IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 January 2024, accepted 23 January 2024, date of publication 30 January 2024, date of current version 12 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3359991

RESEARCH ARTICLE

RIFD-Net: A Robust Image Forgery Detection Network

WUYANG SHAN, (Member, IEEE), DENG ZOU, (Member, IEEE), PENGBO WANG, JINGCHUAN YUE, AOLING LIU, AND JUN LI, (Member, IEEE)
College of Computer Science and Cyber Security, Chengdu University of Technology, Chengdu 610059, China
Corresponding author: Wuyang Shan (shanwuyang@cdut.edu.cn)

ABSTRACT Image splicing forensic technologies reveal manipulations that add or remove objects from images. However, the performance of existing splicing forensic methods is fatally degraded when detecting noisy images, as they often ignore the influence of image noise. In this paper, we propose a new forgery detection network called the robust image forgery detection network (RIFD-Net) based on convolutional neural networks (CNNs). With the help of multi-classifiers and a denoising network, RIFD-Net can effectively filter out multiple types of image noise before forgery detection. To determine the extent of tampering, we follow the Siamese network to calculate the similarity between two image patches, without prior knowledge of forensic traces. Results from extensive experiments on benchmark datasets indicate that our method outperforms existing image splicing forensic methods, achieving a substantial improvement of over 20% in the mean average precision (mAP) for forgery detection. Furthermore, RIFD-Net accurately locates splice areas, even in the presence of noise.

INDEX TERMS Splicing forensics, forgery detection, image denoising, convolutional neural networks.

I. INTRODUCTION

Image authenticity comes into focus when encountering deepfakes on social media [1]. Image authenticity identification specifically refers to the scientific judgment of whether an image has undergone post-processing (or tampering) using technical means. Digital image splicing refers to a method for tampering in which two or more digital images are cropped and merged to produce a new composite image [2]. Image splicing is a commonly used technique used to modify crucial data. Spliced images can tamper with faces, alter billing information, and change official reports to deceive individuals. To validate the authenticity of spliced images, forensic researchers have developed several identification methods, including traditional [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] and deep learning-based [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27] techniques.

Traditional tamper detection methods for splicing identify discrepancies between forensic traces in the original and tampered regions. Inconsistencies include, but are not limited to, inconsistent edges [3], [4], double JPEG compression effects [5], [6], [7], [8], [9], inconsistent lighting [10], and variances in the camera imaging process [11], [12], [13], [14], [15]. Splicing tamper detection methods must meet all required prerequisites.

Recently, deep learning-based methods, particularly CNNs, have been effectively utilized for detecting image tampering via splicing. CNN-based forgery detection techniques are distinct from conventional methods, as they employ multiple series or parallel neural networks to pre-process, extract, and classify images; thus, using a stochastic gradient descent algorithm to optimize the network model. Most CNN-based splicing detection networks operate on either pixel-level [16], [17] or patch-level detection [18], [19], [20], [21], [22], [23], [24], [25], [26], [27]. Wu et al. [17] proposed a CNN-based blind forensics technique for image splicing that, deploys a high-pass filter to preprocess images and reduce the influence of image content on splicing detection. This approach enables the classification of authentic and tampered images. Patch-level detection techniques have been developed over time. Bondi et al. [18] determined if there were two or

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek.

more instances of camera model image patches within an image, to identify tampering by training image patches from a significant number of camera models. Ding et al. [19] introduced a novel image tamper localization method based on a dual-channel U-Net architecture. The detection framework consists of an encoder, feature fusion, and a decoder, where high-pass filters extract tampered image residuals, and a dual-channel encoding network processes and fuses deep features for precise localization. You et al. [20] proposed the SGICD-CF, which relies on sample guidance and CNN features specific to individual camera devices. The method involves partitioning the test image into 64×64 pixel patches, extracting camera-related features and model information using the proposed source camera identification network (SCI-Net), and determining tampering by assessing classification confidence and identifying foreign pixels. Typically, the Siamese network [21], wherein weights are conjoined and shared, is employed to extract features concurrently from two image patches. Huh and Liu et al [22] trained a Siamese network to select two random patches from separate images and identify if they have consistent metadata [23]. Consequently, the problem of image splicing detection is formulated as a binary classification task. Cozzolino et al. [24] introduced the ''noiseprint'' approach that extracts residual noise from two image patch sequences, i.e., noiseprint. In this method, the Siamese network optimizes the extraction of noise fingerprints by continuously updating distance weights in reverse. Mayer and Stamm [25], [26] developed a model based on training a considerable number of image patches from various camera models. The model can compute the similarity of the forensic traces between two image patches. By utilizing a Siamese network, the model extracts features from both image patches in parallel, eliminating the need for prior knowledge in determining forensic similarity.

Nevertheless, these CNN-based methods are effective only in the absence of image noise. During image acquisition, various noises are introduced due to the sensor material properties, working environment, electronic components, and circuit structure. The imperfections in the transmission media and recording equipment further pollute digital images during signal transmission. Image processing may also introduce noise when the input object deviates from expectations [28]. Salt and pepper noise (S&P noise) may occur during sudden interference, analog-to-digital conversion, or bit transmission errors. Gaussian noise is generated in insufficiently bright and unevenly illuminated image sensor fields, circuit component noise, and prolonged high-temperature sensor operation [29]. Gamma noise, uniform noise, and Poisson noise have origins similar to Gaussian noise. Random noise results from the accumulation of randomly generated fluctuations over time, making its value unpredictable at a given instant [28]. Image noise can significantly hinder correct feature extraction, and its impact varies depending on the noise type. Noise introduction during image acquisition and transmission severely impairs the accuracy of forgery detection.

Image denoising has emerged as a potential solution to mitigate the impact of image noise interference. Contemporary denoising techniques predominantly leverage deep networks, gaining popularity for their superior accuracy compared with traditional filter-based methods [30], [31], [32], [33]. DnCNN [31] focuses on learning residuals from clean and noisy images, primarily addressing Gaussian noise. FFDnet [32] extends DnCNN's capabilities by training the model with noisy images of varying intensities, making it more adept at eliminating complex Gaussian noise. CBDnet [33] enhances generalization by combining synthetic and real noisy images during model training. However, existing techniques, including CBDnet, face limitations in effectively removing multiple types of noise. The generalization ability of current denoising methods remains constrained by the diversity of noise types.

We propose a novel denoising model aimed at removing various types of noise, which features a noise prediction module and a denoising module. The precision of noise identification is crucial for the noise prediction module, and to enhance accuracy, we integrated an improved classifier utilizing 9 Inception modules [34]. In the denoising module, we leverage the attention mechanism [35] to train models tailored to different types of noise, thereby aiding CNN models in selecting effective features for diverse learning tasks. In addition, the use of sparse block achieves a balance between denoising and retaining essential features.

Following the denoising model, we designed a forensic module to determine the authenticity of an image. Because the ''Forensic Similarity'' method (which does not require prior knowledge of forensic traces) is applicable [25], we utilize a Siamese network to calculate the similarity between two image patches, thereby determining the extent of tampering. Due to the denoising model's ability to produce clean images, the forensic module can acquire precise forensic traces, which significantly enhances the accuracy of detecting forgeries.

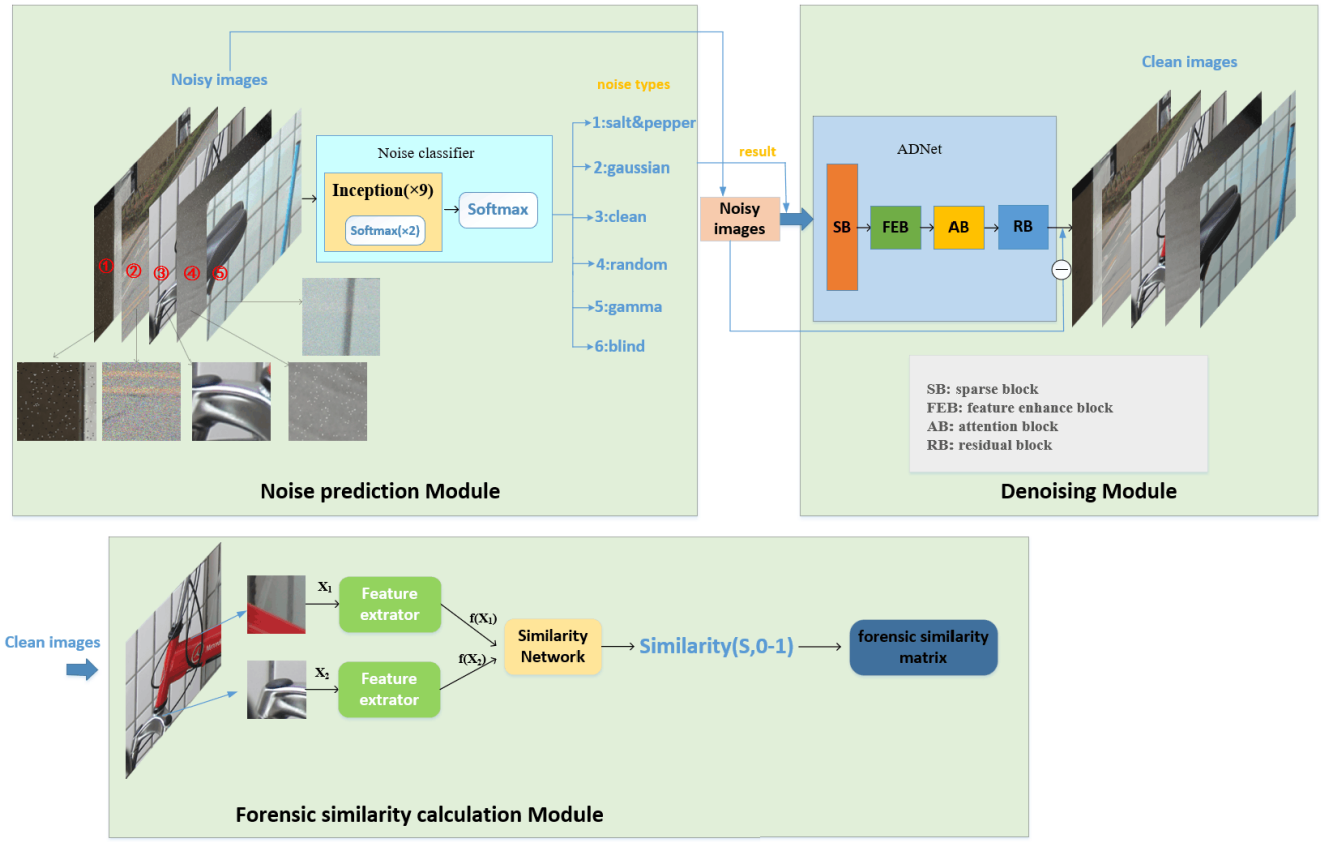
Our proposed method, RIFD-Net, comprises three modules: the noise prediction module, denoising module, and forensic module. The noise prediction and denoising modules remove the interference caused by various image noises and generate clean images. The forensic module extracts possible forensic traces from the images, enabling accurate detection of spliced images.

Our paper makes the following significant contributions:

(1) The proposed RIFD-Net significantly enhances the mAP value of forgery detection by over 20% in noisy scenarios compared with existing methods.

(2) The proposed RIFD-Net can remove different types of noise, distinguishing itself from traditional denoising methods characterized by restricted generalization capabilities, which may not be effective in various noisy scenarios.

(3) The proposed RIFD-Net can maintain a certain balance in the denoising process to ensure that the low-level image features required for forgery detection are erased as little as possible.



In brief: **The input of RIFD-Net is an image. The noise prediction module determines whether the image contains noise and, if so, the type of noise present. The prediction results are then passed to the denoising module. The basic structure of the denoising module is ADNet. ADNet, based on the predicted noise type, selects the corresponding denoising model to generate a clean image. The forensic module divides the clean image into patches, computes the similarity (S) between two image patches at a time, and ultimately generates a forensic similarity matrix for the entire image. The result of the forgery detection is determined by the final output forensic similarity matrix.**

FIGURE 1. RIFD-Net architecture.

The rest of this paper is organized as follows: Section II proposes the RIFD-Net architecture, which is composed of a noise prediction module, a denoising module, and a forensic module, and introduces the selection and role of each of these three modules in detail. The experiments in Section III verify the effectiveness of this combination in forgery detection and its robustness against noise. Finally, Section IV draws the conclusion, and Section V proposes future research possibilities for the area of image forensics.

II. PROPOSED RIFD-NET FOR FORGERY DETECTION

This section provides a detailed description of the proposed RIFD-Net architecture. The RIFD-Net model comprises a noise prediction module, a denoising module, and a forensic module.

A. NETWORK ARCHITECTURE

Fig. 1 presents the RIFD-Net architecture proposed in this paper. To effectively remove various noises, we propose a new denoising model that comprises noise prediction and denoising modules. We trained six dedicated denoising models for four common noise types, a clean environment, and a blind noise type in the denoising module.

The running process of RIFD-Net is described in the brief of the graphical abstract in Fig. 1. An example of the similarity matrix is shown in Fig. 4. For blind noise, if the predicted probabilities for the four known noises and clean situations are exceedingly low, it is classified as blind noise. Detailed insights into each module are provided in subsequent sections.

B. NOISE PREDICTION MODULE

This section describes our noise classifier design for predicting noise types. Many deep neural networks utilize multiple convolutional kernels to extract image features, enabling effective computation and classification based on distinct features—a simple yet powerful technique for image classification. However, increasing network depth within deep networks leads to a significant parameter increase, result-

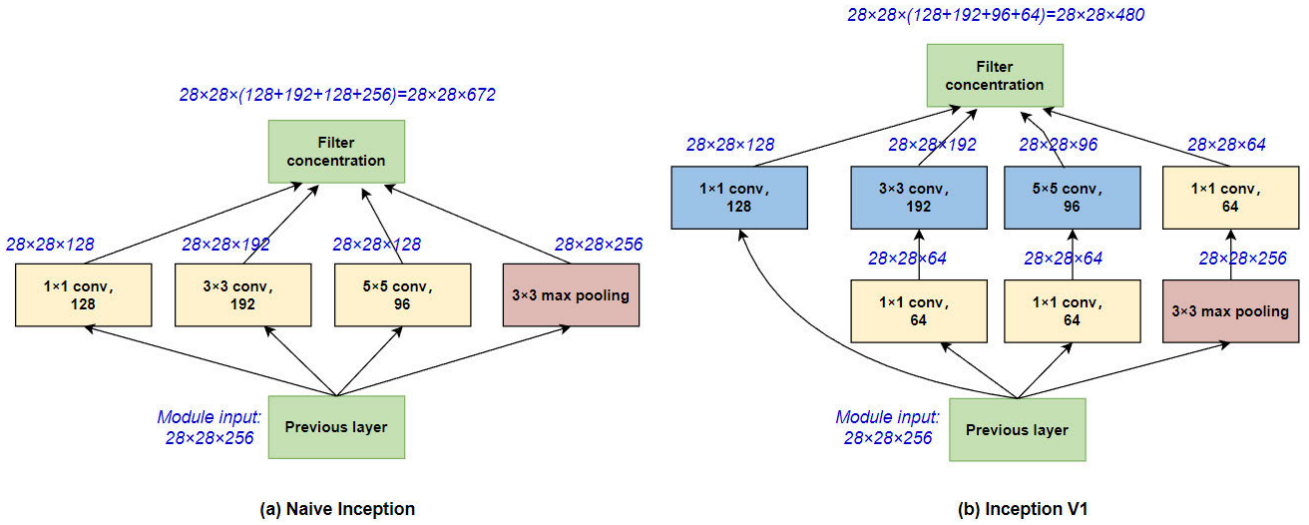


FIGURE 2. Naïve Inception and Inception-v1 structure.

ing in an excessively large network size, prolonged training times, and gradient vanishing issues.

To efficiently predict noise types, we adopt the GoogleNet architecture [34], with a pivotal component being the Inception V1 structure, as shown in Fig. 2(b). The initial concept behind GoogleNet involved stacking various convolution kernels (1 × 1, 3 × 3, 5 × 5, 3 × 3 Pool) instead of a single 3 × 3 convolution kernel. This aims to maintain consistent dimension post-convolution and pooling while summing the channels. This approach diversifies the extracted features, mitigating significant inter-feature correlations. Ultimately, feature maps are concatenated to widen the network, and the Inception module is stacked to increase the network depth. However, a simplistic implementation leads to an explosive increase in computational complexity for each layer. In the native inception (depicted in Fig. 2(a)), all convolution kernels operate on the entire output of the preceding layer, and the computational demand of the 5 × 5 convolution kernel is notably high, resulting in a substantial thickness of the feature maps. To address this issue, 1 × 1 convolution kernels were added before the 3 × 3, before the 5 × 5, and after the max pooling operations, respectively. This addition effectively reduces the thickness of the feature maps, resulting in the network structure of Inception V1 (depicted in Fig. 2(b)).

Assuming that the input feature map size is 28 × 28×256, and the output feature map size is 28 × 28×480, the computational load of the native Inception Module is 854 million. In contrast, Inception V1 reduces the computational load to 358 million, as illustrated in Table 1. In addition, Inception V1 achieves multi-scale feature extraction by stacking multiple convolution kernels of different sizes, which helps the network capture and utilize image features at different scales. As a result, GoogleNet with Inception V1 performs well in noise prediction. We chose a clean environment and four types of noise—S&P, Gaussian, gamma, and random noise as labels for denoising models. Five models were trained to correspond to five labels, and all of them achieved a prediction accuracy of more than 90% on three datasets.

TABLE 1. Comparison of parameters and calculation amount between naive Inception and Inception V1.

Naive Inception Conv Ops		Inception V1 Conv Ops	
[1×1 conv,128]	28×28×128×1×1×256	[1×1 conv,64]	28×28×64×1×1×256
[3×3 conv,192]	28×28×192×3×3×256	[1×1 conv,64]	28×28×64×1×1×256
[5×5 conv,96]	28×28×96×5×5×256	[1×1 conv,128]	28×28×128×1×1×256
		[3×3 conv,192]	28×28×192×3×3×256
		[5×5 conv,96]	28×28×96×5×5×256
		[1×1 conv,64]	28×28×64×1×1×256
Total: 854M ops		Total: 358M ops	

C. DENOISING MODEL

Mainstream denoising networks concentrate on certain types of noise, whereas the attention mechanism extracts different noise information from complex backgrounds. Therefore, we opted to use attention-guided CNN (ADNet) [36] for image denoising. The denoising module utilizes a sparse block (SB), feature enhancement block (FEB), attention block (AB), and reconstruction block (RB). These blocks work together to improve denoising. Fig. 5 illustrates the structure of the 17-layer ADNet denoising module.

$$O_{SB} = f_{SB}(I_N) \tag{1}$$

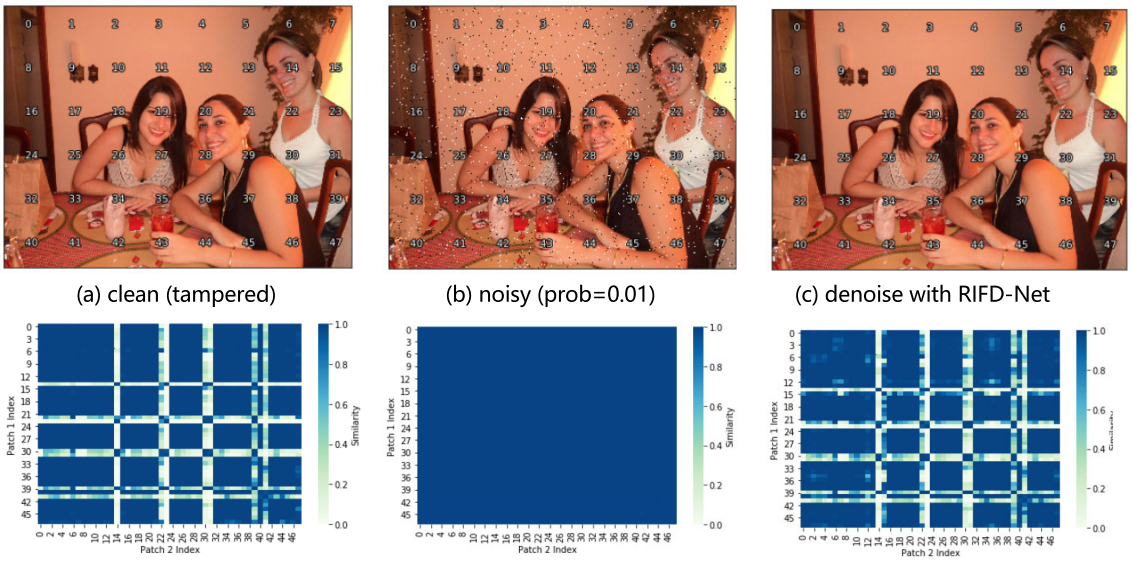
$$O_{FEB} = f_{FEB}(I_N, O_{SB}) \tag{2}$$

$$I_R = f_{AB}(O_{FEB}) \tag{3}$$

$$I_{LC} = I_N - I_R = I_N - f_{AB}\left(f_{FEB}\left(I_N, f_{SB}\left(I_N\right)\right)\right) = I_N - f_{ADNet}(I_N) \tag{4}$$

(a) clean (b) noisy (c) denoise with RIFD-Net

FIGURE 3. An example before and after denoising.



(a) clean (tampered) (b) noisy (prob=0.01) (c) denoise with RIFD-Net

(d) similarity matrix for Fig. (a) (e) similarity matrix for Fig. (b) (f) similarity matrix for Fig. (c)

FIGURE 4. An example of a similarity matrix before and after denoising. Figs. (a)-(c) depict spliced areas, with the woman on the far-right representing them and the numbers indicating the patch index. Correspondingly, Figs. (d)-(f) are confusion matrices displaying the similarity scores of Figs. (a)-(c), in which blue indicates high similarity, and white indicates low similarity.

$$l(\theta) = \frac{1}{2N}\sum_{i=1}^{N} ||f_{ADNet}\left(I_N^i\right) - (I_N^i - I_C^i)||^2 \tag{5}$$

Formulas (1)-(4) represent the input and output of the four blocks. I_N and I_R represent the input noisy image and the predicted residual image, respectively. f represents the functions of each of the four blocks. The brackets represent the input for each block, and the O represents the corresponding output. Finally, we calculate the difference between the noisy image and the clean image as the true residual, and subtract the predicted residual $f_{ADNet}\ (I_N)$ from the true residual as the loss function, as shown in formula (5), where θ stands for parameters in training the denoising model.

In this paper, forensic traces provide important information to reveal splicing inconsistencies. The splicing parts of a spliced image may come from different cameras, which will produce inconsistent images. We combined the steganalysis rich model (SRM) filter [37] and the constrained convolution layer [38] to visualize this splicing inconsistency before and after denoising, as shown in Fig. 6 below. SRM uses image-based statistical features and frequency domain analysis to identify abnormal patterns or inconsistencies in statistical features in images. Ordinary CNNs tend to learn the content of images, and are not suitable for learning content-independent tampering traces. Constrained CNN can learn low-level operational features. In Fig. 6, the first column displays spliced images, including a pristine image and the images with various types of added noise. The second column represents the low-level features of the tampered images obtained using SRM and constrained CNN (to represent forensic traces). The remaining columns depict the forensic traces obtained after processing the noisy images using different denoising methods. It is evident that images with noise exhibit hardly any useful features for forgery detection. After denoising with RIFD-Net, the forensic traces are partially restored, particularly in the case of S&P noise and random noise. Furthermore, RIFD-Net outperforms other denoising methods in preserving low-level image features, which is attributed to the SB in the denoising module.

To capture more contextual information, it is common to enlarge the receptive field during the convolution process. However, expanding the receptive field by increasing the depth and width of the network often results in excessive denoising strength and a rise in the complexity of the

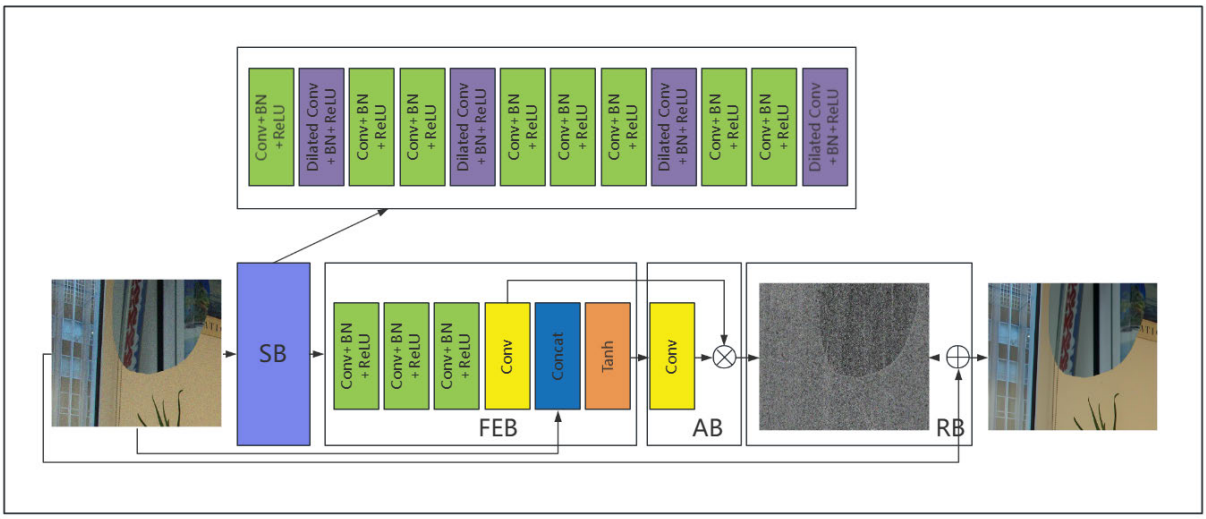


FIGURE 5. Network architecture of ADNet.

denoising model. Therefore, we opt to augment the receptive field through dilated convolutions [36], enabling an increase in size without the need to escalate the depth and width of the network. In SB, the sparse mechanism combining dilated convolution and normal convolution achieves a balance between denoising and preserving low-level features [40], [41]. The 12-layer SB includes two types: dilated Conv+BN+ReLU and Conv+BN+ReLU. Dilated Conv+BN+ReLU denotes dilated convolution with a dilation factor of 2, followed by batch normalization (BN) and Rectified Linear Unit (ReLU) activation. Another type is normal convolution with BN and ReLU. Dilated Conv+BN+ReLU is placed in the second, fifth, ninth, and twelfth layers of ADNet. It is noteworthy that dilated convolution can capture more contextual information [39]. On the basis of this idea, these layers can be considered as high-energy points. Conv+BN+ReLU is set in the first, third, fourth, sixth, seventh, eighth, tenth, and eleventh layers of ADNet and can be viewed as low-energy points. The combination of several high and low-energy points forms the sparse mechanism [36]. The sparse mechanism employs fewer high-energy points to capture more useful information while reducing the complexity of denoising, allowing the preservation of as many low-level features as possible. For the coefficients obtained from sparse coding, denoising can be achieved by filtering out high-frequency or unimportant components. This is because, for sparse representation, noise often manifests as high-frequency or irregular components, whereas the signal exhibits smoother and more regular components.

FEB makes full use of global and local features through a long path to mine more robust features and concatenates noisy images with deeper outputs at the deeper layers, moderating the weakening effect of the shallow layers on the deeper layers in a deeper network. Specifically, 4-layer FEB consists of three types: Conv+BN+ReLU, Conv, and Tanh, where Tanh is the activation function. Conv+BN+ReLU fits layers 13-15 of ADNet with a filter size of $64 \times 3 \times 3 \times 64$. Conv is used for layer 16 in ADNet. Finally, the input noisy image is fused with the output of the 16th layer to enhance the representation ability of the denoising model.

AB uses a convolution of size 1×1 from the 17th layer to compress the obtained features into vectors as weights for the previous stage, which can also improve the efficiency of denoising. Next, AB uses the obtained weights to multiply the output of the 16th layer to extract more significant noise features. These two steps can be expressed in formulas (6) and (7), where Q_t is the output of the convolution from the 17th layer in ADNet.

$$Q_t = C(O_{FEB}) \tag{6}$$

$$I_R = Q_t \times O_{FEB} \tag{7}$$

The attention mechanism can extract hidden features in complex backgrounds, which is beneficial for blind denoising. RB predicts the residual image. The noisy image and the predicted residual image are differenced to obtain a clean image. We used ADNet to train four denoising models. Fig. 3 shows an example before and after denoising, and more denoising results can be seen in Section III-E. To further prove that our denoising module is effective for a wide range of noise, we trained on Poisson noise, uniform noise, blind noise with the same way as other noises. We then tested them on the forensic algorithm, where the blind noise contains unknown noises of different intensities. The generalization performance of the denoising module is demonstrated in Section III-G.

D. FORENSIC MODULE

The purpose of the forensic module is to detect evidence of potential local tampering within the image. To calculate the similarity between image patches, we adopt the Siamese and similarity networks introduced in [25]. From

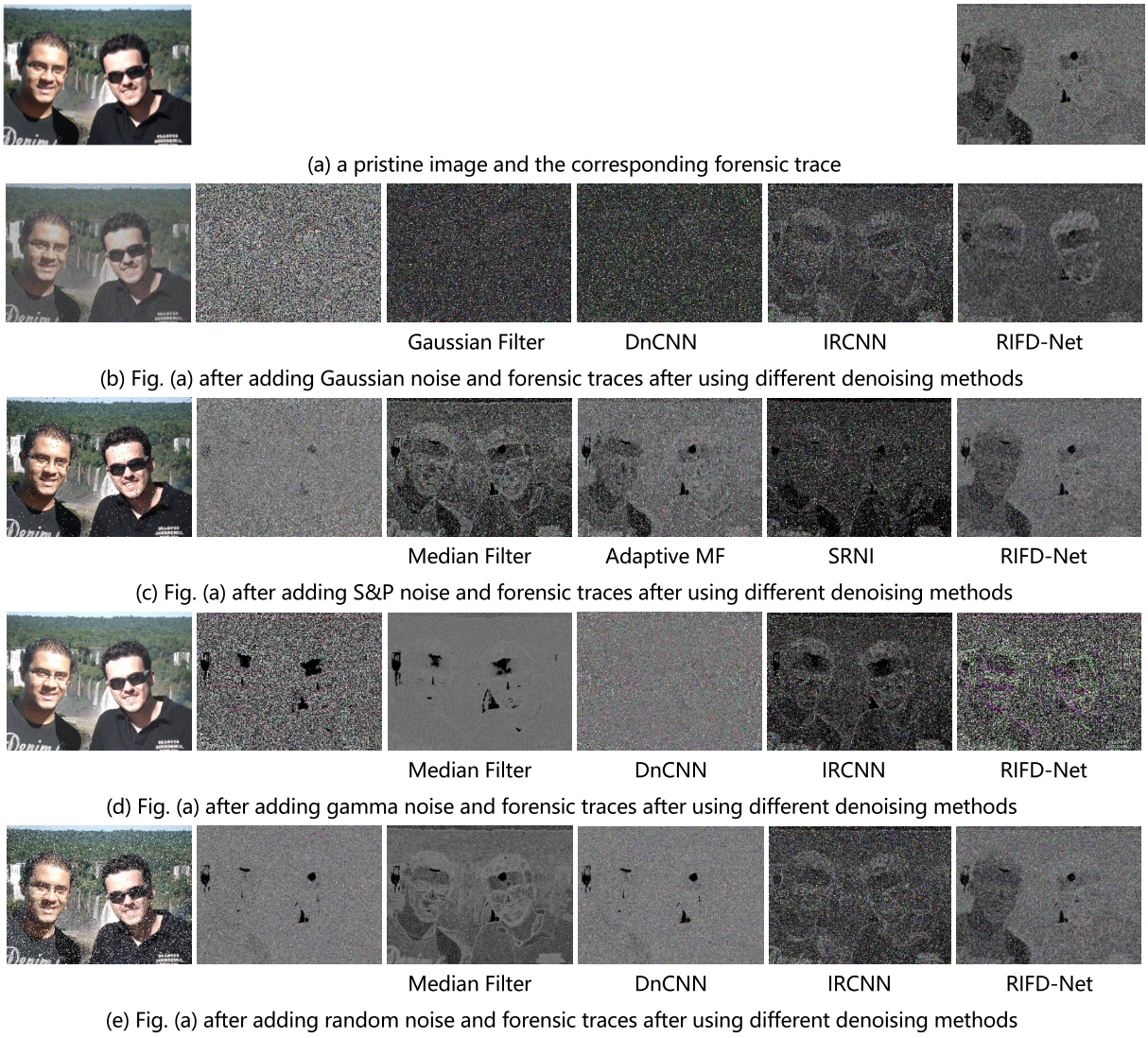


FIGURE 6. **Forensic traces of Fig. a (adding different noises) before and after denoising.**

an image, we select two small patches X_1 and X_2. These patches are then converted into forensic feature vectors $f(X_1)$ and $f(X_2)$ by the feature extractor. The feature vectors are individually mapped to formula (8), which represents a new N-dimensional feature space. The new feature space is capable of recording high-level image information of the pair of patches, X.

$$f : X \to R^N \tag{8}$$

$$S : R^N \times R^N \to [0, 1] \tag{9}$$

The Similarity network, with two fully connected layers and activation functions, compares features of X_1 and X_2. Formula (9) yields a similarity score between 0 and 1, indicating dissimilarity or high similarity in forensic traces. The confusion matrix is generated by computing the patch similarities. Our tampering detection strategy involves calculating the average similarity score of the image patches and establishing a threshold to discern both average and high similarity as indicators of tampering.

Fig. 4 displays the similarity confusion matrices for tampered images in three scenarios. In Fig. 4(e), the similarity between the spliced region and other regions reaches 1, indicating that the method of calculating forensic similarity is ineffective when noise is present, which highlights the importance of high-quality images in the forensic identification process. Observably, RIFD-Net effectively mitigates noise interference on forensic traces, as evidenced by the similarity between Fig. 4(d) and Fig. 4(f).

III. EXPERIMENTS

A. DATASETS

For the noise prediction module, we selected 1000 color images from the Berkeley Segmentation Dataset (BSD) [42] and LIVE1, which are popular sources for image classification. To diversify the data and capture fine noise features,

TABLE 2. Configuration of the training datasets.

	DataSets(Quantity)	Description
Noise prediction module	BSD、LIVE1(1000) Dresden(1000)	2000. per noise model
Denoising module	BSD、LIVE1(1000)	1000. per denoising model
Forensic module	Columbia(363)	180 spliced TIF images
	DSO-1(200)	100 spliced PNG images
	Korus(440)	220 spliced TIF images

we also selected 1000 images from the Dresden Image Database [43]. Furthermore, the transformer's random crop function was utilized to generate five unique images cut from each original image. Ultimately, a training set of 7000 images was used for the noise prediction module. For the denoising module, the same dataset of 1000 color images from BSD and LIVE1 was used as that in the noise prediction module. Specifically, 1000 pictures were included in each of the four noise datasets, with a total of 4000 pictures used for denoising training. The same test set of 1000 images was used for both modules. Table 2 displays the configuration of datasets in the three modules. Fig. 7 displays an image from the BSD dataset and the local details upon the addition of different types of noise. S&P noise and random noise pollution can affect pixel values, whereas Gaussian noise and gamma noise are two types of noise with probability densities following Gaussian and gamma distributions, respectively.

In the forensic module, we utilized Mayer and Stamm's pre-trained model, acquired through the training of tens of thousands of images captured from 95 different cameras. The model carries the parameters and weights of various camera models. The performance of RIFD-Net's forgery detection was then assessed using three publicly accessible datasets. These datasets consisted of the following: 1) the ''Columbia Uncompressed Splicing Database'' [44], which contained 180 spliced TIF images; 2) the ''Carvalho DSO-1 Database'' [45], which comprised 100 spliced PNG images; and 3) the ''Korus Database'' [46], which included 220 spliced TIF images. Fig. 8 shows examples of spliced images from each dataset.

During forgery detection, different images in a dataset may have varying noise types or the absence of noise. Therefore, we created confusion noise datasets based on the original datasets (Table 3 shows the configurations).

B. TRAINING DETAILS

The noise prediction module is utilized to classify noise in images, particularly in color images, where the input contains 3 image channels. The learning rate is set at 0.01 and is reduced exponentially. We employed CrossEntropyLoss as the loss function and Adam optimizer, which can accommodate large-scale data and parameters. The number of iterations and batch size is 100 and 8, respectively. In the denoising module, the depth of the convolutional layers is 17, and the initial parameters are the learning rate of 1e-3, epsilon of 1e-8, beta1 of 0.9, and beta2 of 0.99, which are the BN parameters. The number of iterations and batch size is 10 and 24, respectively.

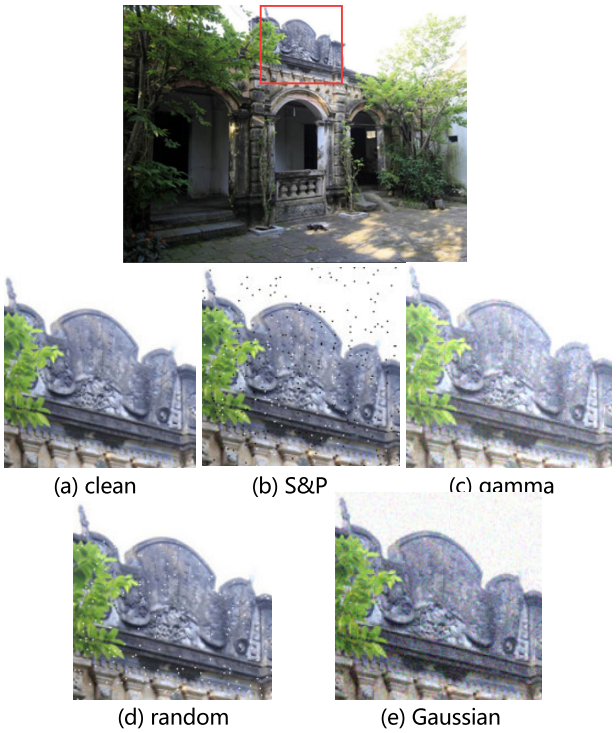


FIGURE 7. An example from the BSD and local details after adding different noises.

We applied Pytorch 1.2.0 and Python 3.6 to train and test the RIFD-Net. Specifically, all experiments in this section were performed on Centos 7.6 server, which contains an UniServer R5300 G3 6248R CPU, 12∗16G RAMs, and 4∗32G NVIDIA Tesla V100S GPUs. Finally, we used CUDA10's CUDNN to accelerate the calculation speed of the GPU.

C. PERFORMANCE OF THE FORENSIC METHODS ON BENCHMARK DATASETS

RIFD-Net has shown high accuracy in detecting image forgery in both clean and noisy environments. In this section, we compare the forensic performance of RIFD-Net with other splicing detection methods in both clean and noisy environments, as presented in Table 4. The evaluation metric used to determine whether an image has been tampered is mAP [22], with higher mAP values indicating better forgery detection performance.

Table 4 verifies that RIFD-Net outperforms other forgery detection methods, both in clean and noisy environments. In a noisy environment, the mAP of RIFD-Net drops within 10%, while the performances of most other methods are greatly weakened. RRU-Net performs well on the Columbia dataset

FIGURE 8. Examples of forgery in benchmark datasets. The top row contains spliced images, and the bottom row displays the corresponding ground truth images.

TABLE 3. Configuration of confusion noise datasets.

Datasets (confusion)	Number of images					
	clean	S&P	Gaussian	gamma	random	total
Confusion _DSO-1	17	35	12	20	15	100
Confusion _Columbia	44	51	25	20	40	180
Confusion_ Korus	52	70	33	30	35	220

because RRU-Net's training set consists of simple and small splicing datasets (including Columbia), and thus RRU-Net performs best on simple spliced images with small resolution. Compared with ''Forensic Similarity'' method, RIFD-Net improves the mAP of forgery detection by more than 20% on three benchmark datasets.

D. PERFORMANCE OF THE NOISE PREDICTION MODULE

This section shows the excellent performance of noise prediction. We applied different types of noise to the benchmark datasets to produce the confusion noise datasets (configuration in Table 3) and evaluated the accuracy of the prediction. Table 5 displays the prediction of our RIFD-Net for noise types, with almost 100% accuracy in clean images, S&P noise, and random noise. There is a slight decline in the discrimination accuracy between gamma noise and Gaussian noise. Nonetheless, the prediction results are impressively accurate, providing a viable foundation for the subsequent denoising module.

E. PERFORMANCE OF THE DENOISING MODULE

The following section provides quantitative and qualitative analysis of the denoising module's performance, comparing it to other mainstream denoising algorithms.

Our quantitative analysis assessed image quality using PSNR (Peak Signal-to-Noise Ratio), where higher values indicate cleaner images. PSNR measures the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the quality of its representation, providing a numerical scale to evaluate the fidelity of images. Fig. 9 illustrates an example of an image before and after denoising.

To quantitatively evaluate our denoising performance, we compared our method with other denoising methods using four DSO-1 datasets with various types of added noise, as detailed in Table 6. Our denoising module achieved average PSNR values of up to 53.0 for S&P noise and random noise. For gamma noise, our method also achieved higher PSNR values. For Gaussian noise, our method is slightly weaker than IRCNN, but within acceptable limits. IRCNN is specialized in removing Gaussian noise and does not perform well in removing other noises, whereas our denoising method is dedicated to removing many different noises. Overall, our proposed method outperforms other denoising methods for different types of noise.

F. PERFORMANCE OF FORGERY DETECTION IN DIFFERENT NOISY ENVIRONMENTS

Table 7 showcases the forensic module's performance across various noisy environments, both before and after the application of our noise prediction and denoising modules. Each dataset is subjected to a single type of noise, denoted by its intensity in brackets. The mAP values for forgery detection accuracy are presented, with the first column depicting the results without using our modules and the second column showing the results post-application. In addition, three other confusion noise datasets were detected.

To assess the effectiveness of RIFD-Net under diverse noise intensities, we retrained denoising models using a training set consisting of images with varying noise intensities

TABLE 4. mAP, performance of forgery detection on benchmark datasets.

	clean			noisy		
	DSO-1	Columbia	Korus	Confusion_ DSO-1	Confusion_ Columbia	Confusion_ Korus
Forensic Similarity [25]	0.96	0.92	0.63	0.65	0.67	0.35
EXIF [22]	0.74	0.95	0.54	0.55	0.80	0.44
E2E-RGB [27]	0.71	0.80	0.56	0.58	0.64	0.55
RRU-Net [47]	0.72	**0.96**	0.62	0.70	**0.90**	0.58
RIFD-Net	**0.96**	0.92	**0.63**	**0.86**	0.85	**0.62**

TABLE 5. Noise prediction results.

Dataset	clean	S&P	gamma	Gaussian	random
	Number (correct predictions)/ Number (total)				
Confusion_DSO-1	17/17	35/35	11/12	19/20	15/15
Confusion_Columbia	44/44	51/51	24/25	19/20	40/40
Confusion_Korus	52/52	70/70	33/33	28/30	35/35

TABLE 6. Average PSNR values (dB) for different denoising methods under four noise conditions on the DSO-1 dataset.

S&P noise		Gaussian noise		gamma noise		random noise	
Noisy	21.6	Noisy	25.0	Noisy	20.0	Noisy	23.6
Median filter	34.5	***Gaussian filter***	31.8	***Median filter***	20.4	***Median filter***	38.7
Adpative median filter [48]	40.7	***DnCNN*** [31]	30.0	***DnCNN***	29.7	***DnCNN***	25.1
SRNI [49]	26.6	***IRCNN*** [50]	**34.8**	***IRCNN***	20.4	***IRCNN***	25.5
Proposed	**53.1**	***Proposed***	34.0	***Proposed***	**34.1**	***Proposed***	**53.0**

pristine image | S&P noise PSNR = 22.1 dB | Gaussian noise PSNR = 24.7 dB | gamma noise PSNR = 19.8 dB | random noise PSNR = 24.6 dB

(a) An example before and after adding four types of noise.

PSNR = 53.5 dB | PSNR = 35.1 dB | PSNR = 39.2 dB | PSNR = 53.7 dB

(b) Results after applying the denoising module to Fig. (a).

FIGURE 9. An evaluation of denoising. The proposed method effectively enhances the image quality with a notably high PSNR.

(1000 images per intensity). Table 8, different from Table 7, illustrates the selection of denoising models for specific noise intensities versus retrained models covering a range of intensities. The noise intensity in Table 7's test dataset is fixed, while Table 8's test dataset spans the noise range of the retrained denoising model.

TABLE 7. mAP values across various noise types and specific intensity environments.

	clean	S&P (0.01)		Gaussian (15)		gamma (10)		random (0.01)		Poisson (0.03)		uniform (10)		blind		confusion	
dataset			our		our		our		our		our		our		our		our
DSO-1	**0.96**	0.56	**0.96**	0.63	**0.90**	0.61	**0.65**	0.66	**0.97**	0.65	**0.78**	0.68	**0.80**	0.64	**0.75**	0.65	**0.86**
Columbia	**0.92**	0.62	**0.92**	0.62	**0.73**	0.61	**0.68**	0.62	**0.94**	0.64	**0.73**	0.66	**0.79**	0.70	**0.78**	0.67	**0.85**
Korus	**0.63**	0.50	**0.65**	0.50	**0.69**	0.50	**0.79**	0.55	**0.65**	0.60	**0.74**	0.62	**0.83**	0.62	**0.69**	0.39	**0.62**

TABLE 8. mAP values across various noise types and diverse intensity environments.

	clean	S&P (0.01~0.05)		Gaussian (10~20)		gamma (5~15)		random (0.01~0.05)		Poisson (0.02~0.05)		uniform (5~15)		confusion	
dataset			our		our		our		our		our		our		our
DSO-1	**0.96**	0.43	**0.86**	0.58	**0.80**	0.53	**0.59**	0.52	**0.87**	0.58	**0.75**	0.65	**0.75**	0.52	**0.74**
Columbia	**0.92**	0.51	**0.80**	0.52	**0.67**	0.52	**0.62**	0.54	**0.83**	0.60	**0.71**	0.59	**0.76**	0.50	**0.72**
Korus	**0.63**	0.45	**0.56**	0.46	**0.62**	0.45	**0.72**	0.44	**0.58**	0.54	**0.69**	0.60	**0.79**	0.37	**0.61**

TABLE 9. Performance of other forensic methods under different conditions.

	clean			confusion noise			Denoising with RIFD-Net		
	DSO-1	Columbia	Korus	DSO-1	Columbia	Korus	DSO-1	Columbia	Korus
NoisePrint (MCC) [24]	0.79	0.78	0.34	0.39	0.52	0.23	0.56	0.66	0.29
EXIF(AP) [22]	0.74	0.95	0.54	0.55	0.80	0.44	0.75	0.94	0.59
E2E-RGB (AUC) [27]	0.63	0.78	0.50	0.50	0.60	0.44	0.59	0.65	0.48
RRU-Net (mAP) [47]	0.72	0.96	0.62	0.70	0.90	0.58	0.72	0.95	0.62

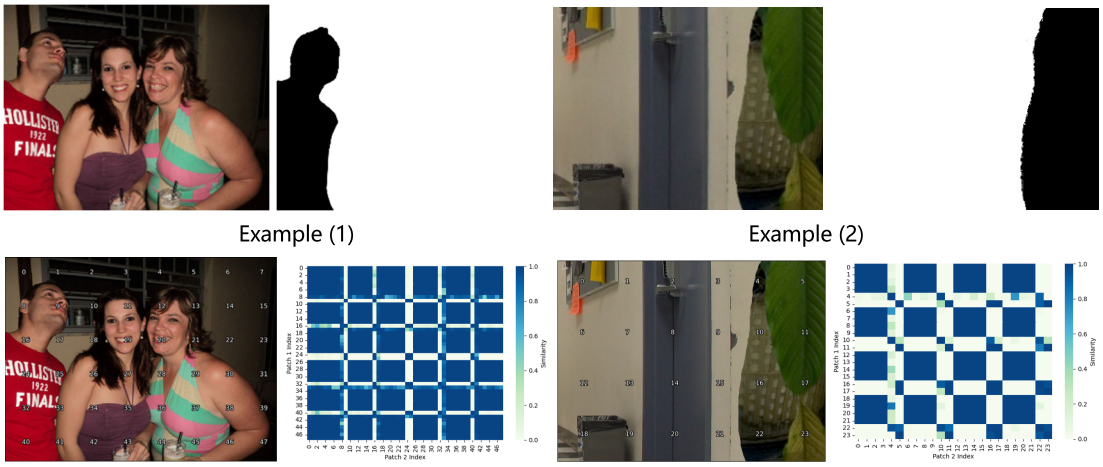
Tables 7 and 8 highlight the significant performance improvement of the proposed method in the presence of noise, particularly excelling in handling S&P and random noise to achieve performance levels comparable to noiseless conditions. However, RIFD-Net exhibits suboptimal results in scenarios with other types of noise. This could be attributed to the inherent challenge of balancing denoising effectiveness with the potential loss of image detail, especially with linear noise. In addition, the suboptimal results may be associated with varying image resolutions, as demonstrated by the notably high mAP values (0.9) on the DSO-1 dataset under Gaussian noise conditions, where images have dimensions of 2048 × 1536 pixels. In contrast, the lower resolution of Columbia images results in substantial degradation during the denoising process. Overall, the experimental results indicate that RIFD-Net enhances the mAP of forgery detection by almost 20% in noisy environments compared to scenarios without the preceding two modules. The denoising module demonstrates exceptional generalization performance, effectively removing various types of noise, including blind noise, while striving to retain forensic traces during the denoising process.

G. GENERALIZATION PERFORMANCE OF THE DENOISING MODULE

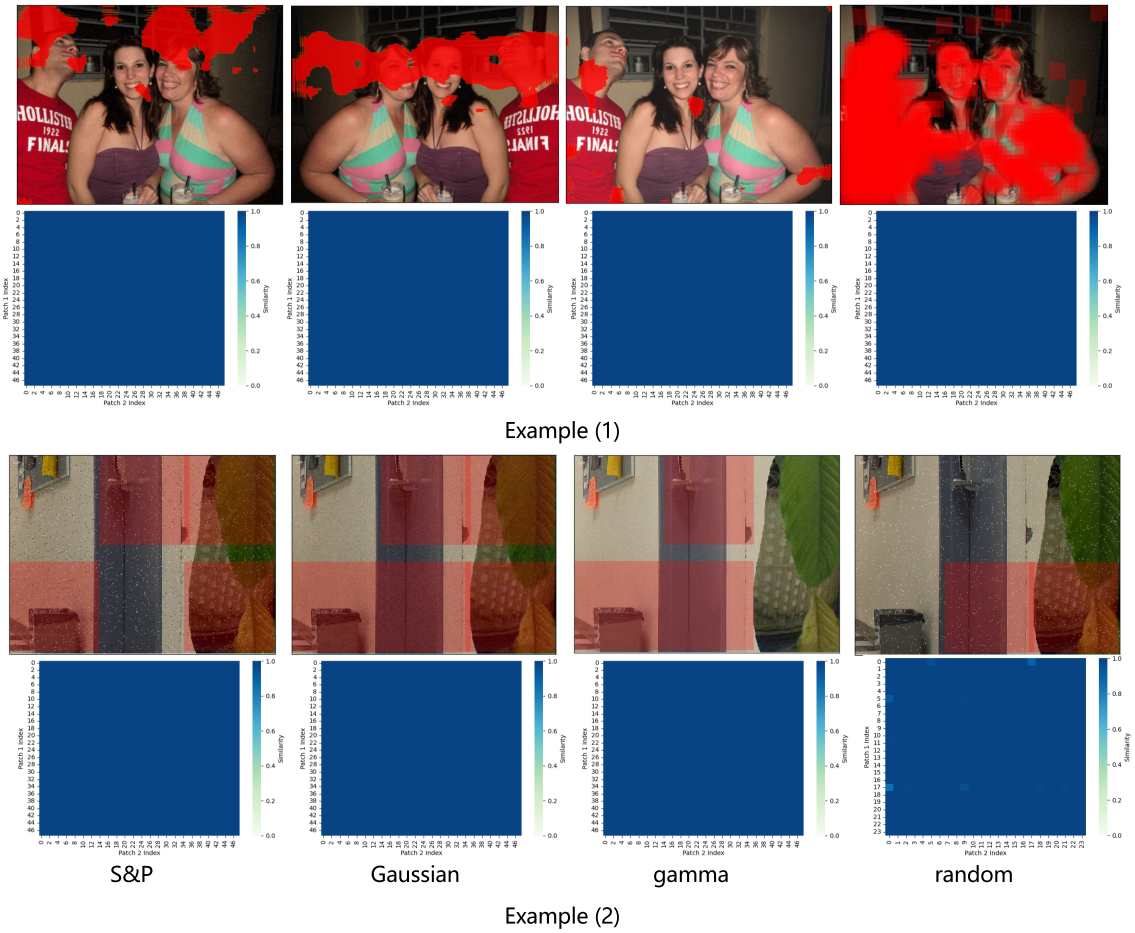
To assess the effectiveness of denoising module in RIFD-Net, we integrated it into other forensic methods and evaluated their performance in three case studies—utilizing clean datasets, datasets with confusion noise, and datasets denoised using RIFD-Net's denoising module. Table 9 summarizes the outcomes, indicating that our approach consistently yields positive results when applied to various forensic methods, even in the presence of noise. This resilience can be attributed to RIFD-Net's capability to eliminate noise while preserving essential forensic traces as much as possible.

H. FORGERY LOCALIZATION

This section presents forgery localization results using a chunk-based approach. The image is segmented based on varying resolutions, and each image patch is numbered. We then computed the similarity matrix, identified image patch indices corresponding to white, and applied them to highlight regions of possible tampering. Different types of noise may affect the localization accuracy of tampered regions because of interference with the similarity matrix calculation. In Fig. 10, the forgery localization results for examples (1) and (2) in a noisy environment are compared to those of "Forensic Similarity" algorithm. Fig. 10(a) shows similarity matrices for clean examples, whereas Fig. 10(b) illustrates forgery localization results and similarity matrices for the "Forensic Similarity" algorithm after adding four different noises. Fig. 10(c) displays the forgery localization results and similarity matrices of our RIFD-Net in four noisy

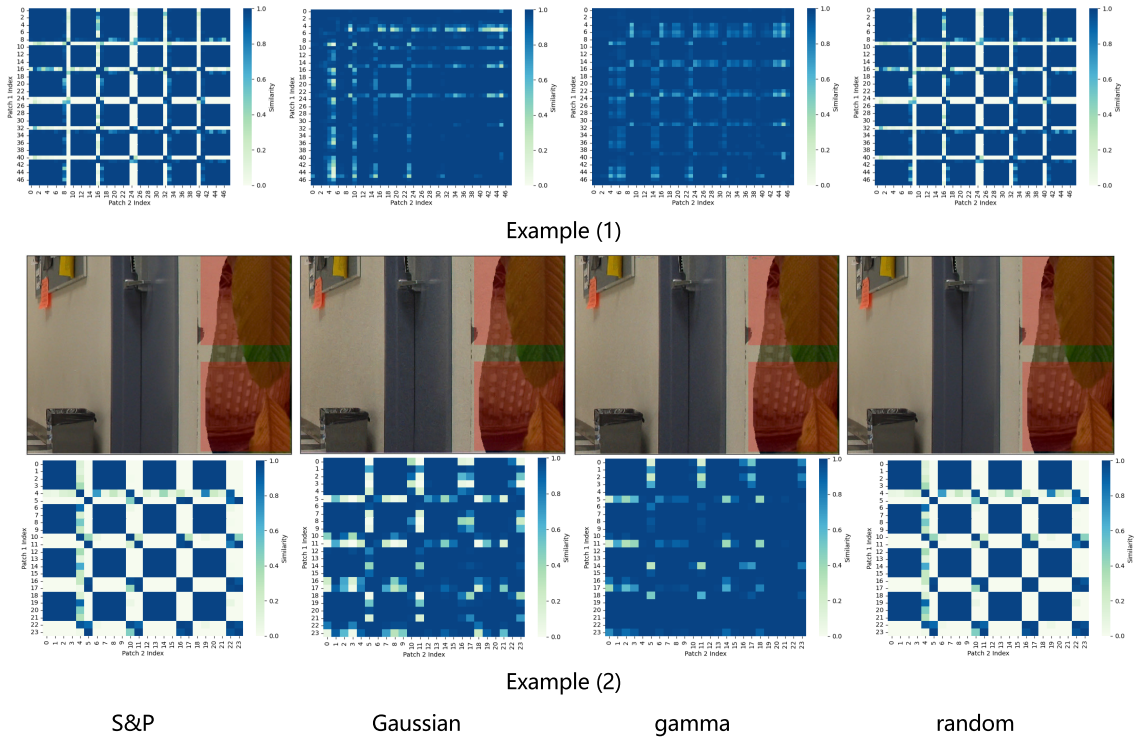


(a) Tampered Image (clean) and the corresponding ground truth (first line), similarity matrices (second line).



(b) localization results (first line) and the corresponding similarity matrices of Forensic Similarity [25] (second line).

FIGURE 10. **Two examples of localization in four noisy environments.**



(c) localization results (first line) and the corresponding similarity matrices of RIFD-Net (second line).

FIGURE 10. ***(Continued.)*** **Two examples of localization in four noisy environments.**

environments. Despite the serious impact of noise on forensic traces, RIFD-Net preserves these traces to a greater extent during denoising, resulting in improved forgery localization, particularly for S&P noise and random noise.

IV. CONCLUSION

In this paper, we introduce RIFD-Net, a novel method for detecting image splicing. Our approach is robust, particularly in challenging scenarios with image noise. RIFD-Net incorporates multiple classifiers and a denoising network to effectively eliminate various forms of image noise, thereby improving the mAP of forgery detection. The forensic module, employing a Siamese network, calculates the similarity between image patches. The denoising capabilities of RIFD-Net play a crucial role in noise elimination, contributing significantly to enhanced the robustness of forensics. In addition, we evaluate and validate the effectiveness of our denoising model in conjunction with other forensic methods.

V. DISCUSSION

In this section, we present a comprehensive discussion of the aforementioned experimental results and limitations, and propose future research directions. The experiments indicate that the current mainstream methods for detecting digital image forgeries perform well only on forged images with high image quality. Once the images are contaminated with noise, the forensic performance significantly deteriorates. RIFD-Net exhibits robust forgery detection performance, whether applied to datasets with high image quality or to datasets subjected to various noise contaminations.

From both subjective and objective evaluations, our proposed RIFD-Net demonstrates favorable outcomes. Subjectively, for a forged image containing noise, the denoising module significantly reduces the noise, resulting in a noticeable improvement compared with the image before denoising. In the forensic module, the difference between the forensic similarity matrices before and after using RIFD-Net is substantial. Virtually no forensic traces were discernible before using RIFD-Net, whereas after application, partial restoration of forensic traces was achieved. The results of forgery localization further validate these observations. Objectively, both the noise prediction module and the denoising module achieve commendable performance according to their respective evaluation metrics. The forensic module exhibits a substantial improvement in the mAP values before and after using RIFD-Net. In addition, RIFD-Net demonstrates high robustness in differentiating various types and intensities of noise environments.

However, RIFD-Net also presents specific limitations and areas that require improvement. Notably, its efficacy in mitigating Gaussian noise and gamma noise is less conspicuous compared with several other types of noise. This deficiency is particularly apparent in the preservation of forensic traces. In instances of high intensity for both Gaussian and gamma noise, despite the denoising module effectively eliminating the noise, forensic traces undergo substantial obliteration.

This is likely because both Gaussian noise and gamma noise adhere to normal and gamma distributions, respectively, exhibiting a certain level of continuity. Throughout the training process, the network endeavors to capture the statistical characteristics of the noise, inadvertently smoothing image details during denoising. This smoothing effect leads to the loss of forensic traces.

This paper highlights the effectiveness of the sparse mechanism, which combines dilated and normal convolution, in achieving a balance between denoising and preserving low-level features. However, the balance of this combination is relative and is only optimal within a limited range of noise intensity. Furthermore, the current strategy of training distinct denoising models for different noise types increases algorithm processing time and complexity. In future work, our objective is to enhance the algorithm by consolidating it into a single denoising model capable of handling various types of noise. This modification severs the dependence on the noise prediction module, resulting in a significant improvement in algorithm efficiency.

ACKNOWLEDGMENT

The authors would like to thank for the detailed and kind comments of the hidden reviewers and the editor whose administrative processing.

REFERENCES

[1] K. H. Rhee, "Detection of spliced image forensics using texture analysis of median filter residual," *IEEE Access*, vol. 8, pp. 103374–103384, 2020, doi: 10.1109/ACCESS.2020.2999308.

[2] W. Wang, J. Dong, and T. Tan, "A survey of passive image tampering detection," in *Proc. Int. Workshop Digit. Watermarking*, 2009, pp. 308–322, doi: 10.1007/978-3-642-03688-0_27.

[3] J. Dong, W. Wang, and T. Tan, "Run-length and edge statistics based approach for image splicing detection," in *Proc. Int. Workshop Digit. Watermarking*, 2008, pp. 76–87, doi: 10.1007/978-3-642-04438-0_7.

[4] W. Wang, J. Dong, and T. Tan, "Effective image splicing detection based on image chroma," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 1257–1260, doi: 10.1109/ICIP.2009.5413549.

[5] J. He, Z. Lin, and L. Wang, "Detecting doctored JPEG images via DCT coefficient analysis," in *Proc. 9th Eur. Conf. Comput. Vis.*, 2006, pp. 423–435, doi: 10.1007/11744078_33.

[6] S. Ye, Q. Sun, and E.-C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *Proc. IEEE Multimedia Expo Int. Conf.*, Jul. 2007, pp. 12–15, doi: 10.1109/ICME.2007.4284574.

[7] W. Luo, Z. Qu, J. Huang, and G. Qiu, "A novel method for detecting cropped and recompressed image block," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2007, pp. 217–220 doi: 10.1109/ICASSP.2007.366211.

[8] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 154–160, Mar. 2009, doi: 10.1109/TIFS.2008.2012215.

[9] S. Wang and X. Niu, "Hiding traces of double compression in JPEG images based on Tabu search," *Neural Comput. Appl.*, vol. 22, no. S1, pp. 283–291, May 2013, doi: 10.1007/s00521-012-0841-5.

[10] M. K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *Proc. 7th Workshop Multimedia Secur.*, Aug. 2005, pp. 1–10, doi: 10.1145/1073170.1073171.

[11] A. C. Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3948–3959, Oct. 2005, doi: 10.1109/tsp.2005.855406.

[12] W. Shan, Y. Yi, R. Huang, and Y. Xie, "Robust contrast enhancement forensics based on convolutional neural networks," *Signal Process., Image Commun.*, vol. 71, pp. 138–146, Feb. 2019, doi: 10.1016/j.image.2018.11.011.

[13] S. Khan and T. Bianchi, "Fast image clustering based on compressed camera fingerprints," *Signal Process., Image Commun.*, vol. 91, Feb. 2021, Art. no. 116070, doi: 10.1016/j.image.2020.116070.

[14] S.-E. Abdosalehi and A. Mahmoodi-Aznaveh, "Splicing localization in tampered blurred images," in *Proc. 4th Int. Conf. Pattern Recognit. Image Anal. (IPRIA)*, Tehran, Iran, Mar. 2019, pp. 46–51, doi: 10.1109/PRIA.2019.8785965.

[15] D. Das, R. Naskar, and R. S. Chakraborty, "Image splicing detection with principal component analysis generated low-dimensional homogeneous feature set based on local binary pattern and support vector machine," *Multimedia Tools Appl.*, vol. 82, no. 17, pp. 25847–25864, Jul. 2023, doi: 10.1007/s11042-023-14658-w.

[16] K. Kadam, S. Ahirrao, K. Kotecha, and S. Sahu, "Detection and localization of multiple image splicing using MobileNet v1," *IEEE Access*, vol. 9, pp. 162499–162519, 2021, doi: 10.1109/ACCESS.2021.3130342.

[17] J. Wu, X. Chang, T. Yang, and K. Feng, "Blind forensic method based on convolutional neural networks for image splicing detection," in *Proc. IEEE 5th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2019, pp. 2014–2018, doi: 10.1109/ICCC47050.2019.9064258.

[18] L. Bondi, S. Lameri, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "Tampering detection and localization through clustering of camera-based CNN features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1855–1864, doi: 10.1109/CVPRW.2017.232.

[19] H. Ding, L. Chen, Q. Tao, Z. Fu, L. Dong, and X. Cui, "DCU-Net: A dual-channel U-shaped network for image splicing forgery detection," *Neural Comput. Appl.*, vol. 35, no. 7, pp. 5015–5031, Mar. 2023.

[20] C. You, H. Zheng, Z. Guo, T. Wang, and X. Wu, "Tampering detection and localization base on sample guidance and individual camera device convolutional neural network features," *Expert Syst.*, vol. 40, no. 1, Jan. 2023, Art. no. e13102, doi: 10.1111/exsy.13102.

[21] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional Siamese networks for change detection," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 4063–4067.

[22] M. Huh, A. Liu, and A. Owens, "Fighting fake news: Image splice detection via learned self-consistency," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2019, pp. 101–117.

[23] B.-C. Chen, P. Ghosh, V. I. Morariu, and L. S. Davis, "Detection of metadata tampering through discrepancy between image content and metadata using multi-task deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1872–1880, doi: 10.1109/CVPRW.2017.234.

[24] D. Cozzolino and L. Verdoliva, "Noiseprint: A CNN-based camera model fingerprint," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 144–159, 2019.

[25] O. Mayer and M. C. Stamm, "Forensic similarity for digital images," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1331–1346, 2020, doi: 10.1109/tifs.2019.2924552.

[26] O. Mayer and M. C. Stamm, "Exposing fake images with forensic similarity graphs," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 5, pp. 1049–1064, Aug. 2020.

[27] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi, "A full-image full-resolution end-to-end-trainable CNN framework for image forgery detection," *IEEE Access*, vol. 8, pp. 133488–133502, 2020.

[28] V. Rohit and J. Ali, "A comparative study of various types of image noise and efficient noise removal techniques," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 10, pp. 1–6, 2013.

[29] C. Boncelet, "Image noise models," in *The Essential Guide to Image Processing*, A. Bovik, Ed. New York, NY, USA: Academic, 2009, pp. 143–167, doi: 10.1016/B978-0-12-374457-9.00007-X.

[30] J. Coady, A. O'Riordan, G. Dooly, T. Newe, and D. Toal, "An overview of popular digital image processing filtering operations," in *Proc. 13th Int. Conf. Sens. Technol. (ICST)*, Dec. 2019, pp. 1–5, doi: 10.1109/ICST46873.2019.9047683.

[31] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[32] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[33] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1712–1722.

[34] C. Szegedy, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[35] S. Chaudhari, V. Mithal, G. Polatkan, and R. Ramanath, "An attentive survey of attention models," *ACM Trans. Intell. Syst. Technol.*, vol. 12, no. 5, pp. 1–32, Oct. 2021.
[36] C. Tian, Y. Xu, Z. Li, W. Zuo, L. Fei, and H. Liu, "Attention-guided CNN for image denoising," *Neural Netw.*, vol. 124, pp. 117–129, Apr. 2020, doi: 10.1016/j.neunet.2019.12.024.
[37] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1053–1061, doi: 10.1109/CVPR.2018.00116.
[38] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2691–2706, Nov. 2018, doi: 10.1109/TIFS.2018.2825953.
[39] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.
[40] X. Zhang, W. Yang, Y. Hu, and J. Liu, "DMCNN: Dual-domain multi-scale convolutional neural network for compression artifacts removal," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 390–394 doi: 10.1109/ICIP.2018.8451694.
[41] Y. Xu, Z. Zhang, G. Lu, and J. Yang, "Approximately symmetrical face images for image preprocessing in face recognition and sparse representation based classification," *Pattern Recognit.*, vol. 54, pp. 68–82, Jun. 2016, doi: 10.1016/j.patcog.2015.12.017.
[42] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, Jul. 2001, pp. 416–423 doi: 10.1109/ICCV.2001.937655.
[43] T. Gloe and R. Böhme, "The'Dresden image database'for benchmarking digital image forensics," in *Proc. ACM Symp. Appl. Comput.*, 2010, pp. 1584–1590 doi: 10.1145/1774088.1774427.
[44] Y.-F. Hsu and S.-F. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2006, pp. 549–552, doi: 10.1109/ICME.2006.262447.
[45] T. J. de Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. de Rezende Rocha, "Exposing digital image forgeries by illumination color classification," *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 7, pp. 1182–1194, Jul. 2013, doi: 10.1109/TIFS.2013.2265677.
[46] P. Korus and J. Huang, "Multi-scale analysis strategies in PRNU-based tampering localization," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 809–824, Apr. 2017, doi: 10.1109/TIFS.2016.2636089.
[47] X. Bi, Y. Wei, B. Xiao, and W. Li, "RRU-Net: The ringed residual U-Net for image splicing forgery detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 30–39, doi: 10.1109/CVPRW.2019.00010.
[48] Z. Zhang, D. Han, J. Dezert, and Y. Yang, "A new adaptive switching median filter for impulse noise reduction with pre-detection based on evidential reasoning," *Signal Process.*, vol. 147, pp. 173–189, Jun. 2018, doi: 10.1016/j.sigpro.2018.01.027.
[49] A. Villar-Corrales, F. Schirrmacher, and C. Riess, "Deep learning architectural designs for super-resolution of noisy images," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 1635–1639 doi: 10.1109/ICASSP39728.2021.9414733.
[50] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3929–3938 doi: 10.1109/CVPR.2017.300.

WUYANG SHAN (Member, IEEE) received the Ph.D. degree from Wuhan University, in 2019. He is currently with the College of Computer Science and Cyber Security, Chengdu University of Technology, Chengdu, China. His current research interests include multimedia forensics, data hiding, and computer vision.

DENG ZOU (Member, IEEE) was born in 1999. He is currently pursuing the M.S. degree with the College of Computer Science and Cyber Security, Chengdu University of Technology, Chengdu, China. His research interests include image forensics and image processing.

PENGBO WANG was born in 1998. He received the master's degree from the School of Computer and Network Security, Chengdu University of Technology, Chengdu, China. His research interests include image forensics and deep learning.

JINGCHUAN YUE was born in 2000. He is currently pursuing the master's degree with the College of Computer Science and Cyber Security, Chengdu University of Technology, Chengdu, China. His research interests include image forensics and image processing.

AOLING LIU was born in 1999. She is currently pursuing the master's degree in computer and network security from the Oxford Brookes College, Chengdu University of Technology. Her research interests include image forensics and image processing.

JUN LI (Member, IEEE) received the Ph.D. degree from the Chengdu University of Technology, in 2011. He is currently a Professor with the College of Computer Science and Cyber Security, Chengdu University of Technology. His current research interests include HPC and computer vision.