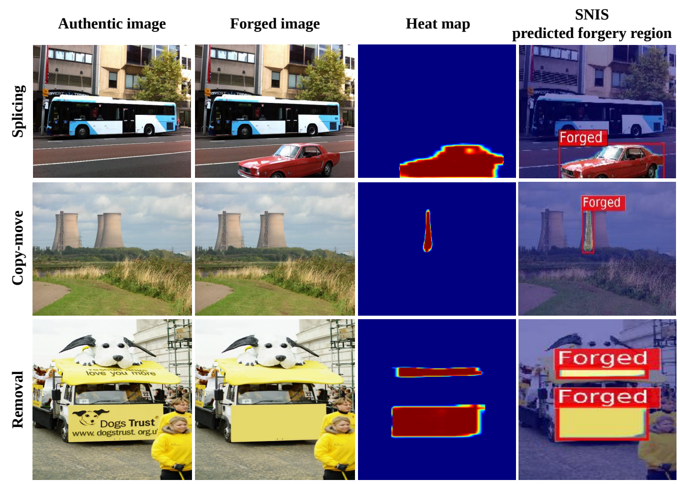


Fig. 1. Examples of images forged by splicing, copy-move, or removal, which are selected from the NIST16 dataset. The first two columns contain examples of authentic and forged images. The third column provides a heat map generated by our SNIS network, which highlights the tampered regions of the forged images. The last column provides the forgery regions predicted using SNIS.

The development of image processing techniques makes it very easy to forge images without leaving any perceptible artifacts. As shown in Fig. 1, splicing, copy-move, and removal are the most commonly used semantic focused operations. Splicing is copying one or more regions from an image and pasting them into other images, copy-move is copying one or more regions from an image and pasting them into the same image, and removal is deleting one or more regions from an authentic image. Image forgery has recently become a potential threat, which has a negative impact on many aspects of our life, such as fake news, false propaganda, bogus certificate, and even blackmail [1], [2]. Therefore, there is an urgent need to develop a well-designed method to assist in the fight against image malicious forgery.

Image forensics technology for locating tampered image regions has attracted growing attention from researchers in recent years. These methods mainly realize image forgery localization by exposing some observable artifacts, such as color inconsistency [3], [4], compression error difference[5], noise inconsistency [6], [7], camera filter array patterns [8],

This work was supported in part by the National Natural Science Foundation of China under Grant 61972142, 61972395, U20A20174, U20B2051, and 62002112, Natural Science Foundation of Hunan Province under Grant 2020JJ4212, and 2021JJ40117.

Jiaxin Chen, Xin Liao and Zheng Qin are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: chenjiaxin@hnu.edu.cn; xinliao@hnu.edu.cn; zqin@hnu.edu.cn).

Wei Wang is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: wwang@nlpr.ia.ac.cn).

Zhenxing Qian is with the School of Computer Science, Fudan University, Shanghai 200433, China (e-mail: zxqian@fudan.edu.cn).

Yaonan Wang is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with the National Engineering Laboratory for Robot Visual Perception and Control Technology, Changsha 410082, China (e-mail: yaonan@hnu.edu.cn).

*Corresponding author: Xin Liao.

semantic patches [9]-[11], and edge feature [12]. Instead of focusing on a specific tampering manipulation, researchers have studied general-purpose forensics [13], [14].

Despite the increasing emergence of fake image detection methods, identifying tampering manipulations and locating forgery regions are extremely challenging in practice. The reason is that image forgery usually involves multiple complex tampering manipulations [15]. For instance, if a forger wants to replace a region in an image using another region from another image, he or she needs to apply splicing to manipulate image regions. In order to hide visually perceptible tampering traces, the forger may tweak the image by utilizing some post-processing manipulations, e.g., Gaussian blurring, and median filtering. Finally, the tampered image will be compressed for storage. The complex image forgery process makes forgery detection difficult.

Previous method [16] can easily expose the tampered area in the forged images without post-processing, while the performance of this method degrades when detecting forged images with post-processing. The dense fully convolutional network [17] focused on book-cover image manipulation localization may also lead to over-fitting when conducting cross-dataset detection. In summary, there are two main challenges when exposing tampered image regions: 1) forgery detection architecture should be protected against multiple post-processing operations; 2) the robustness for cross-dataset detection should be improved.

In order to address the challenges, we propose a signal noise separation-based (SNIS) network to locate tampered regions with post-processed images and identify the type of semantic tampering operation. Using semantic focused operations to manipulate an image is essentially a region superimposed on another region, which can be regarded as a mixture of two independent regions. While using post-processing operations to weaken tampering artifacts is equivalent to adding noise signals to the mixed region. If the main tampered region can be separated from the background region with post-processing noise, the influence of complex background texture and post-processing operations on forgery detection can be reduced or even eliminated. As a result, the semantic tampering traces will become easier to detect. Based on this idea, we design a signal noise separation module based on blind signal separation to address the first challenge. For the second challenge, we propose a multi-scale feature learning module that exploits a parallel atrous convolution architecture to mine multi-scale information of the investigated image, thereby enhancing the global feature representation. The main contributions of this work are summarized as follows:

1) We introduce blind signal separation into the forgery detection of post-processed images, which brings a different viewpoint to forgery detection. Different from existing methods that extract features from images directly, SNIS uses the signal noise separation module to separate the tampered region and background region with noise introduced by post-processing operations. Ultimately, the influence of complex image background and post-processing operations on the forgery detection would be weakened and the detection performance would be improved.
2) We analyze the effectiveness of SNIS to locate post-processed image forgery, which theoretically demonstrates the interpretability of our architecture. Analysis of the forgery process indicates that forgery localization can be transformed into signal noise separation. The optimal solution analysis shows that tampered regions can be separated by finding the optimal separation matrix.
3) We conduct extensive experiments to verify the effectiveness of SNIS in identifying semantic tampering operations and locating tampered regions. Experimental results demonstrate that SNIS achieves promising performance compared to state-of-the-art methods when forged images have experienced multiple post-processing operations. In addition, SNIS performs well in detecting fake images from unknown sources.

The remainder of this paper is organized as follows. Section II discusses related work. Section III formulates the image forgery localization issue with blind signal separation and provides the theoretical analysis. Section IV describes the proposed network architecture. Several experimental analyses are provided in Section V, the results demonstrate the effectiveness of our proposed network. Finally, the concluding remarks are given in Section VI.

II. Related Work

In this section, we briefly review the related works about image operation forensics, whose goal is to identify the tampering operations experienced by images. We then review the works about forgery localization, which focuses on detecting tampered regions.

A. *Image Operation Detection*

Many image forensics techniques have been designed to detect a specific tampering operation, such as median filtering [18], [19], blurring [20], [21], resampling [22], JPEG compression [23], [24], and semantic focused operation [25], [26]. In [18], the statistical properties of the median filter residual were fitted to an autoregressive (AR) model and the AR coefficients were defined as the features for median filtering detection. In [20], an approach was presented to construct an optimal blurring detection classifier based on support vector machines, which used different types of image information, including image color, gradient, and spectral information. A function of the candidate step taking a similar shape to the DCT coefficients distribution of compressed images was introduced in [24] to achieve quantization step estimation for JPEG images. In [26], noise discrepancy was analyzed to expose splicing forgery artifacts. Besides, adaptive singular value decomposition was proposed to improve detection accuracy.

Since the tampering operation used to forge an image is usually unknown, researches on general-purpose operation detection are also of great importance. Li et al. [27] analyzed the properties of local pixels in the residual domain and proposed a universal feature set to identify many common image operations. Bayar et al. [28] developed a constrained convolutional layer to suppress the content of an image and adaptively learn operation detection features. Chen et al. [29] automated the neural network architecture design for multi-purpose image forensics, which generated high-performing CNNs for specific forensic tasks through reinforcement learning. Singh et al. [30] exploited local dense connections and global residual learning for better general-purpose forensics performance by using robust residual dense blocks. Zhan et al. [31] trained a forensic model utilizing prior knowledge transferred from the steganalysis model and presented a parameter transfer strategy for general-purpose operation detection on different databases.

However, a fake image is usually forged by multiple tampering manipulations simultaneously in real-world scenarios. Recently, there have been a lot of efforts to detect multiple operations in digital images [32]-[41]. Yang et al. [32] designed a statistical likelihood function to characterize the mixed compression and enhancement artifacts, and to further estimate parameters of JPEG-domain enhanced images. Liu et al. [33] proposed to use the histogram of the difference image extremum interval to estimate the downscaling factor of the pre-JPEG compressed images. Wang et al. [34] utilized the conversion error, rounding error, and truncation error on the pixel in the spherical coordinate system to detect the recompression in the color images. Wang et al. [35] detected double JPEG compressed images by using quaternion mapping to retrain the relationship between continuously compressed JPEG images.

Considering that the order of operations affects the generated fake image, Chu et al. [37] formulated the order of operations detection problem as a multiple hypotheses testing problem. Then, an information theoretical framework based on multiple hypotheses was proposed to determine whether the order of operations is distinguishable. In [15], order forensics convolutional neural network for detecting image operator chain has been presented, which utilized tampering artifact evidence and local noise residual evidence. In [39], different forensic knowledge integrated by a decision fusion method has been used to identify multiple tampering operations in image operation chains. Our recent work [40] proposed a features decoupling method based on blind signal separation for multiple manipulations identification.

B. Image Forgery Localization

Since image forgery usually uses different semantic focused tampering operations to change image content, image forensics not only needs to identify the types of tampering operations but also needs to further locate the tampered regions. Nowadays, there has been a growing interest to locate image forgery. In [11], a copy-move localization scheme has been proposed, which segmented images into semantically independent patches and then matched keypoints among these patches. In [12], a multi-task fully convolutional network that can learn both surface label and the edge of the spliced region was designed to localize image splicing attacks. In [42], a CAT-Net with RGB and DCT streams was proposed for image splicing localization, which can jointly learn compression artifact features on RGB and DCT domains.

In order to capture evidence of more general semantic tampering operations such as removal and copy-move, Yang et al. [13] proposed a coarse-to-fine architecture to learn unified global manipulation features and finer local features, so that tampered regions can be segmented. In [14], a manipulation localization architecture enabled tampered regions to be segmented out from non-tampered ones by utilizing resampling features, long short-term memory cells, and an encoder-decoder network. Cozzolino et al. [43] designed a Siamese network to extract a camera model fingerprint, so that model-related artifacts are enhanced, and then forgery localization can be achieved.

Considering that the images may experience a post-processing operation, Zhou et al. [16] designed a two-stream Faster R-CNN network that explores both content-related features and noise features to achieve forgery localization. A fully convolutional encoder-decoder architecture based on Photoshop tampering traces detection is proposed in [17] to localize tampered regions. Wu et al. [44] formulated the forgery localization problem as a local anomaly detection problem and introduced a unified deep neural architecture to perform localization without extra preprocessing and postprocessing. Rao et al. [45] proposed a self-supervised domain adaptation network consists of a Siamese architecture and a compression approximation network for JPEG-resistant image forgery localization. In [46], a robust training method was presented to fight against the OSN-shared forgeries, which modeled the noise introduced by OSNs and incorporated noise into the training framework. In real scenarios, the image forgery process is complicated, and the databases from which the images come are different. Therefore, the performance of most methods may degrade when solving image forgery detection in this scenario.

III. Analysis of Image Forgery Localization and Blind Signal Separation

In this section, we first introduce the motivation for applying blind signal separation to image forgery localization. Then, we perform an analogous analysis of image forgery localization and blind signal separation.

A. Motivation

Image forgery with semantic focused tampering manipulations can be regarded as the mixing and superposition of different image regions. Meanwhile, if some post-processing operations are applied to disguise forged images, the traces of post-processing will be confused with those of semantic tampering operations. Thus, we can only extract mixed region information with post-processing noise from the given images, making image forgery localization more difficult.

As demonstrated in Fig. 2, the image forgery detection problem is similar to the blind signal separation problem. Specifically, supposing there are n independent source signals in the blind signal separation problem, and m mixing sensors are used to receive these source signals. Mixed signals can be obtained from these sensors. For instance, there are n people speaking at the same time in a room with m microphones,

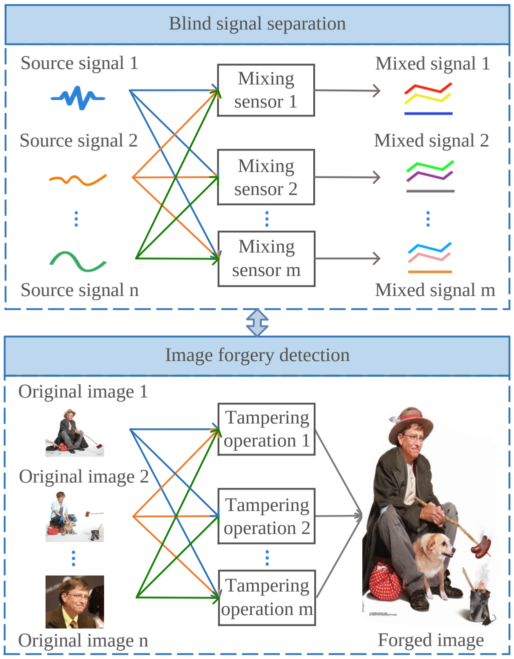


Fig. 2. Illustration of the relationship between image forgery detection and blind signal separation.

and the unique voice source signals sent by each person are simultaneously received by the microphones. Thus, what we derive from the microphones are the mixed voice signals of n people. For image forgery detection problem, a forger may adopt m tampering operations to synthesize n original images into one forged image. The forged regions and the background image are from different images, and there is no correlation between adjacent pixels. Therefore, they can be regarded as independent source signals, and the process of forgery is the process of mixing independent signals, which is consistent with blind signal separation. Because the latter tampering operations will affect the forgery traces of the previous operations in the image, we can only extract mixed forgery information from the forged image.

Correlating blind signal separation and image forgery detection, it can be found that the source signals are unknown in the blind signal separation problem, similarly, the original images containing the background image and the selected forged regions are unknown in the image forgery detection issue. Besides, for blind signal separation, the mixing method of the different signal mixing sensors is unknown and only the mixed signals can be observed. For image forgery detection, the tampering operations containing the semantic tampering operations and post-processing manipulations used are unknown, while the forged image to be detected is known and observation mixed information can be extracted from it.

The goal of blind signal separation is to restore the source signals from the observed mixed signals [47], [48]. Similarly, post-processed image forgery detection aims to separate tampered regions from the background image with post-processing noise. From the perspective of blind signal separation, the image forgery detection problem can be transformed into a signal noise separation problem. The tampered regions are regarded as the main signal, and the background area with post-processing noise is regarded as the noise signal. Through signal noise separation, the masking effect of post-processing operations on image forgery can be eliminated, thereby simplifying image forgery detection and localization.

B. The Analogous Analysis

Since different operations have different effects on image forgery, the confusion between the noise traces caused by these post-processing operations and the traces of semantic manipulations may not be simply overlapped, but interact with each other. Therefore, we utilize matrix multiplication to model the image forgery process. The image forgery without post-processing can be formulated as follows,

$$\begin{aligned} I(i,j) &= \boldsymbol{A}_0 \boldsymbol{I}_{fg}(i,j) + \boldsymbol{A}_1 \boldsymbol{I}_{bg}(i,j) \\ &= \boldsymbol{A}\boldsymbol{s}(i,j), \end{aligned} \tag{1}$$

where (i,j) represents the coordinates of each pixel in the image. $I(i,j)$ is forged image. $I_{bg}(i,j)$ and $I_{fg}(i,j)$ are real background image and foreground forged region. $\boldsymbol{A} = [\boldsymbol{A}_0, \boldsymbol{A}_1]$ is mixing matrix. $\boldsymbol{s}(i,j) = [I_{fg}(i,j), I_{bg}(i,j)]$ represents original source image.

Then, image forgery with post-processing can be formulated as follows,

$$\begin{aligned} I(i,j) &= \boldsymbol{B}(\boldsymbol{A}_0 \boldsymbol{I}_{fg}(i,j) + \boldsymbol{A}_1 \boldsymbol{I}_{bg}(i,j)) \\ &= \boldsymbol{B}\boldsymbol{A}\boldsymbol{s}(i,j), \end{aligned} \tag{2}$$

where $\boldsymbol{B}$ represents the mixing matrix produced by post-processing operations.

The tampered region $I_{fg}(i,j)$ can be viewed as the main source signal. The real background region with traces left by post-processing operations $I_{bg}(i,j)$ can be regarded as a noise source signal. The forged image $I(i,j)$ is the observed mixed signal. Thus, image forgery localization can be transformed into a signal noise separation issue.

In fact, image forgery localization based on signal noise separation is an optimal solution problem. That is, the observed mixed signal is decomposed into several independent components by an optimization algorithm according to the principle of statistical independence. These independent components are an approximate estimation of the main source signal and the noise source signal. Based on the principle that the eigen decomposition of the covariance matrix can make the data irrelevant, we optimize the separation matrix $\boldsymbol{W}$ to make the data in the mixed signal independent of each other and achieve the purpose of separating the main source signal and the noise signal. Formally, we define the signal noise separation as

$$\boldsymbol{W}\boldsymbol{I}(i,j) = \begin{cases} \boldsymbol{W}\boldsymbol{A}\boldsymbol{s}(i,j), \text{without post-processing} \\ \\ \boldsymbol{W}\boldsymbol{B}\boldsymbol{A}\boldsymbol{s}(i,j), \text{with post-processing} \end{cases} \tag{3}$$

If $\boldsymbol{W} = \boldsymbol{A}^+$ or $\boldsymbol{W} = (\boldsymbol{B}\boldsymbol{A})^+$ represents the pseudo inverse matrix of $\boldsymbol{A}$ or $\boldsymbol{B}\boldsymbol{A}$, a identity matrix can be obtained by calculating $\boldsymbol{W}\boldsymbol{A}$ or $\boldsymbol{W}\boldsymbol{B}\boldsymbol{A}$, and then the tampered region can be distinguished from the background image regardless of whether there exists post-processing noise.

$$\boldsymbol{W}\boldsymbol{I}(i,j) = \boldsymbol{s}(i,j) = [I_{fg}(i,j), I_{bg}(i,j)]. \tag{4}$$

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3204753

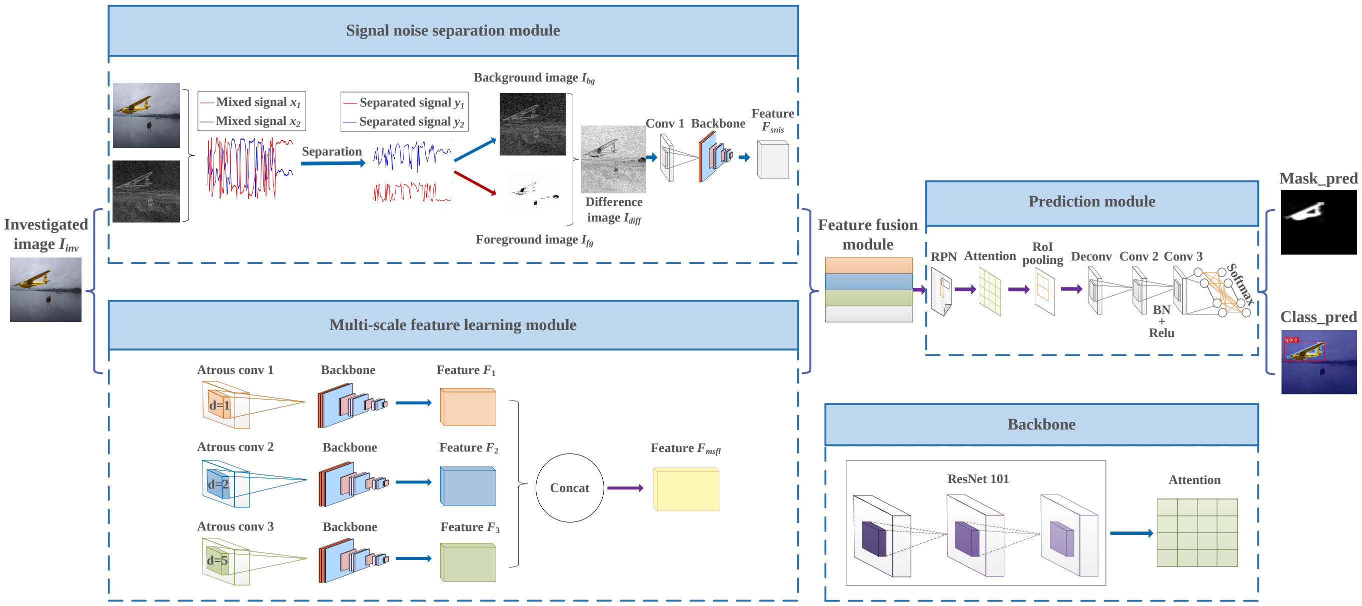


Fig. 3. Overview of the proposed SNIS network for post-processed image forgery detection. The signal noise separation module first distinguishes the foreground tampered image I_{fg} from the background image I_{bg} with post-processing noise by calculating the optimal separation matrix. Then, the difference image I_{diff} is fed to a convolution followed by a backbone layer to learn the local inconsistency feature. The multi-scale feature learning module utilizes a parallel atrous convolution architecture to learn high-level tampering information and obtain the global feature representation. The feature fusion module integrates F_{snis} and F_{msfl} based on cross fusion, which enhances boundary features. The prediction module exploits some computing operations like RPN and attention mechanism to predict the tampered region and the type of semantic tampering operation. BN: Batch-Normalization layer. Relu: Rectified Linear Unit layer. By optimizing the model, SNIS can effectively detect post-processed image forgery.

Through the analogous analysis of image forgery detection and blind signal separation, we find that the forgery localization can be assisted from the perspective of signal noise separation. Based on the optimal solution analysis, post-processing traces left in the tampered regions can be eliminated, making semantic tampering traces easier to detect. Therefore, our signal noise separation module is beneficial to improve the localization performance of post-processed image forgery.

IV. The Proposed Network of Post-processed Image Forgery Detection

In this section, we first demonstrate an overview of the proposed SNIS. Then, the details of the signal noise separation module, multi-scale feature learning module, feature fusion module, and prediction module are introduced, respectively. Finally, we describe the training loss and network implementation details for forgery detection.

A. Overview of the Proposed Network

The main purpose of this work is to locate image tampered regions and identify semantic tampering manipulation used in image forgery. Fig. 3 illustrates the proposed signal noise separation-based network for post-processed image forgery detection. SNIS consists of four components: signal noise separation module, multi-scale feature learning module, feature fusion module, and prediction module.

Consecutive use of semantic tampering manipulation and post-processing operations confuses the traces of image forgery, making it more difficult to locate tampered regions from forged images with post-processing. For weakening the confusion effect of post-processing on forgery traces and achieving robustness against multiple post-processing attacks, we calculate the optimal separation matrix in the signal noise separation module to separate the foreground tampered region I_{fg} and the background region I_{bg} with post-processing noise. Meanwhile, for learning the local inconsistency feature F_{snis}, we calculate the difference image I_{diff} and feed it into a convolution followed by a backbone layer.

For learning high-level tampering information and increasing the detection robustness for different semantic tampering operations and images from different databases, we design a multi-scale feature learning module that uses a parallel atrous convolution architecture to extract more discriminative global feature F_{msfl} directly from forged images, so as to avoid losing too much forensic information. Furthermore, we fuse the features F_{snis} and F_{msfl} based on cross fusion to enhance boundary information in the feature fusion module. Finally, a prediction module using some convolutional computational operations is developed to locate the tampered areas and identify different semantic manipulations. After optimization, the optimal set of parameters for the network can be obtained, which will be utilized to detect post-processed image forgery.

B. Signal Noise Separation Module

The signal noise separation module is designed based on blind signal separation. With the signal noise separation module, the negative impact of complex background and post-processing manipulations on the tampered region localization can be eliminated. The detailed procedures are demonstrated as follows.

1) Build the Laplacian pyramid of the investigated image I_{inv} to obtain image I_t, where the number of pyramid levels is 3. The purpose of this step is to reduce image background interference.
2) Convert images I_{inv} and I_t into one-dimensional signals $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$.
$$\boldsymbol{x}_i = f(I_i) = [I_i(1,:), I_i(2,:), \cdots, I_i(m,:)], \tag{5}$$
where $I_i(k,:)$ represents the pixel value in the k^{th} row of the image I_i. m is the height of the image I_i.
3) Set $\boldsymbol{X} = [\boldsymbol{x}_1; \boldsymbol{x}_2]$ as the observation matrix. For reducing redundant information of the observation matrix, we decentralize the matrix $\boldsymbol{X}$ to obtain $\widetilde{\boldsymbol{X}} = [\widetilde{\boldsymbol{x}}_1; \widetilde{\boldsymbol{x}}_2]$.
$$\widetilde{\boldsymbol{x}}_i = f(I_i) - E\{f(I_i)\}, \tag{6}$$
where $E\{\cdot\}$ computes the mean value of the sampled values of the signal $\boldsymbol{x}_i$.
4) Estimate the whitening matrix $\boldsymbol{V}$ [49] and compute the whitened observation matrix $\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2\}$.
$$\boldsymbol{V} = \boldsymbol{\Lambda}^{-\frac{1}{2}} \boldsymbol{U}^T, \tag{7}$$
$$\boldsymbol{Z} = \boldsymbol{V}\widetilde{\boldsymbol{X}} = \boldsymbol{\Lambda}^{-\frac{1}{2}} \boldsymbol{U}^T \tilde{\boldsymbol{X}}, \tag{8}$$
where $\boldsymbol{\Lambda}$ represents the diagonal matrix formed by the eigenvalues of the covariance matrix of $\tilde{\boldsymbol{X}}$, and $\boldsymbol{U}$ represents the matrix formed by taking the eigenvectors corresponding to each eigenvalue in the covariance matrix as columns.
5) Iteratively obtain the optimal separation matrix $\boldsymbol{W}$ based on blind signal separation [48].
$$\begin{aligned} \boldsymbol{W}(t+1) = {} & E\{g(\boldsymbol{W}(t)\boldsymbol{Z})\boldsymbol{Z}^T\} \\ & - E\{g'(\boldsymbol{W}(t)\boldsymbol{Z})\}\boldsymbol{W}(t), \end{aligned} \tag{9}$$
where $g(\cdot)$ is a cumulative distribution function and $g(\cdot)'$ is the derivative of $g(\cdot)$. $t = 100$ denotes iteration times. $E\{\cdot\}$ calculates the mean value of the sampled values of the whitened observation matrix $\boldsymbol{Z}$.
6) The separated signals $\boldsymbol{y}_i$ is calculated as follows,
$$\boldsymbol{y}_i = \boldsymbol{W}_i \boldsymbol{Z}, \tag{10}$$
where $i = 1, 2$. $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ correspond to the signals of the background image I_{bg} and the foreground tampered image I_{fg}, respectively. Because the foreground tampered image I_{fg} is separated from the background image I_{bg} with post-processing noise, the visual masking effect of the complex background image and post-processing operations on image forgery could be weakened or even eliminated.
7) In order to obtain the difference between the background image and the foreground tampered image, the difference image I_{diff} is calculated as follows,
$$I_{diff}(i,j) = |I_{fg}(i,j) - I_{bg}(i,j)|. \tag{11}$$
8) The difference image I_{diff} is fed into the convolution layer with kernel size 3×3, followed by a backbone layer. The backbone consists of ResNet 101 network [50] and convolutional block attention module (CBAM) [51]. With this architecture, the locally inconsistent features F_{snis} of the difference image can be learned.

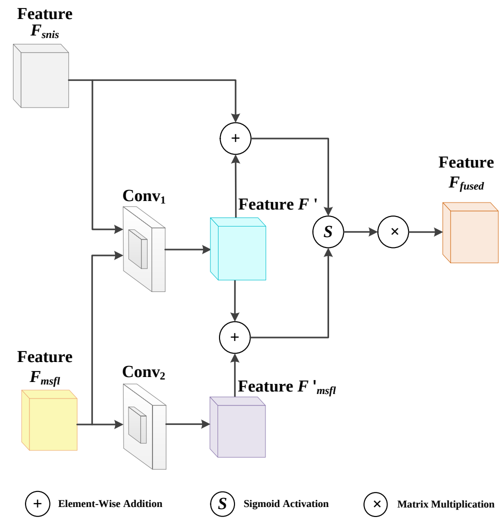


Fig. 4. Illustration of our feature fusion module. This module integrates F_{snis} and F_{msfl} by adopting convolution $Conv_1$. Subsequently, F_{snis} and F', F'_{msfl} and F' are cross fused by element-wise addition, sigmoid activation and matrix multiplication.

C. Multi-Scale Feature Learning Module

The multi-scale feature learning module uses a parallel atrous convolution architecture to mine high-level tampering information at multiple scales. The image forgery process in a real scenario is often complicated, if forensics information is only extracted from narrow local areas, the feature representations may be unstable. Therefore, it is necessary to capture more tampering traces by exploiting a wider range of information. Unlike traditionally using larger convolution kernels or pooling layers, to obtain receptive fields of different scales without changing the output feature map size, we design three parallel atrous convolutional layers.

The output of atrous convolution $y(i_d, j_d)$ of input $x(i_d, j_d)$ at each layer can be computed as
$$y(i_d, j_d) = \sum_{k_1, k_2} \phi(k_1, k_2) \cdot x(i_d + dk_1, j_d + dk_2), \tag{12}$$
where $\phi(k_1, k_2)$ is a $K \times K$ convolution filter. $k_1, k_2 \in [-fl(\frac{K}{2}), fl(\frac{K}{2})]$, and $fl(\cdot)$ is the floor function. d denotes the dilation rate.

In our SNIS, let $K = 3$ and the dilation rate of each layer is $d = \{1, 2, 5\}$. Note that the dilation rates of different atrous convolutions can be used to explore image forgery information from different perspectives.

Then each layer adopts the backbone to learn multi-scale semantic tampering traces. The high-level features from the input color image can be extracted by using ResNet 101. Meanwhile, the CBAM attention module is added to dynamically generate the weights of different connections. Thus, the global feature representation capabilities of the multi-scale feature learning module can be improved and more discriminative features F_{msfl} can be created.

D. Feature Fusion Module

In the feature fusion module, as illustrated in Fig. 4, we introduce a cross-fusion strategy to integrate the features derived from the signal noise separation module and multi-scale feature learning module. Specifically, since the signal noise separation module mainly focuses on local feature extraction, its output feature F_{snis} contains more location and detailed information, but may have problems with missed tampered regions. While the multi-scale feature learning module mainly extracts high-level global feature F_{msfl}, which contains richer semantic information, but may have problems with false positives. The fusion of local features and global features can complement each other, so that the fused features have both location details and tampered regions, strengthening the boundary feature. In order to make full use of local feature and global feature, F_{snis} and F_{msfl} are first integrated into the convolution layer $Conv_1$ with kernel size 3×3. We define the output feature as F'.

Besides, F_{msfl} is input to the convolution layer $Conv_2$ with kernel size 1×1 to reduce the feature channels, and the output feature is defined as F'_{msfl}. For improving the nonlinear expression ability of the model and learning interactive information between F_{snis} and F'_{msfl}, we perform element-wise addition across the feature F_{snis} and F', F'_{msfl} and F'. This form of information superposition can make the original local features F_{snis} contain the semantic information of the global features F'_{msfl}, and the original global features F'_{msfl} contain the details information of the local features F_{snis}. Finally, we adopt sigmoid as the activation function and perform matrix multiplication to obtain the final fused features F_{fused}, which improves tampering boundary information and optimizes the tampered region localization performance.

E. Prediction Module

The prediction module utilizes the region proposal network (RPN) [52] and CBAM attention module to propose the regions of interest (RoI) for bounding box regression, which is defined as F_{roi}. The RoI pooling layer will crop and resize the feature to $b \times 7 \times 7 \times 1024$, where b is the batch size. To upsample and reduce the channels of the feature, a deconvolutional layer is utilized and the output size is $b \times 14 \times 14 \times 256$. Then, a convolution layer Conv2 with kernel size 1×1 further reduces the feature channels to 64, followed by batch normalization and Relu activation function. To reduce model complexity, a convolution layer Conv3 with kernel size 1×1 is exploited. Finally, adopting a softmax layer to predict the mask of the forged region and the type of semantic tampering operation.

F. Training Loss

As described above, we construct a novel signal noise separation-based network for post-processed image forgery detection. In the SNIS network, we use three kinds of losses together to optimize the training model, including RPN loss, mask prediction loss, and operation classification loss. The formula of the training loss L_{SNIS} can be calculated as:

$$L_{\mathsf{SNIS}} = L_{RPN} + L_{mask\_pred} + L_{class\_pred}, \tag{13}$$

Algorithm 1 The training algorithm. The trained model is optimized by `SGD`.

Input: Training dataset $\mathcal{D}$; Ground truth $\mathcal{G}$; training iteration max_iter; learning rate α; batch size b
Output: Trained model θ

```
Initialize α = 0.001 decayed by the factor 0.1 after 40K steps and 90K steps
Initialize b = 256
while θ has not converged do
    for i = 1 → max_iter do
        F_snis = f^snis(D)
        F_msfl = f^msfl(D)
        F_fused = f^fuse(F_snis, F_msfl)
        F_roi = f^RPN(F_fused)
        g_θ ← ∇_θ((1/b) Σ_{i=1}^{i=b} L_SNIS(F_roi, G))
        θ ← θ + α·SGD(θ, g_θ)
    end for
end while
```

where

$$\begin{aligned} L_{RPN} = & \frac{1}{N_{cls}} \sum_i L_{cls}(r_i, r_i^*) \\ & + \lambda \frac{1}{N_{reg}} \sum_i r_i^* L_{reg}(d_i, d_i^*), \end{aligned} \tag{14}$$

note that r_i is the probability of anchor i being a potential tampered area, and r_i^* is the ground-truth label with a positive i. d_i and d_i^* are the 4 dimensional bounding box coordinates for anchor i and the ground-truth. N_{cls} is the size of a mini-batch in the RPN. N_{reg} is the number of anchor locations. λ is used to balance the L_{reg} and L_{cls}. L_{cls} denotes cross entropy loss for RPN, which is defined as follows,

$$L_{cls}(r_i, r_i^*) = -\sum_i r_i log(r_i^*). \tag{15}$$

L_{reg} uses smooth L_1 loss for bounding box regression, which is calculated as follows,

$$L_{reg}(d_i, d_i^*) = \begin{cases} 0.5(d_i - d_i^*)^2, \text{if } |d_i - d_i^*| < 1 \\ |d_i - d_i^*| - 0.5, \text{otherwise} \end{cases} \tag{16}$$

Similarly, L_{mask_pred} utilizes smooth L_1 loss for final bounding box regression. L_1 loss is used to calculate the error between the 4 coordinate values of the predicted tampered region and the ground-truth. When the loss converges, we can get the final bounding box coordinates, which correspond to the tampered region coordinates in the pixel-level predicted mask output by the model. L_{class_pred} uses cross entropy loss for semantic tampering operation classification. With this computable loss function, we train the model as described in Algorithm 1. $f^{snis}(\cdot)$, $f^{msfl}(\cdot)$, $f^{fuse}(\cdot)$, and $f^{RPN}(\cdot)$ represent a series of operations corresponding to the network structure. The `SGD` is employed to optimize the trained model.

The signal noise separation module can learn the local features of the tampered regions, and the multi-scale feature learning module can learn the global semantic features of the image. Through the feature fusion module, the boundary

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3204753

information of the tampered region is enhanced. We then perform related calculations in the prediction module. The predicted mask and probability output by the trained model are the localization result of the tampered region and the classification result of the semantic tampering operation.

G. Implement Details

The proposed network is trained end-to-end. The input image is resized so that the shorter side of the image is equal to 600 pixels. The batch size of the RPN proposal is 256 for training and 300 for testing. We pre-train our model for 110K steps. On the image forgery detection benchmark, the entire model is trained for 60K steps with pre-trained weights.

The learning rate is initially set to 0.001, and then is reduced to 0.0001 and 0.00001 after 40K steps and 90K steps, respectively. The SGD optimizer with default hyperparameters is adopted to optimize the forgery detection model. ALL experiments are conducted on a single NVIDIA 2080 Ti GPU.

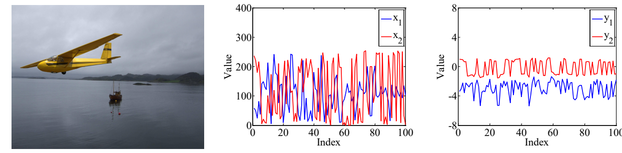
V. Experiments

In this section, following the experiment settings, the simulation of signal noise separation is provided. Secondly, ablation experiments are conducted to demonstrate the effectiveness of the signal noise separation module. Then, we provide a comparison between SNIS and some state-of-the-art methods for the forgery detection performance of forged images that have experienced post-processing operations. Moreover, we present the localization performance of forged images without post-processing using SNIS and other methods. In addition, a cross-dataset performance comparison is shown to illustrate the robustness of our SNIS. Finally, we perform some qualitative comparisons of SNIS and state-of-the-art methods.

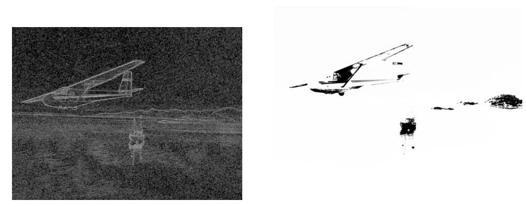
A. Experiment Settings

Pre-trained Model: Since the current standard datasets do not have enough data for deep neural network training, tampering traces may be hard to detect, resulting in unsatisfactory forgery localization performance. Zhou et al. [16] utilized the COCO synthetic dataset [53] to generate the manipulated image dataset. We use the same COCO synthetic dataset as [16] to pre-train our model, which contains 42K forged and authentic image pairs. We believe that the feature representations learned by a pre-trained model for forgery detection with a large-sample synthetic dataset can be effectively transferred to improve the feature learning for detecting forgery with small-sample standard datasets.

We split the training and testing set with the ratio of 9:1, and the same background and forged object will not appear in both the training and testing set. The ResNet 101 network used in SNIS is initialized by ImageNet weights. The output of our pre-trained model is bounding boxes with confidence scores, which represents the probability that the box contains the tampered regions. Average Precision (AP) is used for the COCO detection evaluation, and the detection accuracy can reach 77.5% by adopting the proposed SNIS.



(a) The forged image with post-processing (b) The mixed signals (c) The separated signals



(d) The background image (e) The foreground forged region

Fig. 5. Visualization results of signal noise separation from a post-processed forged image. The forged image is selected from the NIST16 dataset, numbered "NC2016_0942".

Datasets: We compare SNIS with some state-of-the-art methods on NIST Nimble 2016 (NIST16) [54], Columbia [55] and CASIA [56] dataset.

- The NIST16 dataset contains 564 tampered images in JPEG format. Each image is subjected to one of the three semantic focused operations, i.e., splicing, copy-move, and removal. The resolutions of the tampered images range from 500×500 to 5616×3744.
- The Columbia dataset contains 180 uncompressed spliced images, and the resolutions range from 757×568 to 1152×768.
- The CASIA dataset contains 921 spliced and copy-moved images in JPEG format. Besides, the tampered regions are pre-processed before generating spliced or copy-moved images. The resolutions are 384×256.

404 tampered images randomly selected from the NIST dataset are used to train the models in our experiments. Hence, the number of testing images in NIST, Columbia, and CASIA is 160, 180, and 921, respectively.

Evaluation Metric: We use pixel-level F_1 score and Area Under the receiver operating characteristic Curve (AUC) to measure the image forgery localization performance. Besides, we utilize Average Precision (AP) to evaluate the semantic tampering operation classification performance.

SOTA Models: We compare SNIS with the state-of-the-art methods. The **ELA** [5], **NOI1** [6], and **CFA1** [8] are representative of the aforementioned localization methods for a single manipulation. The **RGB-N** [16], **Cons-N** [13], **DFC-N** [17], **Noiseprint** [43], **ManTra-Net** [44], and **OSN** [46] are representative of the aforementioned localization methods for different tampering manipulations.

B. Simulation of Signal Noise Separation

In order to intuitively demonstrate the help of our signal noise separation idea for post-processed image forgery detection, we perform a visual simulation of signal noise separation.

As illustrated in Fig. 5, the forged image is post-processed with Gaussian blur ($w = 3$) followed by JPEG compression

TABLE I
COMPARISON OF ABLATION STUDY: FORGERY LOCALIZATION EVALUATION WITH AND WITHOUT (W/OUT) THE SIGNAL NOISE SEPARATION MODULE (SNISM). THE QUALITY OF JPEG COMPRESSION IS 85. "IMAGES" MEANS THAT THE TESTING FORGED IMAGES HAVE NOT EXPERIENCED POST-PROCESSING OPERATIONS.

Dataset	Method	Images	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
			3	5	7	9	11	3	5	7	9	11
F_1 comparisons on NIST16	w/out snism	0.832	0.824	0.833	0.789	0.790	0.776	0.821	0.819	0.815	0.811	0.797
	with snism	**0.891**	**0.887**	**0.878**	**0.859**	**0.848**	**0.822**	**0.882**	**0.878**	**0.863**	**0.855**	**0.854**
AUC comparisons on NIST16	w/out snism	0.954	0.951	0.958	0.934	0.938	0.929	0.945	0.944	0.941	0.946	0.940
	with snism	**0.981**	**0.981**	**0.980**	**0.975**	**0.973**	**0.965**	**0.977**	**0.979**	**0.976**	**0.974**	**0.973**
F_1 comparisons on Columbia	w/out snism	0.586	0.544	0.552	0.553	0.581	0.602	0.552	0.551	0.559	0.558	0.586
	with snism	**0.681**	**0.651**	**0.654**	**0.657**	**0.666**	**0.659**	**0.632**	**0.639**	**0.653**	**0.644**	**0.648**
AUC comparisons on Columbia	w/out snism	0.669	0.654	0.638	0.645	0.668	0.693	0.649	0.645	0.633	0.648	0.696
	with snism	**0.800**	**0.773**	**0.771**	**0.766**	**0.772**	**0.766**	**0.736**	**0.746**	**0.762**	**0.740**	**0.756**

TABLE II
COMPARISON OF ABLATION STUDY: FORGERY LOCALIZATION EVALUATION WITH AND WITHOUT (W/OUT) THE SIGNAL NOISE SEPARATION MODULE (SNISM). THE QUALITY OF JPEG COMPRESSION IS 70. "IMAGES" MEANS THAT THE TESTING FORGED IMAGES HAVE NOT EXPERIENCED POST-PROCESSING OPERATIONS.

Dataset	Method	Images	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
			3	5	7	9	11	3	5	7	9	11
F_1 comparisons on NIST16	w/out snism	0.832	0.825	0.823	0.818	0.787	0.775	0.822	0.822	0.815	0.809	0.790
	with snism	**0.891**	**0.876**	**0.873**	**0.851**	**0.822**	**0.822**	**0.878**	**0.873**	**0.862**	**0.850**	**0.851**
AUC comparisons on NIST16	w/out snism	0.954	0.950	0.951	0.950	0.934	0.924	0.948	0.950	0.949	0.948	0.944
	with snism	**0.981**	**0.980**	**0.978**	**0.974**	**0.960**	**0.966**	**0.980**	**0.978**	**0.975**	**0.973**	**0.971**
F_1 comparisons on Columbia	w/out snism	0.586	0.551	0.547	0.589	0.571	0.600	0.557	0.588	0.552	0.556	0.589
	with snism	**0.681**	**0.643**	**0.650**	**0.658**	**0.645**	**0.652**	**0.625**	**0.627**	**0.640**	**0.650**	**0.630**
AUC comparisons on Columbia	w/out snism	0.669	0.622	0.619	0.686	0.666	0.692	0.642	0.681	0.638	0.655	0.695
	with snism	**0.800**	**0.751**	**0.761**	**0.767**	**0.742**	**0.740**	**0.724**	**0.720**	**0.737**	**0.762**	**0.736**

($QF = 70$). To facilitate visualization, the image is resized to 10×10. We build the Laplacian pyramid of this image to obtain a new observation image, and then extract the observation mixed signals x_1 and x_2 from these two images based on Eq. (5). Fig. 5 (c) shows the separated signals y_1 and y_2 obtained by applying the signal noise separation based on blind signal separation, which are the estimated main source signal (corresponding to the tampered region) and the estimated noise source signal (corresponding to the background image with traces left by post-processing operations), respectively. As shown in Figs. 5 (d) and 5 (e), converting these signals y_1 and y_2 into images, it can be found that the foreground forged region is separated from the background image, indicating that the signal noise separation helps to reduce the interference of post-processing.

C. Ablation Study

We perform an ablation study to validate the effectiveness of the signal noise separation module, which is utilized for reducing the impact of complex background textures and noise generated by post-processing operations on post-processed image forgery detection. Specifically, we evaluate the performance of our SNIS on localization and classification without and with the signal noise separation module. It should be specified that the architecture composed of the multi-scale feature learning module and the prediction module is equivalent to the architecture without the signal noise separation module.

In real-world scenarios, for hiding the tampering traces of the forged image, Median blur (MB) and Gaussian blur (GB) are two common post-processing operations used to manipulate tampered images. Then, the manipulated image may be compressed with JPEG compression for storage. In this experiment, the testing images selected from NIST16 and Columbia datasets are post-processed with Median blur or Gaussian blur (with kernel size w) followed by JPEG compression (with quality QF).

- For Median blur, the parameter w controls the size of the median convolution kernel. The kernel size must be an odd number greater than 1. We have selected the more commonly used parameter values $\{3, 5, 7, 9, 11\}$ for performance evaluation. The larger the kernel size used, the blurrier the image will be.
- For Gaussian blur, the parameter w controls the size of the Gaussian convolution kernel. The kernel size also must be an odd number greater than 1. We have selected the more commonly used values $\{3, 5, 7, 9, 11\}$ for performance evaluation. The larger the kernel size used, the blurrier the image will be.
- For JPEG compression, the parameter QF controls the compression quality factor. We have selected the more

TABLE III
AP COMPARISON OF ABLATION STUDY ON NIST16 DATASET: OPERATIONS CLASSIFICATION EVALUATION WITH AND WITHOUT (W/OUT) THE SIGNAL NOISE SEPARATION MODULE (SNISM). "MEAN" DENOTES THE MEAN AP FOR SPLICING, COPY-MOVE AND REMOVAL.

Parameters	Method	Post-processed images (MB + JPEG)				Post-processed images (GB + JPEG)			
		Splicing	Copy-Move	Removal	Mean	Splicing	Copy-Move	Removal	Mean
$w = 3$, $QF = 85$	w/out snism	81.17%	93.83%	77.83%	84.46%	79.44%	92.74%	72.96%	81.71%
	with snism	**87.01%**	**99.30%**	**92.52%**	**92.94%**	**85.00%**	**96.21%**	**89.95%**	**90.39%**
$w = 5$, $QF = 85$	w/out snism	83.20%	94.14%	72.84%	83.39%	82.03%	92.51%	74.03%	82.86%
	with snism	**85.71%**	**99.65%**	**91.92%**	**92.43%**	**85.23%**	**98.45%**	**89.36%**	**91.01%**
$w = 7$, $QF = 85$	w/out snism	83.46%	94.44%	69.93%	82.61%	79.89%	93.83%	70.69%	81.47%
	with snism	**85.26%**	**99.88%**	**86.15%**	**90.43%**	**83.94%**	**99.71%**	**92.11%**	**91.92%**
$w = 9$, $QF = 85$	w/out snism	80.95%	93.83%	65.30%	80.03%	**83.43%**	94.44%	65.96%	81.28%
	with snism	**85.99%**	**99.71%**	**82.25%**	**89.31%**	83.35%	**99.71%**	**88.29%**	**90.45%**
$w = 11$, $QF = 85$	w/out snism	80.51%	92.69%	62.52%	78.57%	81.65%	94.44%	60.61%	78.90%
	with snism	**85.05%**	**98.28%**	**80.07%**	**87.80%**	**82.53%**	**99.77%**	**85.80%**	**89.36%**
$w = 3$, $QF = 70$	w/out snism	79.78%	93.53%	76.26%	83.19%	79.34%	92.30%	72.25%	81.30%
	with snism	**85.83%**	**100.00%**	**92.23%**	**92.69%**	**85.32%**	**98.78%**	**93.21%**	**92.44%**
$w = 5$, $QF = 70$	w/out snism	82.61%	94.44%	70.15%	82.40%	79.92%	92.33%	72.67%	81.64%
	with snism	**85.83%**	**99.11%**	**91.12%**	**92.02%**	**84.65%**	**98.89%**	**89.64%**	**91.06%**
$w = 7$, $QF = 70$	w/out snism	81.03%	94.44%	64.67%	80.05%	79.80%	93.47%	68.86%	80.71%
	with snism	**86.40%**	**99.25%**	**87.54%**	**91.06%**	**84.37%**	**99.72%**	**89.81%**	**91.30%**
$w = 9$, $QF = 70$	w/out snism	80.59%	92.38%	62.22%	78.39%	80.26%	93.83%	61.77%	78.62%
	with snism	**85.18%**	**96.37%**	**84.69%**	**88.75%**	**83.48%**	**99.66%**	**87.97%**	**90.37%**
$w = 11$, $QF = 70$	w/out snism	80.36%	92.12%	58.36%	76.95%	81.29%	94.92%	56.46%	77.56%
	with snism	**84.47%**	**97.01%**	**77.83%**	**86.44%**	**83.30%**	**99.88%**	**84.58%**	**89.26%**

TABLE IV
OPERATIONS CLASSIFICATION EVALUATION WITH AND WITHOUT (W/OUT) THE SIGNAL NOISE SEPARATION MODULE (SNISM) ON FAKE IMAGES WITHOUT POST-PROCESSING. THE TESTING IMAGES ARE SELECTED FROM THE NIST16 DATASET. "MEAN" DENOTES THE MEAN AP FOR SPLICING, COPY-MOVE AND REMOVAL.

Method	Splicing	Copy-Move	Removal	Mean
w/out snism	80.40%	95.23%	76.40%	84.01%
with snism	87.83%	99.12%	91.52%	92.82%

commonly used quality factor $\{85\}$. The smaller the compression quality factor, the more image pixels are lost, making the image quality worse. To test image forgery detection performance in lower quality images, we also used parameter value $\{70\}$.

The comparison results of forgery localization and operations classification under different post-processing types and strengths are shown in Tables I-III. We can find that the localization and classification performance of the proposed network with signal noise separation module outperforms without signal noise separation module when the forged images have undergone post-processing operations.

This is because the signal noise separation module can separate the main signal representing the tampered area from the background region with noise caused by the post-processing operations, thereby eliminating the negative effect of the post-processing operations and complex background texture on the forgery detection and improving the detection performance.

In addition, by observing the results in the third column of Tables I and II, it can be found that our SNIS can also achieve great localization performance on fake images without post-processing. Table IV provides the operations classification performance on fake images that have not experienced post-processing operations. The average AP value with the signal noise separation module is 92.82%, which is 8.81% higher than the average AP value without this module. These results indicate that our proposed method can obtain better operations classification performance on forged images without post-processing.

D. Comparison to SOTA Methods on Fake Images with Post-processing

In this experiment, we test the robustness of SNIS against post-processing attacks and compare it with **RGB-N** [16], **Cons-N** [13], **DFCN** [17], **Noiseprint** [43], **ManTra-Net** [44], and **OSN** [46]. The network architectures of **RGB-N**, **Cons-N** and **DFCN** are fed tampered images from NIST16 to train the detection model. **Noiseprint**[1], **ManTra-Net**[2] and **OSN**[3] have not released the training code, thus, we use the released trained models. The training images have not undergone post-processing, while the testing images have experienced post-processing. Table V shows the comparison of F_1 score and AUC between SNIS and the SOTA methods on NIST16, where the quality factor of JPEG compression is 85. We can notice that the proposed SNIS performs the best in terms of

[1]https://github.com/grip-unina/noiseprint
[2]https://github.com/ISICV/ManTraNet
[3]https://github.com/HighwayWu/ImageForensicsOSN

TABLE V
COMPARISONS OF THE IN-DATASET FORGERY LOCALIZATION EVALUATION. THE TESTING IMAGES ARE SELECTED FROM NIST16, AND POST-PROCESSED IMAGES CAN BE GENERATED BY USING MEDIAN BLUR OR GAUSSIAN BLUR FOLLOWED BY JPEG COMPRESSION (WITH QUALITY 85).

Metric	Method	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
		3	5	7	9	11	3	5	7	9	11
F_1	**RGB-N** [16]	0.206	0.216	0.215	0.203	0.214	0.218	0.219	0.205	0.207	0.197
	Cons-N [13]	0.772	0.739	0.672	0.649	0.644	0.753	0.735	0.710	0.707	0.689
	DFCN [17]	0.260	0.259	0.259	0.258	0.258	0.260	0.260	0.260	0.260	0.260
	Noiseprint [43]	0.135	0.133	0.132	0.131	0.130	0.135	0.132	0.130	0.130	0.129
	ManTra-Net [44]	0.149	0.149	0.147	0.145	0.140	0.148	0.148	0.147	0.145	0.144
	OSN [46]	0.293	0.254	0.237	0.225	0.210	0.310	0.289	0.277	0.265	0.249
	SNIS	**0.887**	**0.878**	**0.859**	**0.848**	**0.822**	**0.882**	**0.878**	**0.863**	**0.855**	**0.854**
AUC	**RGB-N** [16]	0.691	0.706	0.699	0.672	0.710	0.703	0.706	0.688	0.674	0.679
	Cons-N [13]	0.954	0.922	0.900	0.887	0.878	0.943	0.926	0.923	0.913	0.913
	DFCN [17]	0.598	0.597	0.596	0.595	0.595	0.598	0.598	0.599	0.600	0.600
	Noiseprint [43]	0.519	0.522	0.502	0.517	0.520	0.519	0.518	0.520	0.520	0.514
	ManTra-Net [44]	0.567	0.567	0.561	0.552	0.537	0.565	0.564	0.562	0.556	0.551
	OSN [46]	0.681	0.675	0.672	0.667	0.659	0.666	0.664	0.659	0.651	0.640
	SNIS	**0.981**	**0.980**	**0.975**	**0.973**	**0.965**	**0.977**	**0.979**	**0.976**	**0.974**	**0.973**

TABLE VI
COMPARISONS OF THE IN-DATASET FORGERY LOCALIZATION EVALUATION. THE TESTING IMAGES ARE SELECTED FROM NIST16, AND POST-PROCESSED IMAGES CAN BE GENERATED BY USING MEDIAN BLUR OR GAUSSIAN BLUR FOLLOWED BY JPEG COMPRESSION (WITH QUALITY 70).

Metric	Method	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
		3	5	7	9	11	3	5	7	9	11
F_1	**RGB-N** [16]	0.223	0.220	0.211	0.218	0.210	0.214	0.221	0.206	0.204	0.201
	Cons-N [13]	0.756	0.691	0.667	0.675	0.624	0.741	0.731	0.697	0.717	0.680
	DFCN [17]	0.260	0.260	0.259	0.258	0.258	0.260	0.260	0.261	0.260	0.260
	Noiseprint [43]	0.131	0.129	0.128	0.128	0.126	0.135	0.133	0.132	0.131	0.130
	ManTra-Net [44]	0.147	0.148	0.147	0.144	0.139	0.147	0.147	0.146	0.146	0.144
	OSN [46]	0.300	0.277	0.262	0.249	0.243	0.287	0.269	0.259	0.250	0.235
	SNIS	**0.876**	**0.873**	**0.851**	**0.822**	**0.822**	**0.878**	**0.873**	**0.862**	**0.850**	**0.851**
AUC	**RGB-N** [16]	0.718	0.718	0.690	0.704	0.707	0.704	0.698	0.674	0.679	0.684
	Cons-N [13]	0.942	0.919	0.906	0.900	0.872	0.922	0.922	0.911	0.927	0.911
	DFCN [17]	0.598	0.597	0.596	0.595	0.596	0.598	0.598	0.599	0.599	0.599
	Noiseprint [43]	0.503	0.503	0.506	0.504	0.503	0.519	0.517	0.517	0.515	0.516
	ManTra-Net [44]	0.567	0.565	0.559	0.548	0.535	0.565	0.565	0.559	0.557	0.552
	OSN [46]	0.665	0.673	0.672	0.662	0.657	0.644	0.636	0.633	0.625	0.614
	SNIS	**0.980**	**0.978**	**0.974**	**0.960**	**0.966**	**0.980**	**0.978**	**0.975**	**0.973**	**0.971**

two performance metrics, outperforming the SOTA methods by 0.468 (i.e., **RGB-N** [16]), 0.108 (i.e., **Cons-N** [13]), 0.490 (i.e., **DFCN** [17]), 0.595 (i.e., **Noiseprint** [43]), 0.567 (i.e., **ManTra-Net** [44]), and 0.457 (i.e., **OSN** [46]) for different post-processing cases.

To verify the effectiveness of our method under different compression ratios, we also conduct tests with a compression factor of 70. Table VI shows the comparison results. From this Table, it is evident that the SNIS performance is 0.460 better than **RGB-N** [16], 0.109 better than **Cons-N** [13], 0.486 better than **DFCN** [17], 0.594 better than **Noiseprint** [43], 0.563 better than **ManTra-Net** [44], and 0.459 better than **OSN** [46] in forged region localization when tampered images have undergone different post-processing operations.

Since **DFCN** [17], **Noiseprint** [43], **ManTra-Net** [44] and **OSN** [46] did not support operation identification, we compare our SNIS with **RGB-N** [16] and **Cons-N** [13] on the verification of semantic tampering operation classification performance. Table VII shows the operation identification comparison results. The proposed SNIS can achieve an average AP of 90.57% and outperform other SOTA methods, which demonstrates that SNIS is more robust to multiple post-processing attacks in terms of semantic focused tampering operation classification.

In the case of image forgery including post-processing, the reason why our SNIS achieves better performance than the existing methods is that we calculate the separation matrix for the mixed signal observed from the forged image based on

TABLE VII

AP COMPARISONS OF THE IN-DATASET OPERATIONS CLASSIFICATION EVALUATION. THE TESTING IMAGES FROM NIST16 ARE POST-PROCESSED BY MEDIAN BLUR OR GAUSSIAN BLUR, AND FOLLOWED BY JPEG COMPRESSION. "MEAN" DENOTES THE MEAN AP FOR SPLICING, COPY-MOVE AND REMOVAL.

Parameters	Method	Post-processed images (MB + JPEG)				Post-processed images (GB + JPEG)			
		Splicing	Copy-Move	Removal	Mean	Splicing	Copy-Move	Removal	Mean
$w = 3$, $QF = 85$	**RGB-N** [16]	79.79%	17.38%	55.49%	50.89%	78.03%	13.46%	57.83%	49.77%
	Cons-N [13]	70.09%	76.67%	70.83%	72.53%	70.15%	85.67%	74.22%	76.68%
	SNIS	**87.01%**	**99.30%**	**92.52%**	**92.94%**	**85.00%**	**96.21%**	**89.95%**	**90.39%**
$w = 5$, $QF = 85$	**RGB-N** [16]	77.80%	20.47%	43.62%	47.30%	76.60%	20.57%	46.61%	47.93%
	Cons-N [13]	68.27%	84.32%	64.68%	72.42%	68.29%	81.32%	73.69%	74.43%
	SNIS	**85.71%**	**99.65%**	**91.92%**	**92.43%**	**85.23%**	**98.45%**	**89.36%**	**91.01%**
$w = 7$, $QF = 85$	**RGB-N** [16]	74.58%	19.01%	29.81%	41.13%	72.79%	11.90%	36.00%	40.23%
	Cons-N [13]	67.56%	78.44%	56.68%	67.56%	68.39%	78.84%	66.60%	71.28%
	SNIS	**85.26%**	**99.88%**	**86.15%**	**90.43%**	**83.94%**	**99.71%**	**92.11%**	**91.92%**
$w = 9$, $QF = 85$	**RGB-N** [16]	71.58%	23.94%	20.51%	38.68%	71.99%	11.80%	34.84%	39.54%
	Cons-N [13]	63.71%	67.56%	48.07%	59.78%	67.77%	81.75%	60.53%	70.01%
	SNIS	**85.99%**	**99.71%**	**82.25%**	**89.31%**	**83.35%**	**99.71%**	**88.29%**	**90.45%**
$w = 11$, $QF = 85$	**RGB-N** [16]	70.25%	14.46%	20.99%	35.23%	70.09%	11.90%	27.32%	36.44%
	Cons-N [13]	64.73%	69.35%	48.33%	60.80%	68.45%	79.54%	58.50%	68.83%
	SNIS	**85.05%**	**98.28%**	**80.07%**	**87.80%**	**82.53%**	**99.77%**	**85.80%**	**89.36%**
$w = 3$, $QF = 70$	**RGB-N** [16]	80.11%	15.92%	53.53%	49.85%	80.33%	16.16%	50.86%	49.12%
	Cons-N [13]	71.06%	84.36%	80.68%	78.70%	68.76%	88.27%	78.75%	78.59%
	SNIS	**85.83%**	**100.00%**	**92.23%**	**92.69%**	**85.32%**	**98.78%**	**93.21%**	**92.44%**
$w = 5$, $QF = 70$	**RGB-N** [16]	77.53%	18.66%	40.27%	45.49%	78.90%	13.84%	40.36%	44.37%
	Cons-N [13]	69.95%	87.66%	61.30%	72.97%	68.48%	88.89%	67.52%	74.96%
	SNIS	**85.83%**	**99.11%**	**91.12%**	**92.02%**	**84.65%**	**98.89%**	**89.64%**	**91.06%**
$w = 7$, $QF = 70$	**RGB-N** [16]	75.15%	19.57%	26.96%	40.56%	76.07%	10.20%	37.08%	41.12%
	Cons-N [13]	67.25%	77.47%	58.30%	67.67%	66.08%	82.03%	61.26%	69.79%
	SNIS	**86.40%**	**99.25%**	**87.54%**	**91.06%**	**84.37%**	**99.72%**	**89.81%**	**91.30%**
$w = 9$, $QF = 70$	**RGB-N** [16]	74.38%	16.21%	22.80%	37.50%	71.55%	13.10%	32.83%	39.16%
	Cons-N [13]	65.25%	67.88%	48.30%	60.48%	64.16%	88.89%	59.63%	70.90%
	SNIS	**85.18%**	**96.37%**	**84.69%**	**88.75%**	**83.48%**	**99.66%**	**87.97%**	**90.37%**
$w = 11$, $QF = 70$	**RGB-N** [16]	69.56%	16.54%	18.38%	34.83%	70.21%	12.74%	24.66%	35.87%
	Cons-N [13]	63.33%	62.93%	47.12%	57.79%	65.40%	77.78%	54.78%	65.99%
	SNIS	**84.47%**	**97.01%**	**77.83%**	**86.44%**	**83.30%**	**99.88%**	**84.58%**	**89.26%**

the principle that the eigen decomposition of the covariance matrix makes the source signals statistically irrelevant, so as to decompose the main source signal and the noise signal, and realize the separation of the tampered region and the background image with post-processing noise. This separation simplifies the detection of semantic tampering traces and optimizes the detection performance.

E. Comparison to SOTA Methods on Fake Images without Post-processing

For forged images without post-processing, we verify the tampered region localization performance of SNIS and compare it with methods **ELA** [5], **NOI1** [6], **CFA1** [8], **RGB-N** [16], **Cons-N** [13] and **DFCN** [17]. Table VIII provides the comparison of F_1 and AUC score between SNIS and these SOTA methods. The results of **ELA**, **NOI1**, and **CFA1** are replicated from the literature [16]. The results of **RGB-N**, **Cons-N**, **DFCN** and SNIS are obtained from models trained on the NIST16 dataset.

TABLE VIII

F_1 SCORE AND AUC COMPARISONS ON THREE STANDARD DATASETS WHEN IMAGES HAVE NOT UNDERGONE POST-PROCESSING OPERATIONS.

Method	NIST16		Columbia		CASIA	
	F_1	AUC	F_1	AUC	F_1	AUC
ELA [5]	0.236	0.429	0.470	0.581	0.214	0.613
NOI1 [6]	0.285	0.487	0.574	0.546	0.263	0.612
CFA1 [8]	0.174	0.501	0.467	0.720	0.207	0.522
RGB-N [16]	0.722	0.937	**0.697**	**0.858**	0.361	**0.766**
Cons-N [13]	**0.917**	**0.989**	0.689	0.747	0.330	0.661
DFCN [17]	0.260	0.600	0.023	0.499	0.030	0.501
SNIS	0.891	0.981	0.681	0.800	**0.368**	0.701

We can clearly see that SNIS outperforms **ELA**, **NOI1**, and **CFA1** on NIST16, Columbia, and CASIA datasets. This is because these methods all focus on forgery localization for a specific operation and only extract partial tampering artifacts, which limits their localization performance in different

TABLE IX
COMPARISONS OF THE CROSS-DATASET EVALUATION. THE TESTING IMAGES ARE DERIVED FROM COLUMBIA AND CASIA, AND POST-PROCESSED IMAGES CAN BE CREATED BY USING MEDIAN BLUR OR GAUSSIAN BLUR FOLLOWED BY JPEG COMPRESSION (WITH QUALITY 85).

Metric	Method	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
		3	5	7	9	11	3	5	7	9	11
	RGB-N [16]	0.471	0.477	0.475	0.471	0.477	0.459	0.473	0.459	0.473	0.474
	Cons-N [13]	0.555	0.568	0.563	0.549	0.574	0.565	0.576	0.558	0.555	0.547
	DFCN [17]	0.025	0.037	0.044	0.047	0.047	0.024	0.024	0.025	0.025	0.024
F_1 comparisons on Columbia	**Noiseprint** [43]	0.437	0.430	0.424	0.422	0.421	0.436	0.430	0.424	0.422	0.420
	ManTra-Net [44]	0.440	0.440	0.440	0.439	0.437	0.440	0.439	0.437	0.436	0.434
	OSN [46]	**0.652**	0.625	0.620	0.618	0.609	**0.652**	0.635	0.618	0.602	0.573
	SNIS	0.651	**0.654**	**0.657**	**0.667**	**0.659**	0.632	**0.640**	**0.653**	**0.644**	**0.648**
	RGB-N [16]	0.570	0.581	0.569	0.572	0.581	0.549	0.570	0.563	0.576	0.579
	Cons-N [13]	0.646	0.673	0.668	0.586	0.617	0.652	0.689	0.663	0.640	0.635
	DFCN [17]	0.500	0.501	0.502	0.503	0.503	0.500	0.500	0.500	0.499	0.499
AUC comparisons on Columbia	**Noiseprint** [43]	0.525	0.527	0.524	0.528	0.524	0.527	0.525	0.522	0.511	0.518
	ManTra-Net [44]	0.519	0.519	0.517	0.515	0.512	0.519	0.516	0.513	0.511	0.508
	OSN [46]	0.768	0.756	0.754	0.755	0.751	**0.771**	**0.765**	0.760	**0.751**	0.734
	SNIS	**0.773**	**0.771**	**0.766**	**0.772**	**0.767**	0.736	0.746	**0.762**	0.740	**0.756**
	RGB-N [16]	0.203	0.198	0.198	0.197	0.195	0.202	0.198	0.199	0.200	0.200
	Cons-N [13]	0.294	0.268	0.259	0.264	0.245	0.285	0.282	0.259	0.235	0.246
	DFCN [17]	0.031	0.032	0.033	0.034	0.033	0.031	0.031	0.032	0.032	0.032
F_1 comparisons on CASIA	**Noiseprint** [43]	0.180	0.170	0.162	0.158	0.156	0.181	0.171	0.161	0.158	0.156
	ManTra-Net [44]	0.198	0.198	0.197	0.197	0.197	0.198	0.198	0.197	0.197	0.197
	OSN [46]	**0.341**	0.237	0.178	0.138	0.118	0.317	0.230	0.136	0.103	0.085
	SNIS	0.325	**0.319**	**0.313**	**0.286**	**0.289**	**0.320**	**0.296**	**0.295**	**0.279**	**0.275**
	RGB-N [16]	0.600	0.593	0.593	0.596	0.586	0.603	0.590	0.589	0.599	0.597
	Cons-N [13]	0.639	0.616	0.606	0.605	0.580	0.627	0.616	0.610	0.585	0.601
	DFCN [17]	0.500	0.500	0.501	0.501	0.501	0.500	0.500	0.500	0.501	0.501
AUC comparisons on CASIA	**Noiseprint** [43]	0.526	0.523	0.529	0.525	0.526	0.525	0.528	0.523	0.530	0.530
	ManTra-Net [44]	0.503	0.503	0.503	0.502	0.502	0.503	0.503	0.502	0.502	0.501
	OSN [46]	**0.676**	0.617	0.584	0.560	0.550	0.657	0.610	0.564	0.546	0.538
	SNIS	0.672	**0.680**	**0.691**	**0.673**	**0.665**	**0.668**	**0.660**	**0.673**	**0.664**	**0.682**

semantic tampering operations. Our SNIS can learn high-level information from multiple perspectives and construct global feature representation by using the multi-scale feature learning module. Thus, SNIS can capture tampering traces of different semantic operations.

The difference between the localization performance of **DFCN** and SNIS is that the training sample generation strategy proposed by [17] has strong pertinence, which limits its performance for natural scene images. In addition, it can be found that the localization performance of SNIS is slightly weaker than that of **RGB-N** and **Cons-N**.

The reason is that in the absence of post-processing operations, there is no post-processing noise in the forged image. For a forged image with relatively smooth textures, the background region has little negative interference to the forgery detection. Therefore, the superiority of using the signal noise separation module to weaken or eliminate such interference to improve the detection performance is not obvious. However, for the complex image forgery process, such as retouching forged images with post-processing, there exist some interference noise signals. The signal noise separation module can eliminate the influence of these noise signals on forgery detection, which greatly improves the detection performance. The results in Tables V and VI illustrate that the detection performance of SNIS is better than **RGB-N** and **Cons-N** when the tampered images have experienced post-processing.

F. Cross-dataset Performance Comparison

To evaluate the robustness of SNIS to cross-dataset, we conduct several experiments that are trained on NIST16 but tested on Columbia and CASIA. The results of cross-dataset detection when the JPEG compression quality factor is 85 are shown in Table IX. For fake images selected from Columbia, we can notice that the average detection performance of SNIS is 0.184 better than **RGB-N**, 0.101 better than **Cons-N**, 0.439 better than **DFCN**, 0.230 better than **Noiseprint**, 0.228 better than **ManTra-Net**, and 0.017 better than **OSN** under different post-processing cases. Besides, for fake images selected from CASIA, the SNIS outperforms the SOTA methods by 0.088 (i.e., **RGB-N**), 0.053 (i.e., **Cons-N**), 0.220 (i.e., **DFCN**), 0.140 (i.e., **Noiseprint**), 0.136 (i.e., **ManTra-Net**), and 0.097 (i.e., **OSN**).

We also test the cross-dataset performance when the JPEG compression quality factor is 70. The results are provided in

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3204753

TABLE X
COMPARISONS OF THE CROSS-DATASET EVALUATION. THE TESTING IMAGES ARE DERIVED FROM COLUMBIA AND CASIA, AND POST-PROCESSED IMAGES CAN BE CREATED BY USING MEDIAN BLUR OR GAUSSIAN BLUR FOLLOWED BY JPEG COMPRESSION (WITH QUALITY 70).

Metric	Method	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
		3	5	7	9	11	3	5	7	9	11
F_1 comparisons on Columbia	**RGB-N** [16]	0.470	0.479	0.476	0.479	0.472	0.473	0.482	0.470	0.473	0.473
	Cons-N [13]	0.542	0.553	0.538	0.549	0.591	0.560	0.547	0.543	0.552	0.537
	DFCN [17]	0.024	0.036	0.044	0.046	0.047	0.022	0.023	0.023	0.023	0.023
	Noiseprint [43]	0.436	0.430	0.424	0.422	0.421	0.436	0.430	0.424	0.423	0.421
	ManTra-Net [44]	0.442	0.440	0.439	0.436	0.436	0.439	0.438	0.436	0.434	0.433
	OSN [46]	0.629	0.615	0.608	0.606	0.600	0.610	0.606	0.589	0.567	0.514
	SNIS	**0.643**	**0.650**	**0.658**	**0.645**	**0.652**	**0.625**	**0.627**	**0.640**	**0.650**	**0.630**
AUC comparisons on Columbia	**RGB-N** [16]	0.575	0.576	0.579	0.589	0.567	0.584	0.591	0.562	0.586	0.575
	Cons-N [13]	0.638	0.649	0.626	0.626	0.669	0.642	0.647	0.625	0.637	0.622
	DFCN [17]	0.500	0.501	0.502	0.503	0.503	0.500	0.500	0.500	0.499	0.499
	Noiseprint [43]	0.529	0.525	0.525	0.527	0.524	0.525	0.521	0.527	0.527	0.525
	ManTra-Net [44]	0.520	0.518	0.516	0.511	0.510	0.517	0.515	0.511	0.509	0.507
	OSN [46]	**0.751**	0.742	0.740	0.736	0.733	**0.745**	**0.745**	**0.742**	0.728	0.703
	SNIS	**0.751**	**0.761**	**0.767**	**0.742**	**0.740**	0.724	0.720	0.737	**0.762**	**0.736**
F_1 comparisons on CASIA	**RGB-N** [16]	0.199	0.202	0.200	0.200	0.199	0.200	0.198	0.200	0.199	0.199
	Cons-N [13]	0.283	0.263	0.265	0.262	0.264	0.273	0.271	0.251	0.232	0.222
	DFCN [17]	0.031	0.032	0.033	0.034	0.032	0.031	0.031	0.032	0.032	0.032
	Noiseprint [43]	0.181	0.171	0.161	0.158	0.156	0.180	0.171	0.166	0.158	0.156
	ManTra-Net [44]	0.198	0.198	0.197	0.197	0.197	0.198	0.198	0.197	0.197	0.197
	OSN [46]	0.242	0.165	0.121	0.091	0.078	0.204	0.156	0.093	0.064	0.050
	SNIS	**0.338**	**0.303**	**0.309**	**0.299**	**0.289**	**0.309**	**0.294**	**0.287**	**0.279**	**0.270**
AUC comparisons on CASIA	**RGB-N** [16]	0.594	0.608	0.600	0.598	0.597	0.591	0.595	0.592	0.597	0.597
	Cons-N [13]	0.619	0.608	0.616	0.605	0.616	0.611	0.613	0.593	0.578	0.581
	DFCN [17]	0.501	0.500	0.501	0.501	0.500	0.501	0.500	0.500	0.501	0.501
	Noiseprint [43]	0.534	0.529	0.524	0.522	0.526	0.533	0.528	0.530	0.528	0.535
	ManTra-Net [44]	0.503	0.503	0.502	0.502	0.502	0.503	0.502	0.502	0.502	0.501
	OSN [46]	0.617	0.579	0.557	0.539	0.531	0.596	0.572	0.541	0.527	0.520
	SNIS	**0.674**	**0.659**	**0.684**	**0.674**	**0.681**	**0.656**	**0.656**	**0.668**	**0.676**	**0.674**

TABLE XI
CROSS-DATASET EVALUATION: THE SNIS MODEL IS TRAINED ON CASIA BUT TESTED ON COLUMBIA, AND THE TESTING IMAGES ARE POST-PROCESSED WITH MEDIAN BLUR OR GAUSSIAN BLUR FOLLOWED BY JPEG COMPRESSION.

Compression parameters	Metric	Post-processed images (MB + JPEG)					Post-processed images (GB + JPEG)				
		3	5	7	9	11	3	5	7	9	11
QF=85	F_1	0.728	0.710	0.689	0.666	0.614	0.663	0.638	0.589	0.575	0.547
	AUC	0.803	0.785	0.768	0.745	0.699	0.736	0.714	0.671	0.662	0.631
QF=70	F_1	0.708	0.673	0.667	0.634	0.593	0.665	0.635	0.605	0.570	0.555
	AUC	0.783	0.755	0.746	0.715	0.679	0.739	0.713	0.686	0.653	0.636

Table X. It can be found that the SNIS achieves competitive cross-dataset localization performance than these SOTA methods. Although SNIS cannot achieve high F_1 scores on the CASIA database, the AUC scores of our SNIS are relatively high under all cases, reaching an average of 0.67. Hence, SNIS still significantly outperforms the existing ones.

To verify the performance of cross-dataset detection when training the network with images from different datasets, we use the CASIA dataset to train the proposed SNIS, and the images from the Columbia dataset are utilized as the testing images. Table XI provides the cross-dataset detection performance using SNIS under different post-processing scenarios. It can be found that the trained model can achieve great localization results.

The aforementioned comparison results illustrate that the performance of SNIS is superior to these SOTA methods in handling the post-processed image forgery detection issue. Meanwhile, the SNIS is robust to cross-dataset detection. That may be because most SOTA models capture tampering artifacts that are weakened by post-processing, which brings a negative impact on the forgery localization. On the contrary, SNIS makes the traces of semantic tampering operation not

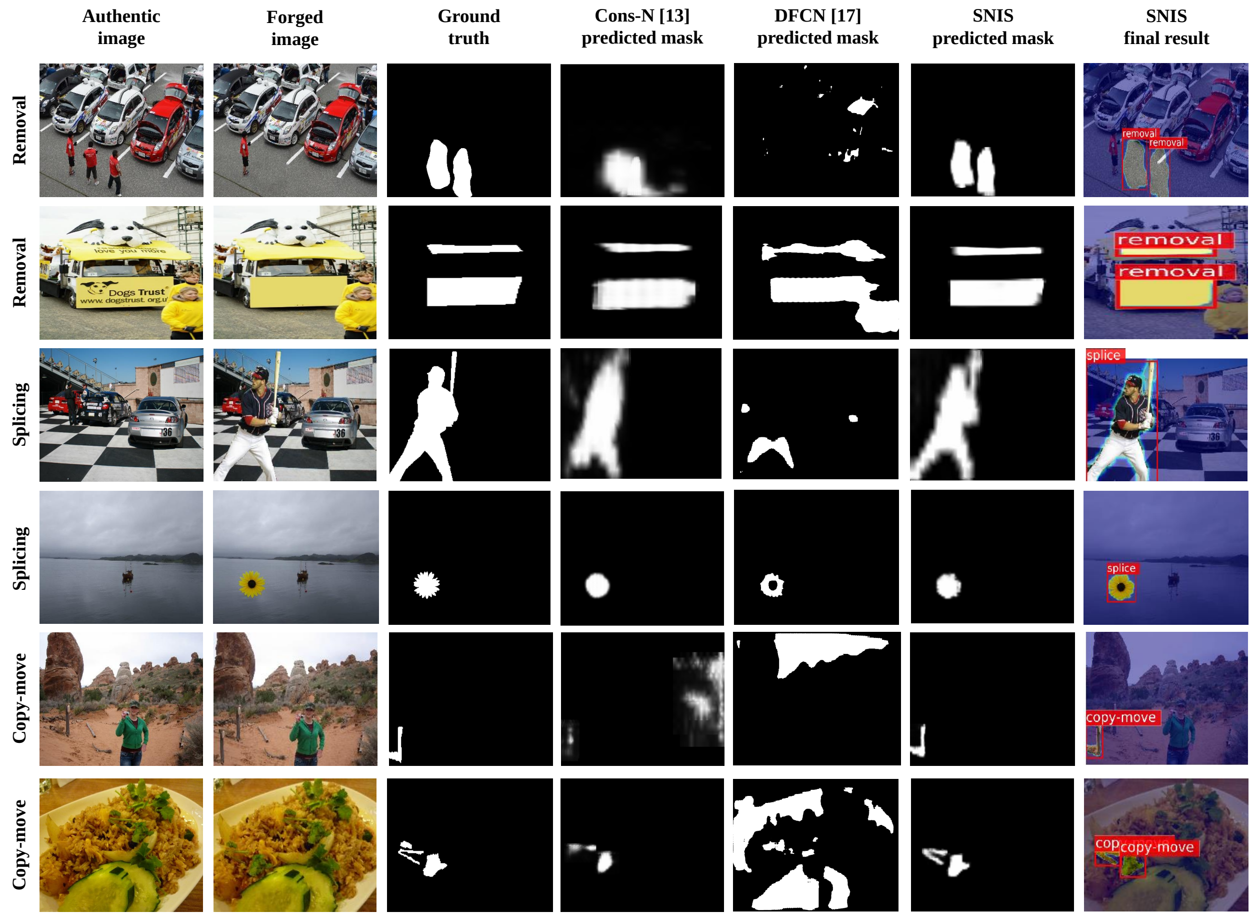


Fig. 6. Qualitative results for post-processed image forgery detection on NIST16 dataset. From left to right: the first two columns are examples of authentic and forged images (the forged images are post-processed by Gaussian blur with kernel size 3 and JPEG compression with quality 70); the third column shows the ground truth, which marks the tampered regions; the fourth column provides the results using **Cons-N**; the fifth column provides the results using **DFCN**; the last two columns provide results using our SNIS network, where the type of tampering operation is displayed in the last column, showing that SNIS not only can locate the tampered area but also identify different semantic tampering operations.

covered by traces of post-processing operations by separating the forged region from the background region with post-processing noise. Furthermore, we focus on multi-scale information learning to explore high-level global features by utilizing the designed parallel atrous convolution architecture, which benefits the forgery detection model.

G. Qualitative Comparison

In Fig. 6, we provide some qualitative results for comparison of our proposed SNIS architecture, **Cons-N**, and **DFCN** in image forgery detection. The images are selected from the NIST16 dataset. It should be noted that the tampered images are post-processed by Gaussian blur (with kernel size 3) and JPEG compression (with quality 70). As shown in the figure, our SNIS is effective to identify different semantic tampering manipulations and can obtain better localization results than other methods in complex post-processing scenarios.

Moreover, comparing the first and second rows, the third and fourth rows respectively, it can be found that the localization performance of **Cons-N** and **DFCN** on fake images with complex background textures is lower than that on the images with simple textures, and our SNIS can achieve good performance in any case. This is because the complex background region has rich information, which is interference information when locating tampered regions and will affect the localization performance. **Cons-N** and **DFCN** directly extract information from the image to learn forensic features, which will also learn interference information, so the forensic performance in forged images with complex backgrounds decreases. SNIS can distinguish the tampered area from the complex background area through the signal noise separation module, which can weaken or even eliminate the interference effect of the complex background. Thus, SNIS can obtain great forensic results.

VI. Conclusion

In this paper, we theoretically illustrate that post-processed image forgery detection can be transformed into signal noise separation. Based on this analysis, we propose a signal noise separation-based (SNIS) network for tampered region localization and semantic tampering operation classification. SNIS introduces a signal noise separation idea to eliminate the visual masking effect of complex background and post-processing operations on forgery localization. Meanwhile, this network is capable to learn multi-scale information via a parallel atrous convolution architecture, which significantly improves the forgery detection performance. Extensive experiments show that SNIS achieves competitive and robustness detection performance in different image forgery cases. As part of our future effort, we will try to extend SNIS to design a new forgery detector that is resistant to more sophisticated post-processing attacks.

REFERENCES

[1] C. Li and Y. Li, "Color-decoupled photo response non-uniformity for digital image forensics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 2, pp. 260-271, Feb. 2012.

[2] C. Qin, E. Liu, G. Feng, and X. Zhang, "Perceptual image hashing for content authentication based on convolutional neural network with multiple constraints," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 11, pp. 4523-4537, Nov. 2021.

[3] T. J. De Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. De Rezende Rocha, "Exposing digital image forgeries by illumination color classification," *IEEE Transactions on Information Forensics Security*, vol. 8, no. 7, pp. 1182-1194, Jul. 2013.

[4] W. Wang, J. Dong, and T. Tan, "Effective image splicing detection based on image chroma," *in Proc. International Conference on Image Processing*, Cairo, Egypt, Nov. 2009, pp. 1257-1260.

[5] Neal Krawetz, "A picture's worth...digital image analysis and forensics," *Hacker Factor Solutions*, 2007.

[6] B. Mahdian and S. Saic, "Using noise inconsistencies for blind image forensics," *Image and Vision Computing*, vol. 27, no. 10, pp. 1497-1503, Sep. 2009.

[7] X. Pan, X. Zhang, and S. Lyu, "Exposing image splicing with inconsistent local noise variances," *in Proc. IEEE International Conference on Computational Photography*, Seattle, WA, USA, Apr. 2012, pp. 1-10.

[8] P. Ferrara, T. Bianchi, A. De Rosa, and A. Piva, "Image forgery localization via fine-grained analysis of cfa artifacts," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1566-1577, Oct. 2012.

[9] H. Li, W. Luo, X. Qiu, and J. Huang, "Image forgery localization via integrating tampering possibility maps," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1240-1252, May 2017.

[10] K. Ye, J. Dong, W. Wang, J. Xu, and T. Tan, "Image forgery detection based on semantic image understanding," *in Proc. CCF Chinese Conference on Computer Vision*, Tianjin, China, Oct. 2017, pp. 472-481.

[11] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics Security*, vol. 10, no. 3, pp. 507-518, Mar. 2015.

[12] R. Salloum, Y. Ren, and C.-C. Jay Kuo, "Image splicing localization using a multi-task fully convolutional network (MFCN)," *Journal of Visual Communication Image Representation*, vol. 51, pp. 201-209, Jan. 2018.

[13] C. Yang, H. Li, F. Lin, B. Jiang, and H. Zhao, "Constrained R-CNN: A general image manipulation detection model," *in Proc. IEEE International Conference on Multimedia Expo*, London, UK, Jul. 2020, pp. 1-6.

[14] 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 Transactions on Image Processing*, vol. 28, no. 7, pp. 3286-3300, Jul. 2019.

[15] X. Liao, K. Li, X. Zhu, and K. J. Ray Liu, "Robust detection of image operator chain with two-stream convolutional neural network," *IEEE Journal of Selected Topics Signal Processing*, vol. 14, no. 5, pp. 955-968, Jun. 2020.

[16] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," *in Proc. IEEE/CVF Conference on Computer Vision Pattern Recognition*, Salt Lake City, UT, USA, Jun. 2018, pp. 1053-1061.

[17] P. Zhuang, H. Li, S. Tan, B. Li, and J. Huang, "Image tampering localization using a dense fully convolutional network," *IEEE Transactions on Information Forensics Security*, vol. 16, pp. 2986-2999, Apr. 2021.

[18] X. Kang, M. C. Stamm, A.Peng, and K. J. R. Liu, "Robust median filtering forensics using an autoregressive model," *IEEE Transactions on Information Forensics Security*, vol. 8, no. 9, pp. 1456-1468, Sep. 2013.

[19] C. Chen, J. Ni, and J. Huang, "Blind detection of median filtering in digital images: A difference domain based approach," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4699-4710, Dec. 2013.

[20] W. Xu, J. Mulligan, D. Xu, and X. Chen, "Detecting and classifying blurred image regions," *in Proc. IEEE International Conference on Multimedia Expo*, San Jose, CA, Jul. 2013, pp. 1-6.

[21] C. Tang, X. Liu, S. An, and P. Wang, "BR^{2}Net: Defocus blur detection via a bidirectional channel attention residual refining network," *IEEE Transactions on Multimedia*, vol. 23, pp. 624-635, Apr. 2021.

[22] 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, Feb. 2005.

[23] Z. Fan and R. L. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 230-235, Feb. 2003.

[24] W. Li, X. Li, R. Ni, and Y. Zhao, "Quantization step estimation for JPEG image forensics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 7, pp. 4816-4827, Jul. 2022.

[25] Y. Li and J. Zhou, "Fast and effective image copy-move forgery detection via hierarchical feature point matching," *IEEE Transactions on Information Forensics Security*, vol. 14, no. 5, pp. 1307-1322, May 2019.

[26] B. Liu and C.-M. Pun, "Locating splicing forgery by adaptive-SVD noise estimation and vicinity noise descriptor," *Neurocomputing*, vol. 38, pp. 172-187, Apr. 2020.

[27] H. Li, W. Luo, X. Qiu, and J. Huang, "Identification of various image operations using residual-based features," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 1, pp. 31-45, Jan. 2018.

[28] 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, Nov. 2018.

[29] Y. Chen, Z. Wang, Z. J. Wang, and X. Kang, "Automated design of neural network architectures with reinforcement learning for detection of global manipulations," *IEEE Journal of Selected Topics Signal Process.*, vol. 14, no. 5, pp. 997-1011, Aug. 2020.

[30] G. Singh and P. Goyal, "GIMD-Net: An effective general-purpose image manipulation detection network, even under anti-forensic attacks," *in Proc. International Joint Conference on Neural Networks*, Shenzhen, China, Jul. 2021, pp. 1-8.

[31] Y. Zhan, Y. Chen, Q. Zhang, and X. Kang, "Image forensics based on transfer learning and convolutional neural network," *in Proc. ACM Conference on Information Hiding and Multimedia Security*, Philadelphia, PA, USA, Jun. 2017, pp. 165-170.

[32] J. Yang, G. Zhu, Y. Luo, S. Kwong, X. Zhang, and Y. Zhou, "Forensic analysis of JPEG-domain enhanced images via coefficient likelihood modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1006-1019, Mar. 2022.

[33] X. Liu, W. Lu, Q. Zhang, J. Huang, and Y. Shi, "Downscaling factor estimation on pre-JPEG compressed images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 618-631, Mar. 2020.

[34] J. Wang, H. Wang, 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 Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2736-2749, Aug. 2020.

[35] H. Wang, J. Wang, X. Luo, Y. Zheng, B. Ma, J. Sun, and S. Kr. Jha, "Detecting aligned double JPEG compressed color image with same quantization matrix based on the stability of image," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 4065-4080, Jun. 2022.

[36] J. You, Y. Li, J. Zhou, Z. Hua, W. Sun, and X. Li, "A transformer based approach for image manipulation chain detection," *in Proc. ACM International Conference on Multimedia*, Chengdu, China, Oct. 2021, pp. 3510-3517.

[37] X. Chu, Y. Chen, and K. J. R. Liu, "Detectability of the order of operations: An information theoretic approach," *IEEE Transactions on Information Forensics Security*, vol. 11, no. 4, pp. 823-836, Apr. 2016.

[38] S. Gao, X. Liao, and X. Liu, "Real-time detecting one specific tampering operation in multiple operator chains," *Journal of Real-Time Image Processing*, vol. 16, no. 3, pp. 741-750, Mar. 2019.

[39] J. Chen, X. Liao, and Z. Qin, "Identifying tampering operations in image operator chains based on decision fusion," *Signal Processing Image Communication*, vol. 95, pp. 1-10, Apr. 2021.

[40] J. Chen, X. Liao, W. Wang, and Z. Qin, "A features decoupling method for multiple manipulations identification in image operation chains," *in Proc. IEEE International Conference on Acoustics, Speech Signal Processing*, Toronto, ON, Canada, Jun. 2021, pp. 2505-2509.

[41] X. Liao, Z. Huang, L. Peng, and T. Qiao, "First step towards parameters estimation of image operator chain," *Information Sciences*, vol. 575, pp. 231-247, Jun. 2021.

[42] M. Kwon, I. Yu, S. Nam, and H. Lee, "CAT-Net: Compression artifact tracing network for detection and localization of image splicing," *in Proc. IEEE Winter Conference on Applications of Computer Vision*, Waikoloa, HI, USA, Jan. 2021, pp. 375-384.

[43] D. Cozzolino and L. Verdoliva, "Noiseprint: A CNN-based camera model fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 2, pp. 144-159, May 2020.

[44] 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 Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, Jun. 2019, pp. 9535-9544.

[45] Y. Rao and J. Ni, "Self-supervised domain adaptation for forgery localization of JPEG compressed images," *in Proc. IEEE/CVF International Conference on Computer Vision*, Montreal, QC, Canada, Oct. 2021, pp. 15014-15023.

[46] H. Wu, J. Zhou, J. Tian, and J. Liu, "Robust image forgery detection over online social network shared images," *in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, Louisiana, Jun. 2022, pp. 13440-13449.

[47] C. Jutten and J. Herault, "Space or time adaptive signal processing by neural network models," *in Proc. International Conference on Neural Network Computing*, Mar. 1987, pp. 206-211.

[48] A. Hyvärinen, "A family of fixed-point algorithms for independent component analysis," *in Proc. IEEE International Conference on Acoustics, Speech Signal Processing*, Munich, Germany, Apr. 1997, pp. 3917-3920.

[49] A. Tharwat, "Independent component analysis: An introduction," *Applied Computing and Informatics*,vol. 17, no. 2, pp. 222-249, Aug. 2020.

[50] K. He, X. Zhang, S., and J. Sun, "Deep residual learning for image recognition," *in Proc. IEEE/CVF Conference on Computer Vision Pattern Recognition*, Las Vegas, NV, USA, Jun. 2016, pp. 770-778.

[51] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," *in Proc. European Conference on Computer Vision*, Munich, Germany, Sep. 2018, pp. 3-19.

[52] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, Jun. 2017.

[53] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," *in Proc. European Conference on Computer Vision*, Zurich, Switzerland, Sep. 2014, pp. 740-755.

[54] Nist manipulation evaluation dataset, 2016. https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation.

[55] Columbia image splicing detection evaluation dataset, 2004. http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataSet/AuthSplicedDataSet.htm.

[56] J. Dong, W. Wang, and T. Tan, "Casia image tampering detection evaluation database," *in Proc. IEEE China Summit International Conference on Signal Information Processing*, Beijing, China, Jul. 2013, pp. 422-426.

Jiaxin Chen received the B.S. degree in software engineering from Central China Normal University, Wuhan, Hubei, China, in 2017. Currently, she is pursuing the Ph.D. degree in the College of Computer Science and Electronic Engineering at Hunan University. Her research focuses on multimedia forensics and artificial intelligence.

Xin Liao received the B.E. and Ph.D. degrees in information security from Beijing University of Posts and Telecommunications in 2007 and 2012, respectively. He is currently an Associate Professor and a Doctoral Supervisor with Hunan University, China. He worked as a Post-Doctoral Fellow with the Institute of Software, Chinese Academy of Sciences, and also a Research Associate with The University of Hong Kong. From 2016 to 2017, he was a Visiting Scholar with the University of Maryland, College Park, USA. His current research interests include multimedia forensics, steganography, and watermarking. He is a member of Technical Committee (TC) on Multimedia Security and Forensics of AsiaCPacific Signal and Information Processing Association, TC on Computer Forensics of Chinese Institute of Electronics, and TC on Digital Forensics and Security of China Society of Image and Graphics. He is serving as an Associate Editor for the IEEE Signal Processing Magazine. He is a senior member of the IEEE.

Wei Wang received the B.S. degree in Computer Science and Technology from North China Electric Power University, in 2007, and Ph.D. degree in Pattern Recognition from the Institute of Automation, Chinese Academy of Sciences (CASIA) in 2012. He is currently an associate professor of National Laboratory of Pattern Recognition (NLPR), CASIA. He is a member of IEEE, CCF (China Computer Federation), CSIG (China Society of Image and Graphics), etc. He is also a member of technical committee (TC) on Computer Vision of CCF, TC on Digital Forensics and Security of CSIG, etc. His current research interests include artificial intelligence and its security problem, image/video forensics and steganalysis, and information content security.

Zhenxing Qian received the B.S. and Ph.D. degrees from the University of Science and Technology of China (USTC), in 2003 and 2007, respectively. He is currently a Professor with the School of Computer Science, Fudan University. He has published more than 100 peer-reviewed articles on international journals and conferences. His research interests include information hiding, image processing, and multimedia security.

Zheng Qin received the Ph.D. degree in computer software and theory from Chongqing University, China, in 2001. From 2010 to 2011, he served as a Visiting Scholar at the Department of Computer Science, Michigan University. He is a professor in the College of Computer Science and Electronic Engineering, Hunan University, where he serves as the vice dean. He also serves as the director of Hunan Key Laboratory of Big Data Research and Application, the vice director of Hunan Engineering Laboratory of Authentication and Data Security. He is a member of China Computer Federation (CCF) and IEEE, respectively. His main interests are network and data security, privacy, data analytics and applications, machine learning, and applied cryptography.

Yaonan Wang received the Ph.D. degree in electrical engineering from Hunan University, Changsha, China, in 1994. He was a Postdoctoral Research Fellow with the Normal University of Defence Technology, Changsha, China, from 1994 to 1995. From 1998 to 2000, he was a Senior Humboldt Fellow in Germany, and, from 2001 to 2004, he was a Visiting Professor with the University of Bremen, Bremen, Germany. Since 1995, he has been a Professor with the College of Electrical and Information Engineering, Hunan University. His current research interests include robotics and image processing.