

## Research Article

# Image Source Identification Using Convolutional Neural Networks in IoT Environment

Yan Wang,<sup>1</sup> Qindong Sun ,<sup>1,2</sup> Dongzhu Rong,<sup>1</sup> Shancang Li,<sup>3</sup> and Li Da Xu<sup>4</sup>

<sup>1</sup>Shaanxi Key Laboratory of Network Computing and Security, Xi'an University of Technology, 710048, China

<sup>2</sup>School of Cyber Science and Engineering, Xi'an Jiaotong University, 710049, China

<sup>3</sup>Department of Computer Science, UWE Bristol, BS16 1QY, UK

<sup>4</sup>Department of IT and DS, Old Dominion University, Norfolk, VA, USA

Correspondence should be addressed to Qindong Sun; [qdongsun@xjtu.edu.cn](mailto:qdongsun@xjtu.edu.cn)

Received 7 July 2021; Accepted 24 August 2021; Published 11 September 2021

Academic Editor: Zhihan Lv

Copyright © 2021 Yan Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Digital image forensics is a key branch of digital forensics that based on forensic analysis of image authenticity and image content. The advances in new techniques, such as smart devices, Internet of Things (IoT), artificial images, and social networks, make forensic image analysis play an increasing role in a wide range of criminal case investigation. This work focuses on image source identification by analysing both the fingerprints of digital devices and images in IoT environment. A new convolutional neural network (CNN) method is proposed to identify the source devices that token an image in social IoT environment. The experimental results show that the proposed method can effectively identify the source devices with high accuracy.

## 1. Introduction

The IoT is revolutionizing our everyday lives provisioning a wide range of novel applications leverage on ecosystems of smart and highly heterogeneous devices [1]. The use of the fifth-generation mobile network (5G) has brought wide coverage, large connection, and low delay network access services to the IoT. In the face of heterogeneous network access technology, mobile IoT data presents the characteristics of massive, heterogeneous, and dynamic. Machine learning enables computers to automatically learn and analyze big data and then to make decisions and predictions about events in the real world [2]. With the wide application of IoT devices, the security of data in massive IoT devices has attracted much attention. Especially in the research of digital forensics, multimedia information of IoT devices has important analytical significance.

In recent years, social network platforms, such as Twitter, Facebook, WeChat, Instagram, and Weibo, have been increasingly used in our daily events and are changing the way we are communicating [3]. Related reports pointed out that in 2020, online social network users have reached

3.8 billion [4], and these users can publish and obtain various information on social network platforms to achieve the purpose of mutual communication and exchange. However, the development of various image editing software also provides convenience for criminals to use social networks to spread forged information. As a transmission medium between users and social network platforms [5], smart phones play an important role in the behavior of users using social platforms to publish and share multimedia content [6]. On the other hand, criminals can use smart phones to post faked image information on social network platforms. Therefore, a combination of smart phones and social network platforms used for image source identification has certain research significance. The research can help law enforcement officers to collect more criminal evidence to ensure the security of social network platforms and social stability.

The accuracy of traditional camera source identification mainly relies on the compression strength of the image that needs to be suppressed before noise fingerprint extraction [7]. It is therefore only suitable for camera source identification scenes with high-quality image factors. The images

published on social network platforms are compressed, and the traditional camera source identification method has low accuracy. In this paper, a novel camera source identification model based on a convolutional neural network (CSI-CNN) is proposed to extract the image noise fingerprint and compare it with the preestimated device fingerprint. The matching degree is evaluated based on the similarity of the two fingerprints and then determines the source of the image.

In summary, the major contributions of the proposed work are fourfold:

- (1) A novel method that combines smart mobiles and social network platforms for image source identification is proposed
- (2) A new CNN is designed to extract the fingerprint characteristics of image noise on social networks and to match the device fingerprint to identify the camera source device of the image
- (3) A loss function is proposed based on deep learning method to effectively extract the noise fingerprint of the test image
- (4) A new dataset was constructed to test the user identification framework based on camera fingerprints

## 2. Related Work

As we all know, the information shared on social networks is often dominated by images. It is of great significance for multimedia forensics to trace the source of these images and identify the camera source by matching them with the camera they belong to. It provides an effective method for network evidence collection by law enforcement officers in the event of cybercrime. To fully understand the relationship between the social network platform images and the camera to which it belongs, a detailed overview of the existing image traceability technology is carried out. The existing widely used image traceability methods mainly include camera source identification based on photo response nonuniformity (PRNU) and camera source identification based on deep learning techniques.

### 2.1. Camera Source Identification Method Based on PRNU.

The PRNU is mainly based on the use of digital imaging equipment in the production process due to the imperfection of manufacturing of the CCD sensor array, resulting in the imaging equipment photosensitive elements of the photosensitive characteristics of small differences, e.g., the most widely used is the PRNU feature proposed by [8], in which Chen et al. highlighted that the camera noise pattern can be used as a unique fingerprint for source camera identification [9] and image forgery detection. In [10], Li focused on enhancing the characteristics of PRNU and constructing a series of corresponding functions to improve the individual recognition effect of PRNU equipment. Subsequently, others thought that the color interpolation step would have an impact on the recognition of PRNU, so an algorithm

for extracting PRNU only for noninterpolated pixels was proposed. [11] is committed to the transformation of PRNU features, using principal component analysis and hash mapping to reduce the dimension of PRNU, thereby improving the recognition rate of features. [12] based on PRNU's camera source identification method, by collecting images taken by different devices, using PRNU extraction algorithm to extract image fingerprints from these images, and then using methods such as average or maximum likelihood estimation to perform fingerprints on the device and then calculate the correlation between each device fingerprint and a given test image, to determine the camera object that took the given test image. [13] used wavelet filters to enhance camera's sensor pattern noise output, applied threshold formulas to remove scene details, and enhanced PRNU quality and pattern information content through enhancement methods to improve recognition accuracy. [14] proposed a new method of linear Gaussian filter kernel estimation based on PRNU noise. The core idea of the method is to treat PRNU noise as identifying fingerprints and to compare the noise residuals of clean images and query images respectively. The noise residuals extracted in JPEG are correlated, and the linear relationship between the two is obtained through mathematical derivation. This method has a certain effect on the source recognition of the image after JPEG compression.

### 2.2. Camera Source Identification Method Based on Deep Learning.

With the development of artificial intelligence technology and the increase of available image datasets, deep learning technology is gradually introduced into the field of image forensics. Also, deep learning technology can extract the best features from a large number of training datasets, avoiding the limitations of artificially designed features. Due to the rise of social networking sites such as Twitter, Facebook, WeChat, Instagram, and Weibo, researchers can easily obtain a large number of images with complete tags, use these images as research objects to extract image features, and then, use the larger-scale dataset to verify the effectiveness of the algorithm. For example, [15] applied convolutional neural network (CNN) to camera source identification for the first time, directly learning the characteristics of each camera from the acquired images for identification. [16] proposed a camera model recognition method based on CNN. The preprocessing layer is added to the CNN model, including a high-pass filter applied to the input image. CNN is used for feature extraction, and finally, the recognition score of each camera model is output to classify the image. [17] proposed a solution to identify small-size image source camera, through transformation learning to train three fusion residual networks for saturated images, smooth images, and other images, from the three residual networks (ResNet) [18] learning features in the residual block to more accurately recognize the input image. [19] proposed a method of learning twin neural networks, which uses a unique structure to rank the similarity between input contents. The predictive ability of the network is used not only for new data but also for new categories in unknown distribution. By applying it in image forensics, the accuracy and universality of picture recognition can be

improved. Also, [20] used the DnCNN [21] network models, extracted higher-quality image noise fingerprints, and performed correlation calculations based on the device fingerprints estimated by the maximum likelihood estimation to update the model parameters for better feature learning.

So far, due to the extensiveness and heterogeneity of data information on social network platforms and the difficulty of high computational complexity caused by large-scale datasets for camera source identification algorithms, it is of great significance to combine the traditional PRNU-based noise estimation with the deep learning-based noise estimation and apply it to camera source identification and network forensics.

Based on the investigation of the above-related work, this paper integrates PRNU and deep learning to design a camera source identification network (CSI-CNN) based on image noise fingerprint feature extraction, which optimizes the fully convolutional networks (FCN) [22]; network structure added the bottleneck residual block [18], combined with the idea of wavelet denoising for design. Based on the correlation between the preestimated PRNU device fingerprint and the social network image noise fingerprint extracted by CSI-CNN, a new loss function is designed to train, update network parameters, extract higher-quality image noise fingerprints, and obtain higher camera source identification accuracy.

### 3. The Proposed Method

The core idea of the social network image source identification method proposed in this paper is to identify the camera device source of the images posted by the user on the social network. That is, the noise fingerprint features can be extracted from the images on the social network through CSI-CNN designed in this paper, and the extracted noise fingerprint is correlated with the preestimated camera fingerprints; afterwards, the calculated correlation is used to determine whether the image on the social network is a real image taken by the camera held by the user. Camera fingerprint estimation and social network noise fingerprint extraction are the key contents of camera source identification, which will be introduced in detail in this section.

**3.1. Camera Fingerprint Extractions.** The social network image source identification method based on camera source recognition requires preestimation of the camera fingerprint, that is, the PRNU value. The specific process includes two parts: determining the camera sensor output model and PRNU estimation.

**3.1.1. Camera Sensor Output Model.** The imaging process of the camera is very complicated. The light is focused on Charge-Coupled Device (CCD) or Complementary Metal-Oxide Semiconductor (CMOS). The CCD or CMOS completes the conversion of optical signals to signals, and the electrical signals are converted into digital by analog to digital converter. The signal is converted into a digital image through digital signal processing.

In the camera imaging process, the sensor will leave sensor pattern noise (SPN) in any image taken, which is an inherent feature of digital cameras, which is mainly caused by photo response nonuniformity and fixed-pattern noise (FPN). Even with the same type of sensor, the output value of the photosensitive unit will be different, which produces PRNU. It is unique to a single sensor. Aiming at the complexity and polymorphism of camera imaging, [8] proposed a camera sensor output model:

$$I = g^\gamma \cdot [(1 + K)Y + \xi]^\gamma + \Theta. \quad (1)$$

Among them,  $I$  represents the noise image,  $K$  represents the multiplicative factor, which is the zero-average noise signal leading to PRNU, and  $g$  represents the color channel gain coefficient. The gain coefficient adjusts the pixel intensity level according to the sensitivity of the pixels in the red, green, and blue spectral bands to get the correct white balance.  $\gamma$  represents the gamma correction coefficient.  $\xi$  represents other noise.  $\Theta$  represents quantization noise. Equation (1) is expanded into Taylor's formula and expressed as:

$$I = I_0 + I_0 \cdot K^\gamma + \Theta. \quad (2)$$

Among them,  $I_0$  represents a clean image without noise.  $K$  stands for PRNU.  $\theta$  indicates that the noise includes fixed-pattern noise, quantization noise, shot noise, etc.

**3.1.2. PRNU Estimation.** The camera fingerprint  $K$  value can be estimated from  $N$  images taken by the camera. The specific process is as follows:

- (1) Use denoising filter  $F$  for denoising

$$I'_0 = F(I). \quad (3)$$

Among them,  $I'_0$  represents the image after removing the additive noise.  $F$  represents the denoising filter.  $I$  represents a noisy image.

- (2) Get noise residual

$$W = I - I'_0 = IK + \eta. \quad (4)$$

Among them,  $W$  represents the noise residual, and  $\eta$  is the set of all noises except multiplicative noise.

- (3) PRNU estimation

The maximum likelihood estimation can be used to estimate the value of  $K$ ; it can be expressed as [23, 24]:

$$K = \frac{\sum_{K=1}^N W_K I_K}{\sum_{K=1}^N I_K^2}. \quad (5)$$

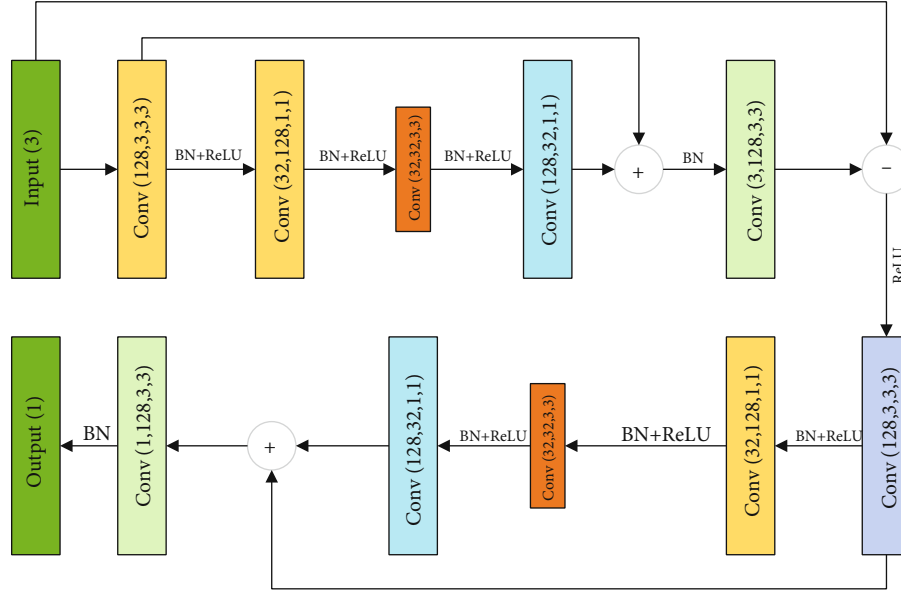


FIGURE 1: CSI-CNN network structure. This network is a fully convolutional network and does not change the length and width of the input image, its input is a 3-channel RGB image, and the output is a single-channel noise residual image.

**3.2. Image Noise Extractions.** After obtaining the PRNU of the device, it is necessary to extract the noise of the test image. CSI-CNN is designed in this section, noise fingerprint can be extracted through CSI-CNN, and the correlation calculation is performed with the preestimated PRNU value to determine whether the test image belongs to the corresponding device. This section proposes the CSI-CNN network model and introduces in detail how to build the network structure and the training process of the model.

**3.2.1. Network Structure.** The overall network structure of the proposed CSI-CNN is shown in Figure 1. The construction ideas mainly include:

- (1) The middle layer uses batch normalization (BN) and convolution kernel stacking ideas. The main reason for adopting this idea is that when the neural network is trained using minibatch in this paper, different batch data distributions are different, the network must learn to adapt to different distributions in each iteration, which will greatly reduce the training speed of the network. Using the BN method for data standardization can speed up the training process and improve the denoising performance. Using a stack of full convolution kernels allows the network to accept inputs of any size
- (2) The network structure design uses the bottleneck residual block and uses a  $1 \times 1$  convolution kernel to subtly reduce the feature dimension and reduce the number of network parameters, to prevent the occurrence of overfitting

According to the above network construction ideas, the input of CSI-CNN is the image to be tested  $y = kx + v$ , where  $k$  is multiplicative noise (noise fingerprint) [25],  $v$  is additive noise (background noise) [26], and  $x$  is clean image. Unlike

the SPN-CNN model training a set of models for one image data, the CSI-CNN proposed in this paper has better generalization. It can be applied to multiple cameras after one training and can achieve a good training effect. The network structure of CSI-CNN is shown in Figure 1. (1) Conv+ReLU: for the input layer,  $128 \times 3$  convolution kernels with a size of  $3 \times 3$  are used for convolution [27, 28], and ReLU (Rectified Linear Unit) is used to achieve nonlinear output between neurons. (2) Conv+ReLU+BN: for the self-network, this paper uses the bottleneck residual block, which passes through  $1 \times 1 \times 128$ ,  $3 \times 3 \times 32$ , and  $1 \times 1 \times 32$  convolution kernels, performs convolution, and performs batch normalization and ReLU activation function to output a 128-dimensional feature matrix. (3) Conv: for the output layer, a convolution kernel with a size of  $3 \times 3 \times 128$  is used to output the image multiplicative noise fingerprint  $w$ . Table 1 shows the parameter list of the network.

**3.2.2. Model Training.** Figure 2 shows the training framework of the CSI-CNN network proposed in this paper. First, the dataset is divided into a verification dataset, a fingerprint estimation dataset, a training dataset, and a test dataset at a ratio of 1:1:6:2. Then, we use the camera pictures in the fingerprint estimation set to estimate camera's fingerprint set by Section 3.1.2 PRNU estimation process, which is called  $K$ set. We start the network training process, randomly extract an image  $I$  from the training set, and take a subimage  $I'$  from it according to the preset size and then randomly one label  $\in \{0, 1\}$ ; when label = 1, take the source camera fingerprint image  $I$  from  $K$ set and take the subgraph  $K'$  at the same position as  $I'$  from  $K$  and output  $\{1, I', K'\}$  as a pair; when label = 0, randomly select a subgraph  $K''$  with the same size as  $I'$  from the set  $K$ set -  $K$  and output  $\{0, I', K''\}$  as a pair. In the experiment, each batch contains 64 pairs, and the default size of the subimage is  $64 \times 64$ .

The loss function designed in this paper uses the cosine distance to measure the similarity between the network output and the predicted PRNU value and calculates the loss through the idea of segmentation, and finally, uses it to update the parameters in the network. It enables the network to better extract the characteristics of noise fingerprint for camera source identification.

$$\text{loss}(x, y, l) = l \cdot \max(1 - \rho(x, y), 0) + (1 - l) \cdot [\min(\rho(x, y), 0) \cdot \varphi + |\rho(x, y)|]. \quad (6)$$

Among them,  $\rho(x, y) = (x \cdot y) / (\|x\| \cdot \|y\|)$ ,  $x$  represents the noise residual of a single image output by the network,  $y$  represents the camera fingerprint estimated by the method in Section 3.1.2,  $l \in \{0, 1\}$ ; when  $l = 1$ , it means that  $x$  and  $y$  are in the same position on the same camera; otherwise,  $l = 0$ . At that time, the loss function became

$$\text{loss}(x, y) = \min(\rho(x, y), 0) \cdot \varphi + |\rho(x, y)|. \quad (7)$$

This means that  $x$  and  $y$  are not from the same position of the same camera. We hope that  $\rho(x, y)$  is as close as possible to 0, whether it is from  $0^+$  to 0. It is still close to 0 from  $0^-$ , so the loss function adds  $|\rho(x, y)|$ . However, the two cases of  $\rho(x, y)$  taking positive and negative must be treated differently. When label = 1, the loss function becomes

$$\text{loss}(x, y) = \max(1 - \rho(x, y), 0). \quad (8)$$

This means that from the same position of the same camera, we should hope that  $\rho(x, y)$  is as close as possible to 1. This trend should be closer to  $1^-$ , and the loss function for  $\rho(x, y)$  the penalty for negative numbers is very large, so the loss becomes a number greater than 1. Different from the loss function,  $\text{MSE}(x, y) = (1/n) \sum_{i=1}^n (x_i - y_i)^2$  is proposed in [20]. The loss function based on cosine distance proposed in this paper measures the degree of similarity between image's noise fingerprint and camera's PRNU fingerprint in the direction, while the loss function in [20] can only measure the absolute difference in space between the two.

## 4. Experimental Verification and Result Analysis

**4.1. Dataset Description and Data Preprocessing.** To evaluate the performance of camera source identification of the proposed method, we use the following four datasets for testing.

**4.1.1. Vision.** This dataset was established by [29]. The images and videos of this standard evaluation library come from 35 widely used smart phones, including 11 different brands in total: *Apple, Asus, Huawei, Lenovo, LG Electronics, Microsoft, OnePlus, Samsung, Sony, Wiko, and Xiaomi*. They collected 7565 mobile phone images on Facebook, 34427 images through WhatsApp, and 1914 videos downloaded from WhatsApp and YouTube. Each model of mobile phone images are divided into five categories: smooth images, original phone images, high-quality Facebook images, low-

TABLE 1: The number of network parameters in CSI-CNN.

Network layer name	Parameter number
Convolution layer 1	$128 \times 3 \times 3 \times 3 + 128 = 3584$
BatchNorm	$128 + 128 = 256$
Activation function ReLU	0
Convolution layer 2	$32 \times 128 \times 1 \times 1 = 4096$
BatchNorm	$32 + 32 = 64$
Activation function ReLU	0
Convolution layer	$32 \times 32 \times 3 \times 3 = 9216$
BatchNorm	$32 + 32 = 64$
Activation function ReLU	0
Convolution layer 4	$128 \times 32 \times 1 \times 1 = 4096$
Function ADD	0
BatchNorm	$128 + 128 = 256$
Convolution layer 5	$3 \times 128 \times 3 \times 3 + 3 = 3459$
BatchNorm	$3 + 3 = 6$
Activation function ReLU	0
Function sub	0
Activation function ReLU	0
Convolution layer 6	$128 \times 3 \times 3 \times 3 + 128 = 3584$
BatchNorm	$128 + 128 = 256$
Activation function ReLU	0
Convolution layer 7	$32 \times 128 \times 1 \times 1 = 4096$
BatchNorm	$32 + 32 = 64$
Activation function ReLU	0
Convolution layer 8	$32 \times 32 \times 3 \times 3 = 9216$
BatchNorm	$32 + 32 = 64$
Activation function ReLU	0
Convolution layer 9	$128 \times 32 \times 1 \times 1 = 4096$
Function ADD	0
Convolution layer 10	$1 \times 128 \times 3 \times 3 + 1 = 1153$
BatchNorm	$1 + 1 = 2$

quality Facebook images, and WhatsApp images. The resolutions of the image are  $1280 \times 720$  and  $640 \times 480$ .

**4.1.2. Kaggle.** This dataset comes from a competition on the identification of mobile phone image sources held on the Kaggle [30] website. This competition also provides participants with a standard evaluation library. The standard evaluation library is divided into two parts, one is the training library, the other is the evaluation library. The images in the training library come from 10 mobile phones, with a total of 2750 images. Each mobile phone took 275 images, and the content of these images is selected from different scenes. The evaluation library includes a total of 2640 images. These images come from the same model of the mobile phone as the training library, but not the same mobile phone. Half of the images have been manually processed, compressed, and enlarged in different proportions,

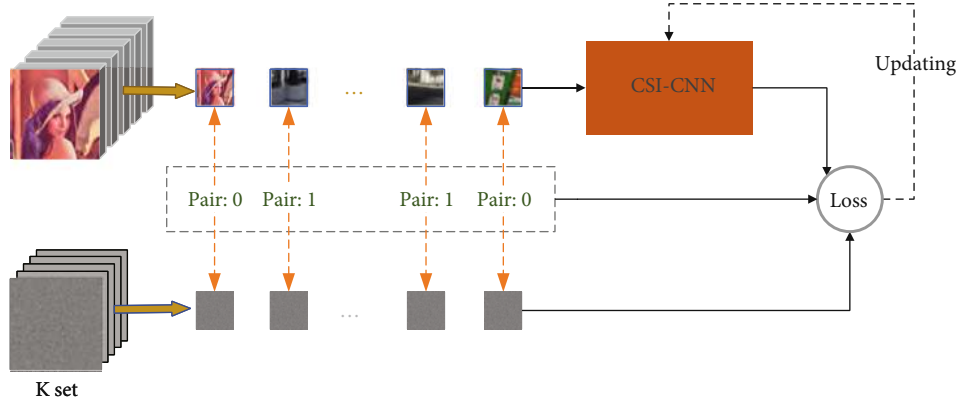


FIGURE 2: CSI-CNN network training process.

TABLE 2: Correlation calculation results on our dataset.

Model Image	Honor 10	iPhone 6	Nubia Z17	Redmi note8	Galaxy S5	Wavelet
Honor 10	0.0767	-0.0068	-0.0040	0.0058	-0.0005	0.0591
iPhone 6	-0.0067	0.0990	-0.0051	0.0026	-0.0052	0.0777
Nubia Z17	-0.0047	-0.0056	0.0670	0.0002	0.0013	0.0474
Redmi note8	0.0035	-0.0027	0.0051	0.1033	0.0024	0.0821
Galaxy S5	-0.0136	-0.0250	0.0042	0.0088	0.2610	0.2044

and some have been gamma-corrected, and the image size has been cropped to  $512 \times 512$ .

**4.1.3. Daxing.** This dataset was established by [31]. It collects images and videos from a wide range of smart phones of different brands, models, and devices. The dataset includes 43,400 images and 1,400 videos, which were taken by 90 smart phones of 22 models from 5 brands.

**4.1.4. Proposed.** Due to the small number of pictures of a single camera in the above datasets, the model cannot be trained well. To better estimate the performance of the algorithm proposed in this paper, we use 5 different models of mobile phones, including *iPhone 6*, *Galaxy S5*, *Nubia Z17*, *Redmi note8*, and *Honor 10*. We randomly took 1000 different images with each model of mobile phone.

This paper preprocesses the collected dataset. First, all images in the dataset are cropped into blocks in the central area and then are randomly selected as the input data of CSI-CNN from the cropped blocks for training.

**4.2. Comparison Method and Evaluation Index.** During the experiment, this paper selects different control methods and evaluation indicators according to different experimental purposes, and all comparison methods are experimented on the datasets used in this paper.

**4.2.1. Comparison Method.** When evaluating the denoising model, this paper compares with the wavelet filter denoising model and DnCNN [21], using these methods to obtain the noisy image of the downloaded image on the social platform and correlate the noisy image with the extracted noise fin-

gerprint calculation to obtain the correspondence between the camera and the social platform image.

**4.2.2. Evaluation Index.** When evaluating the performance of the CSI-CNN network model, this paper uses accuracy (ACC), receiver-operating characteristic (ROC), and area under curve (AUC) as evaluation indicators, which are defined as follows:

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FP}}. \quad (9)$$

ROC curve, the abscissa of the curve, is the false-positive rate (FPR), and the ordinate is the true case rate (TPR).

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \quad (10)$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (11)$$

Among them, TP represents the number of samples taken by a certain camera and classified by the model as belonging to the camera. FP represents the number of samples that the image does not belong to a certain camera but is classified as belonging to this camera by the model. FN represents the number of samples whose images did not belong to a certain camera and were classified by the model as not belonging to the camera. TN represents the number of samples taken by a certain camera and classified by the model as not being taken by the camera. AUC is the area under ROC.

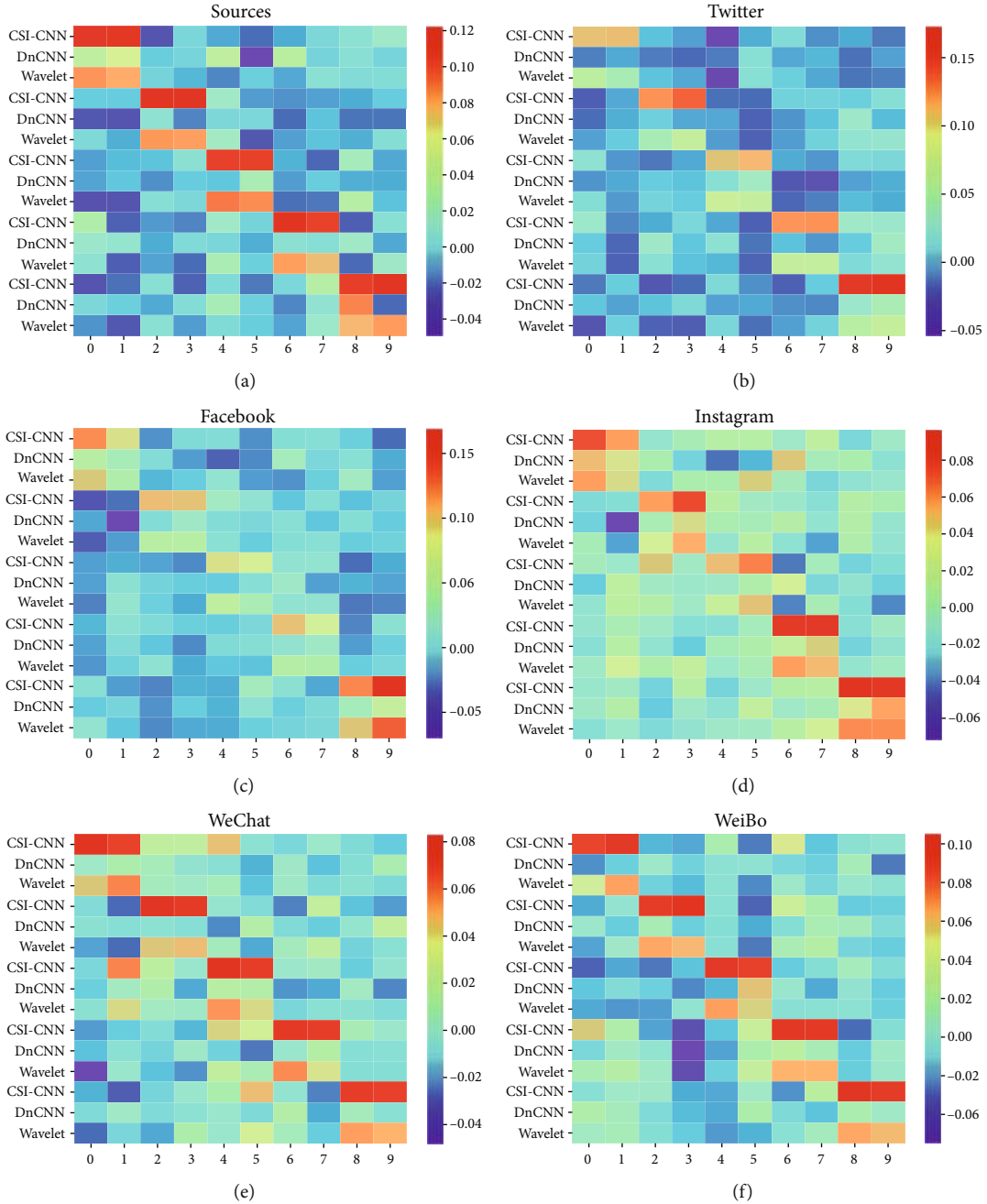


FIGURE 3: (a–f) The thermodynamic diagram of the correlation coefficient obtained by uploading our dataset to five platforms and taking two pictures randomly from each camera in the dataset.

It is equivalent to Mann-Whitney  $U$  test and can be calculated as follows [32]:

$$\text{AUC} = \frac{\sum_{i \in p} \text{ran } k_i - ((M+1)M)/2}{MN}. \quad (12)$$

Among them,  $M$  represents the number of positive samples, and  $N$  represents the number of negative samples.  $p$  represents a positive sample.  $\text{ran } k_i$  represents the descending rank of  $i$  in the sample set.

**4.3. Experimental Results and Evaluation.** The running environment of the experiment is Ubuntu16.04LTS operating system equipped with PyTorch1.0.1 and Python3.6. The experiments are run on NVIDIA graphics card GeForce GTX1080-Ti.

This paper uses four datasets and uploads them to five social platforms to obtain twenty different datasets. Experiments are performed on these datasets, respectively, and the correlation between the noise fingerprint obtained by CSI-CNN and the PRNU camera fingerprint is estimated by Section 3.1.2. Use correlation as a basic research object for performance analysis and evaluation.

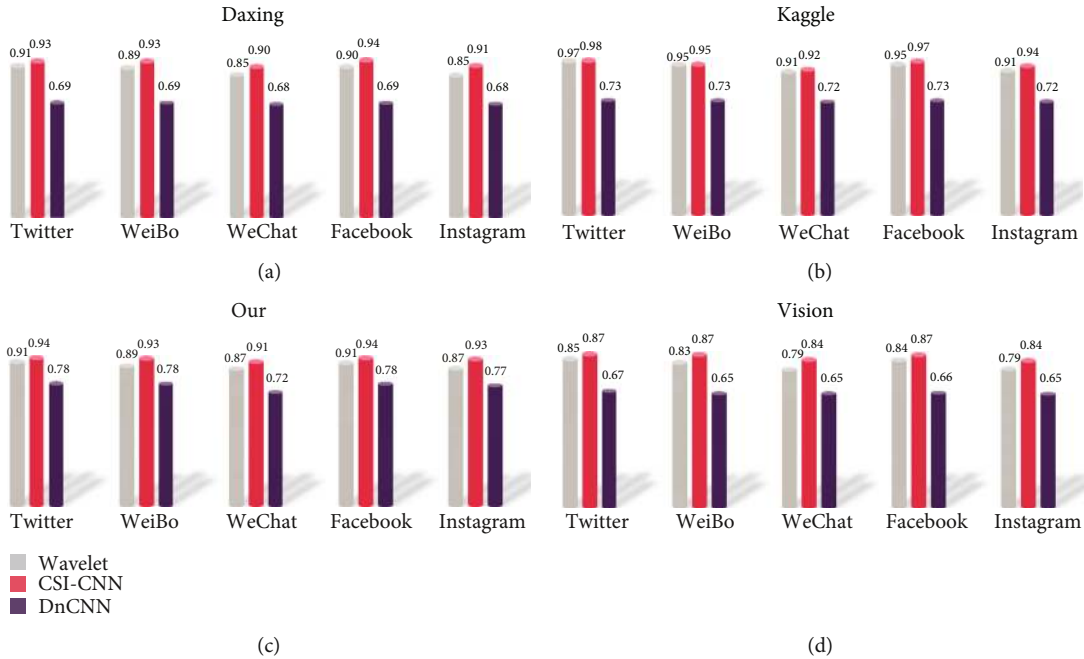


FIGURE 4: (a–d) The histogram representation of AUC values obtained by applying three different algorithms to four datasets on five platforms.

**4.3.1. Image Denoising Experiment Results and Performance Comparison.** In order to perform camera source identification on the image data of the social network platform, it is first necessary to extract noise fingerprints from the image. The quality of noise fingerprint extraction directly affects the performance of camera source identification. In the Our dataset, this paper randomly takes out 200 images from the test dataset of each camera and tests the mean value of the correlation coefficient with each camera’s fingerprint. The results are shown in Table 2. The experimental results show that the algorithm in this paper can extract the noise fingerprint of the picture very well.

**4.3.2. Camera Source Recognition Experiment and Performance Comparison.** In order to test the performance of the CSI-CNN camera source identification method proposed in this paper. We perform camera source identification by correlating the noise fingerprint extracted by CSI-CNN with the corresponding camera fingerprint estimated in Equation (5). This paper compares the performance from the four aspects of NCC, ACC, ROC curve, and AUC value. The experiment shows the universality and robustness of CSI-CNN in image traceability. Figure 3 shows NCC between the acquired image and the corresponding camera recognition after the four datasets are uploaded to five social networking platforms and are compared with the experimental results of DnCNN and wavelet denoiser. Experimental results show that the NCC identified by CSI-CNN is higher than those identified by DnCNN and wavelet filters.

In order to better analyze and evaluate the proposed CSI-CNN camera source identification algorithm, we use the representative indicators of deep learning-related performance evaluation to analyze and evaluate its performance.

TABLE 3: Accuracy comparison (ACC).

Dataset	Platform	Algorithm		
		CSI-CNN	Wavelet	DnCNN
Our	Twitter	0.950	0.887	0.453
	Facebook	0.864	0.792	0.580
	Instagram	0.836	0.708	0.552
	WeChat	0.803	0.727	0.423
	WeiBo	0.937	0.843	0.460
Kaggle	Twitter	0.936	0.928	0.508
	Facebook	0.928	0.880	0.524
	Instagram	0.840	0.756	0.492
	WeChat	0.813	0.808	0.436
	WeiBo	0.892	0.864	0.508
Vision	Twitter	0.744	0.696	0.384
	Facebook	0.708	0.676	0.352
	Instagram	0.660	0.560	0.372
	WeChat	0.648	0.548	0.368
	WeiBo	0.728	0.604	0.368
Daxing	Twitter	0.832	0.780	0.420
	Facebook	0.836	0.880	0.524
	Instagram	0.840	0.768	0.424
	WeChat	0.764	0.652	0.408
	WeiBo	0.808	0.768	0.428

Figure 4 shows that the test images in Our, Kaggle, Vision, and Daxing datasets are uploaded to Twitter, Facebook, WeChat, Instagram, and Weibo, and their AUC values are calculated. The experimental results show that the method proposed in this paper has better results.



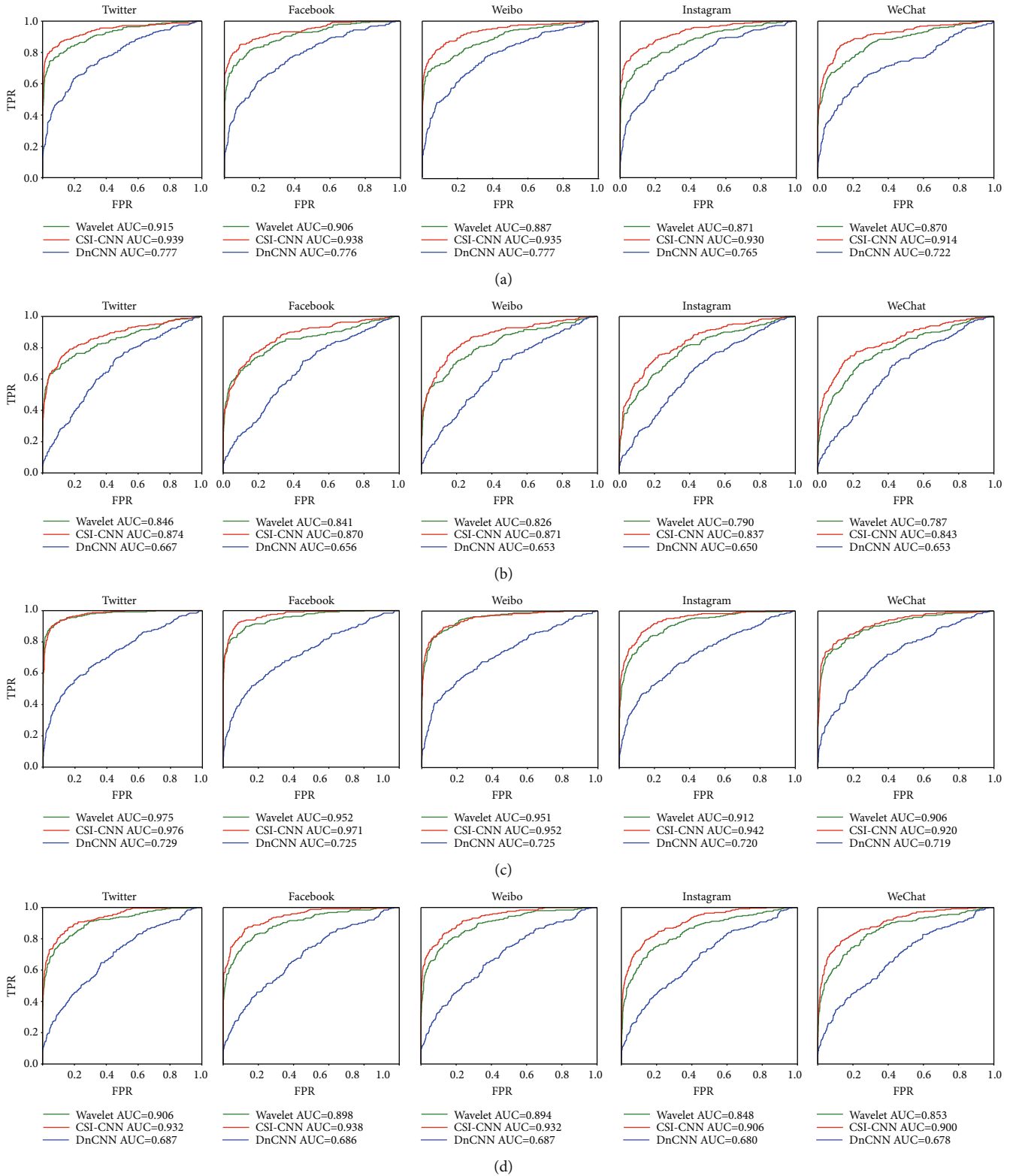


FIGURE 5: The ROC curves obtained by applying three different algorithms to four datasets on five platforms. (a-d) Our, Vision, Kaggle, and Daxing datasets.

In this paper, the camera with the largest correlation coefficient with the image is used as the source camera, and on this basis, the ACC value is calculated. Table 3 shows the ACC value of image camera source identification down-

loaded by social network platforms. The experimental results show that for five social network platform images with different quality factors, CSI-CNN has a higher ACC value than the current popular wavelet filter and DnCNN

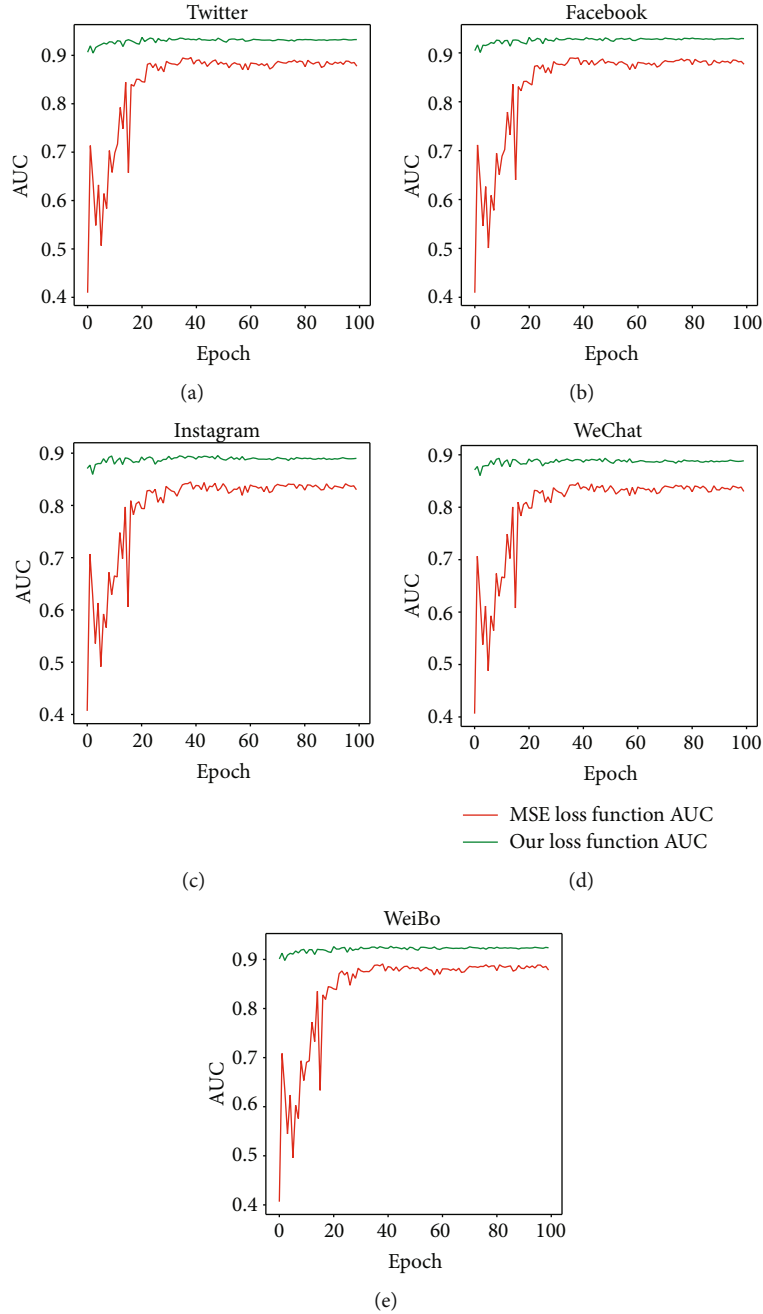


FIGURE 6: The mean-square error (MSE) loss function and Our loss function were tested on five platforms, respectively, and the AUC change curve on each epoch.

camera source recognition algorithm. Also, compared with other datasets, Vision and Daxing datasets have very low accuracy of these three algorithms. The fundamental reason is that there are a lot of flat images in the Vision and Daxing datasets, such as the blue sky, white clouds, and walls. After compression by the social platform compression algorithm, the flat image has a serious loss of high-frequency noise information, which makes it impossible to extract effective noise fingerprints, to calculate the correlation with the device fingerprint.

To further evaluate the performance of the algorithm designed in this paper, the image camera source recognizes

the ROC curve (as shown in Figure 5). The experimental results show that for the five social network platform images, CSI-CNN and the currently popular DnCNN and wavelet filter camera source recognition have good performance.

In order to improve the accuracy of camera source recognition, this paper designs a new loss function. To test the effectiveness of the loss function proposed in this paper, we use the Daxing dataset for training and testing, and the ratio of the training set to the test set is 3:1. The initial learning rate is 0.001 and iterates 100 epochs, and each iteration is 30 times, and the learning rate becomes 0.2 times of the original for model training. As shown in Figure 6,

experimental results show that compared with the loss function proposed in [20], the loss function can make the model converge faster, and the training result is more stable.

## 5. Conclusion

Multimedia forensics is an important research topic in the field of computer security. The combination of online social networks and smart phones is of great significance to crime prevention, evidence collection, and the security of IoT devices. In this paper, a CSI-CNN is proposed to extract noise fingerprints from pictures on social networks and match the extracted noise fingerprint with camera fingerprints to identify the camera source. We conduct experiments on five online social network platforms with different image compression levels. The experimental results show that the CSI-CNN network model proposed in this paper has a higher recognition effect than the current popular DnCNN and wavelet filter camera source recognition algorithms.

With the development of deep learning and the diversification of forensic data, the method proposed in this paper may have some limitations. To overcome these problems, we will use pure deep learning methods to train the features of a large number of heterogeneous forensic data and extend the research object to the video data of social networks.

## Data Availability

The labeled datasets used to support the findings of this study can be provided on request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The research presented in this paper is supported in part by the National Natural Science Foundation (No. U20B2050) and the Youth Innovation Team of Shaanxi Universities (No. 2019-38).

## References

- [1] S. Li, L. D. Xu, and S. Zhao, "5G Internet of Things: a survey," *Journal of Industrial Information Integration*, vol. 10, pp. 1–9, 2018.
- [2] C. Zhang and Y. Lu, "Study on artificial intelligence: the state of the art and future prospects," *Journal of Industrial Information Integration*, vol. 23, p. 100224, 2021.
- [3] S. Li, Q. Sun, and X. Xiaolong, "Forensic analysis of digital images over smart devices and online social networks," in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1015–1021, Exeter, UK, June 2018.
- [4] S. Kemp, "Digital 2020: 3.8 billion people use social media. Website," 2020, <https://wearesocial.com/blog/2020/01/digital-2020-3-8-billion-people-use-social-media>.
- [5] C. Pasquini, C. Brunetta, A. F. Vinci, V. Conotter, and G. Boato, "Towards the verification of image integrity in online news," in *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 1–6, Turin, Italy, June 2015.
- [6] M. Chernyshev, S. Zeadally, Z. Baig, and A. Woodward, "Mobile forensics: advances, challenges, and research opportunities," *IEEE Security & Privacy*, vol. 15, no. 6, pp. 42–51, 2017.
- [7] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: a new approach towards general purpose image manipulation detection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2691–2706, 2018.
- [8] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.
- [9] T. Filler, J. Fridrich, and M. Goljan, "Using sensor pattern noise for camera model identification," in *2008 15th IEEE International Conference on Image Processing*, pp. 1296–1299, San Diego, CA, USA, October 2008.
- [10] C.-T. Li, "Source camera identification using enhanced sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 280–287, 2010.
- [11] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva, "PRNU-based forgery detection with regularity constraints and global optimization," in *2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSp)*, pp. 236–241, Pula, Italy, September 2013.
- [12] L. Debiasi and A. Uhl, "Blind biometric source sensor recognition using advanced PRNU fingerprints," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 779–783, Nice, France, August 2015.
- [13] B. Balamurugan, S. Maghilan, and M. R. Kumar, "Source camera identification using SPN with PRNU estimation and enhancement," in *2017 International Conference on Intelligent Computing and Control (I2C2)*, pp. 1–6, Coimbatore, India, June 2017.
- [14] J. Wang, G. Wu, J. Li, and S. K. Jha, "A new method estimating linear Gaussian filter kernel by image PRNU noise," *Journal of Information Security and Applications*, vol. 44, pp. 1–11, 2019.
- [15] L. Baroffio, L. Bondi, P. Bestagini, and S. Tubaro, "Camera identification with deep convolutional networks," 2016, <https://arxiv.org/abs/1603.01068>.
- [16] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification with the use of deep convolutional neural networks," in *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, Abu Dhabi, United Arab Emirates, December 2016.
- [17] P. Yang, R. Ni, Y. Zhao, and W. Zhao, "Source camera identification based on content-adaptive fusion residual networks," *Pattern Recognition Letters*, vol. 119, pp. 195–204, 2019.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [19] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for oneshot image recognition," in *ICML deep learning workshop*, vol. 2, Lille, 2015.

- [20] M. Kirchner and C. Johnson, "SPN-CNN: boosting sensor-based source camera attribution with deep learning," in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, Delft, Netherlands, December 2019.
- [21] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.
- [22] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, Boston, MA, USA, June 2015.
- [23] S. Chakraborty and M. Kirchner, "PRNU-based image manipulation localization with discriminative random fields," *Electronic Imaging*, vol. 2017, no. 7, pp. 113–120, 2017.
- [24] P. Korus and J. Huang, "Multi-scale analysis strategies in PRNU-based tampering localization," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 809–824, 2016.
- [25] D. L. Wang and J. Chen, "Supervised speech separation based on deep learning: an overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.
- [26] D. Cozzolino and L. Verdoliva, "Noiseprint: a CNN-based camera model fingerprint," 2018, <https://arxiv.org/abs/1808.08396>.
- [27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, <https://arxiv.org/abs/1409.1556>.
- [28] A. Rakhlin, *Convolutional Neural Networks for Sentence Classification*, GitHub, 2016.
- [29] D. Shullani, M. Fontani, M. Iuliani, O. A. Shaya, and A. Piva, "Vision: a video and image dataset for source identification," *EURASIP Journal on Information Security*, vol. 2017, no. 1, Article ID 15, 2017.
- [30] "IEEEs Signal Processing Society camera model identification," 2019, <https://www.kaggle.com/c/sp-society-cameramodel-identification>.
- [31] H. Tian, Y. Xiao, G. Cao, Y. Zhang, Z. Xu, and Y. Zhao, "Daxing smartphone identification dataset," *IEEE Access*, vol. 7, pp. 101046–101053, 2019.
- [32] S. J. Mason and N. E. Graham, "Areas beneath the relative operating characteristics (ROC) and relative operating levels (rol) curves: statistical significance and interpretation," *Quarterly Journal of the Royal Meteorological Society*, vol. 128, no. 584, pp. 2145–2166, 2002.