

Article Information

Title	CNN-based Camera Model Identification Using Image Noise in Frequency Domain
Authors	Tiantian Cai; Zhanjian Shao; Yoichi Tomioka; Yuanyuan Liu; Zhu Li
Citation	2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 2019, pp. 3518-3524, doi: 10.1109/SMC.2019.8914375.
Copyright	© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Note	<p>This is the author's accepted version of the paper published in <i>Proceedings of 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)</i>.</p> <p>The final version is available at IEEE Xplore DOI: https://doi.org/10.1109/SMC.2019.8914375</p>

CNN-based Camera Model Identification using Image Noise in Frequency Domain

Tiantian Cai ^{*}, Zhanjian Shao ^{*}, Yoichi Tomioka [†], Yuanyuan Liu ^{*}, Zhu Li ^{*}
^{*} College of Electronic and Information Engineering, Hangzhou Dianzi University
Hangzhou, China

Email: {caitian, szj, liuyuan, lz1126}@hdu.edu.cn

[†] School of Computer Science and Engineering, the University of Aizu
Aizu-Wakamatsu, Japan

Email: ytomioka@u-aizu.ac.jp.

Abstract— Camera model identification has been studied extensively within digital image forensics as a deterrence to secret photography and image forgery. The feasibility of convolutional neural networks (CNNs) has been proven for an image classification algorithm. CNN-based algorithms have been proposed for camera model classification, and they focus on training with image/noise in a spatial domain. However, because the periodic characteristics of image noise are one of the essential types of information for model classification, it is more efficient to train CNN models with image noise in a frequency domain. In this paper, we propose a CNN-based approach for camera model classification that trains a CNN model with high-frequency components of images in the frequency domain. In the experiments, we evaluated the accuracy of camera model/brand classification using a Dresden image dataset. We achieved 97.35% and 99.32% accuracy, respectively, for 14-model classification of 256×256 image patches and full images. Using this approach, our results indicated a 1.84% and 1.35% improvement, respectively, compared with a state-of-the-art method. We also achieved 100% accuracy for 10-brand classification.

I. INTRODUCTION

With the rapid adoption of digital cameras and the widespread use of image processing software, such as Photoshop, iPhoto and ACDSee, editing and tampering with digital photos has become very easy. At the same time, related information security issues are becoming more and more prominent. If forged images are widely used in formal media, scientific research, and insurance evidence, this will seriously affect social systems that depend on reliable visual information. Digital image forensics technology came into being to ensure the authenticity and integrity of the image [1], [2]. Image Forensics mainly includes source identification [3] and forgery detection [4]. As an important research direction of digital image forensics technology, the purpose of source camera identification is to determine the imaging device from the acquired image itself, which is specifically divided into two problems: (1) to classify images or videos according to the type of device that generates them [5], [6]; (2) to detect specifically which device was used to record a particular digital image or video [7], [8]. Source camera forensics can be used to identify the authenticity and legitimacy of images in

court presentations, as well as to track the source of illegal videos and images.

An imaging device leaves some traces in an image during its generation process. These traces are caused by the characteristics of the physical device, because some device characteristics have device uniqueness which it is difficult to eliminate. We can use such device characteristics as the fingerprint of a device. At present, the source identification of digital cameras can be divided roughly into two categories: source identification based on correlation detection [9], [10], [11], [12], and source identification based on pattern classification [13], [14], [15], [16], [17], [18].

Source identification based on correlation detection is used to determine the source by extracting “device fingerprints” from different images created by the same camera and comparing their correlations. Lukas *et al.* proposed a method using sensor pattern noise as a unique fingerprint for camera identification in [9], and it was extended in [10], [11], and [12].

Source camera identification based on pattern classification can be used to classify images according to some characteristics of the source camera using machine learning techniques. In this approach, it is important to extract appropriate image features. In [14] and [15], features extracted in a frequency domain are used for model or brand classification. Tuama *et al.* [17] first proposed a camera model classification method for image patches based on convolutional neural networks (CNNs). Also, the CNN-based model classification methods for large images have been proposed in [5] and [18]. In these approaches, a CNN model can learn automatically in the training stage what kinds of features are appropriate for source model classification. As shown in [5], [17] and [18], CNN has a high potential to achieve highly accurate model classification. In [17], noise is extracted from the input images and inputted to CNN. In [5] and [18], images are input directly to a CNN model. Both methods feed inputs to CNN models and train them in a spatial domain.

On the other hand, as shown in [14] and [15], features extracted in a frequency domain play an important role in model or brand classification. For example, the periodic characteristic of image noise is one of the important types of

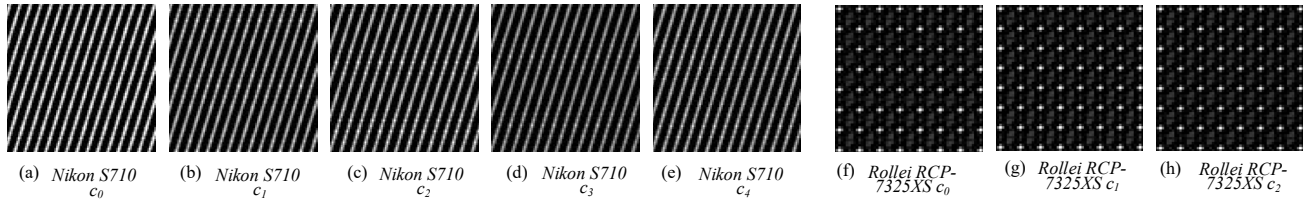


Figure 1. The autocorrelation images of estimated PRNU noise of each device. (a)-(e) are the autocorrelation images from *Nikon CoolPixS710*. (f)-(h) are the autocorrelation images from *Rollei RCP-7325XS*.

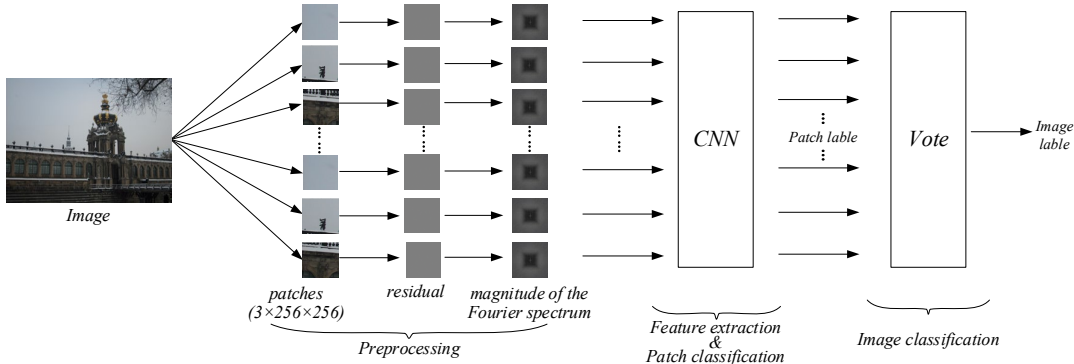


Figure 2. The process of the proposed method for model classification.

information for model classification. Such features can be seen as peak signals of a frequency domain.

In this paper, we propose a CNN-based model/brand classification method that trains CNN using image noise in a frequency domain. We divide an input image into patches, extract noise from the patches, and transform the noise by fast Fourier transform (FFT). We train CNN with the magnitude of the Fourier spectrum of the image noise. Based on the majority voting of classification results of all patches, we identify the source model/brand of the input image. To the best of our knowledge, this is the first CNN-based model/brand classification method that trains a CNN model in a frequency domain. Based on training in the spatial domain, we enable a smaller CNN model to achieve a higher accuracy for camera model classification. In the experiments, we evaluated the accuracy of camera model/brand classification using a Dresden image dataset. We achieved 97.35% and 99.32% accuracy, respectively, for 14-model classification of 256×256 image patches and full images. For these results, we gained a 1.84% and 1.35% improvement, respectively, compared with a state-of-the-art CNN-based approach proposed in [17]. We also achieved 100% accuracy for a 10-brand classification of full images.

The rest of this paper is organized as follows. Section II introduces some background knowledge of camera model classification and camera brand classification. Section III shows the proposed method for camera model classification based on CNN using frequency domain features. Section IV introduces the details of the experimental setup and the experimental results. Finally, we summarize the paper in Section V.

II. MODEL/BRAND CLASSIFICATION

A. Problem definition

The technology of camera model identification depends on the assumption that images obtained from the same device

have inherent features. These features are only related to the unique hardware structure and imaging algorithm of the device. Even if the scene is the same, there are some subtle differences among the generated digital images which are from different devices. Extracting and analyzing these different features enables source identification of the image generation device.

Image source identification is used to determine what acquisition device generates a given image. The technology includes the following aspects: the type of imaging device, the device brand, the device model, and even the device individual. This paper focuses on digital camera brand/model classification issues. Camera brand classification aims to identify the camera brand attribute of the images, such as *Canon* or *Sony*. The purpose of the camera model classification is to identify the camera model attribute of the images, such as the *Canon Ixus 55* or the *Canon Ixus 70*. In practical applications, we can choose the appropriate classification target according to different needs.

B. Periodic Feature in Image Noise

There are several types of pattern noises in an image acquired by a digital imaging device, which are common to images acquired by the same device/model/brand. Photo-response non-uniformity noise is one type of pattern noise, and it is generated by the sensitivity difference of an image sensor. Because it is a unique characteristic of each imaging device, we can use PRNU noise for camera identification. Moreover, there is pattern noise called non-unique artifact (NUA). NUA is caused by image sensor design, color filter pattern, lens characteristics, and so on. Some NUA are held in common by images sourced from the same model/brand and different from images sourced from other models/brands. Thus, they are useful for model/brand classification and distinction.

In [19], Thomas *et al.* reported that periodic NUA patterns could be identified using the autocorrelation image of image

noise. We show examples of NUA patterns for two camera models in Figure 1. They were generated as in [19]. We estimated PRNU noise using 50 flat-filed images. Because it is difficult to extract only PRNU noise, this estimated PRNU noise included a component of NUA. Then, we calculated the autocorrelation of the estimated PRNU noise. We can see that *Nikon S710* has a diagonal periodic NUA pattern and *Rollei RCP-7325XS* has a grid-like NUA pattern. There are several types of periodic features in each image noise pattern, and these are significant clues for model/brand classification. It is more difficult to extract such periodic patterns from one image than PRNU noise estimated from multiple images. Therefore, it is important to determine how to identify and extract periodic features useful for model/brand classification.

The methods proposed in [5], [17], and [18] use CNN to extract appropriate features for model classification from an input image. These methods input an image or extracted noise represented in a spatial domain to a CNN model. Therefore, their CNN model starts the analysis from a spatial domain. However, in order to identify the frequency characteristics of periodic NUA patterns, the Fourier spectrum of image noise is more suitable. We can detect the dominant periodic signals as peaks in the Fourier spectrum. Therefore, in this paper, we propose a CNN-based camera model/brand classification approach using the frequency domain feature of image noise.

III. THE PROPOSED METHOD FOR CAMERA MODEL/BRAND IDENTIFICATION

This section introduces the detailed process of camera model identification proposed in this paper (see Figure 2.). This method consists of four steps: (1) image preprocessing to divide an image into patches and obtain noise from each patch, (2) feature extraction of each patch, (3) patch classification, and (4) image classification using majority voting according to the results of patch classification.

A. Image preprocessing

The first step involves dividing each image into a set of non-overlapping patches of size $3 \times 256 \times 256$ (i.e., channels \times height \times width). Dividing the image into patches can get more CNN training data. Moreover, by using smaller patches, we can use a smaller and lighter CNN architecture in the next step. Then, for each patch, we extracted noise for the three channels R, G, and B, respectively.

As in [17], we used two filtering methods to obtain the noise: a high-pass filter and a wavelet-based denoising filter [20].

In this paper, the high-pass filtering method in [21] is used to obtain the high-frequency component:

$$HFC(I) = I * \frac{1}{12} \begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix} \quad (2)$$

where I is an input image. We refer to this type of noise as a high-frequency component (HFC) in this paper.

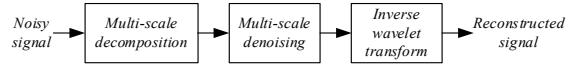


Figure 3. Flow chart of wavelet denoising.

Figure 3 shows the basic steps involved in a wavelet-based denoising filter [20]. The multi-scale wavelet transform is performed on an image. Then, the wavelet coefficients of the signal are extracted at each scale. The wavelet coefficients potentially related to noise are removed from each sub-band. Finally, the inverse wavelet transform is used to reconstruct the signal. The noise can be calculated by subtracting the denoised image from the original image. We refer to this type of noise as wavelet-based residual noise (WRN) in this paper.

As mentioned in Section II, the extracted image noise contains periodic features, which can be used for camera model classification. It is known that the Fourier spectrum can describe periodic image textures as a set of peaks. It is easier for convolutional layers and pooling layers to detect peaks in a frequency domain than to extract a periodic pattern from a spatial domain. Therefore, we apply the FFT to the extracted noise. In this paper, we use the magnitude of the Fourier spectrum of HFC/WRN extracted from each patch as an input to CNN.

B. Feature extraction

The frequency domain feature image is fed to a CNN model for feature extraction through the convolutional layers and the pooling layers.

Convolutional layers are used to learn the characteristics of input data. We usually use multi-layer convolution to get deeper feature maps. The convolutional layer consists of a number of convolutional kernels that perform partial window sliding on the transmitted feature map to obtain a new feature map. The weight of the convolution kernel is obtained through learning and will not be changed during the convolution process. That is the idea of parameter sharing, which shows that we extract the same features of different positions of the original image through a convolution kernel. With more kernels, we can extract more different image features, and the network will have better performance.

To give the CNN nonlinear properties, an activation function follows a convolutional layer. The commonly used functions are *sigmoid*, *tanh*, *ReLU* functions. *ReLU* does not require input normalization to prevent the functions from reaching saturation. The *ReLU* activation function increases the sparseness of the network, makes the extracted features more representative, and can improve the generalization performance of the network. We can see that most of the advanced convolutional neural networks basically use the *ReLU* function or its extensions such as *ReLU6* and *Leaky ReLU*. In this paper, we also use *ReLU* as an activation function.

The pooling layer is often behind the convolutional layer. The result of pooling is to reduce feature dimension and reduce parameters. The purpose of pooling is to maintain invariance (rotation, translation, scaling, etc.), and to avoid over-fitting and improve the generalizability of the model. The most common pooling operations are mean pooling and max pooling. The mean pooling is to calculate the average of its

TABLE I. EXPERIMENTAL DATASET FOR CAMERA MODEL CLASSIFICATION

	Brand	Model	Num. of image patches
1	AgfaPhoto	DC-733s	30349
2		DC-830i	39204
3		Sensor 530s	55585
4	Canon	Ixus 55	15680
5	Fujifilm	FinePix J50	22680
6	Kodak	M1063	64960
7	Nikon	D200	55800
8	Olympus	M1050SW	28560
9	Panasonic	DMC-FZ50	37100
10	Panasonic	DMC-FZ50	14620

TABLE II. CLASSIFICATION ACCURACY OF IMAGE PATCHES

	Method	HFC 1171	HFC+FFT ours	WRN 1171	WRN+FFT ours
12	AlexNet	99.06%	99.17%	97.05%	98.93%
	GoogleNet	99.30%	99.46%	99.25%	99.29%
	Tuama's Net	99.04%	99.19%	98.54%	98.88%
14	AlexNet	95.33%	96.75%	96.15%	95.86%
	GoogleNet	95.51%	97.35%	96.53%	96.35%
	Tuama's Net	94.37%	96.48%	94.48%	95.45%

neighbor pixels as the pooled value for each pixel, it can reduce the variance of the estimated value caused by the limited size of the neighborhood and retain more background details. The max pooling is to select the maximum value as the pooled value, it can reduce the offset of the estimation mean value caused by the error of the convolutional layer parameter and retain more texture details [22].

We train our CNN model so that deeply cascaded convolutional, activation, and pooling layers can extract appropriate features from frequency domain images contaminated by signals corresponding to random noise and scene contents.

C. Patch Classification

In the fully connected layer, each neuron is fully connected to all the neurons in the previous layer. Its main function is to combine the local features that extracted from the previous convolutional layer and classify them [23]. The fully connected layer is a traditional multilayer perceptron that uses a softmax activation function at the output layer. The softmax

function turns a vector whose input is an arbitrary value into a vector whose range is 0–1, where the sum of all values in the vector is 1. We can get the prediction label for each patch, which corresponds to the label with the maximum output.

D. Image Classification

Since each part of an image contain unique information, the richness of the information is different. Thus, the classification label of the full image is decided by majority voting based on statistical analysis of patch classification results.

IV. EXPERIMENTS AND EVALUATIONS

This section consists of three experiments: the first experiment is to evaluate the performance of our proposed model classification method for single patches; the second experiment is to evaluate image classification accuracy of the proposed model classification method using the majority voting method; the third experiment is to evaluate the performance of the proposed method for camera brand classification. In the model and brand classification, we compare the proposed method with state-of-the-art methods proposed in [17] and [14], respectively.

For each experiment, we use the images from the standard benchmark Dresden database [24]. The database contains 74 camera instances of 27 different camera models, and they are all JPEG formats with a quality factor (QF) of over 75. In the three experiments, we set the patch size to 256×256. In this paper, to demonstrate the effectiveness of training in a frequency domain, we use three different CNN models. They are AlexNet [23], GoogleNet [25], and the CNN model proposed by Tuama *et al.* [17] which is referred to as Tuama's Net in this paper. AlexNet was proposed by Krizhevsky *et al.* in 2012. It is a classic model of CNN in image classification. AlexNet is an 8-layer deep network, in which 5 layers of convolutional layers and 3 layers of fully connected layers, excluding the LRN layer and the pooling layer. GoogleNet was proposed by Szegedy *et al.* in 2014. Its core is to propose an Inception structure, which enables the network to extract more features while maintaining the same amount of computation and improve the utilization efficiency of the network for computing resources. GoogleNet is a 22-layer deep network, excluding the pooling layer. Tuama's Net is a relatively small net obtained by tuning the AlexNet model. It consists of 3 convolutional layers and 3 fully connected layers.

TABLE III. CONFUSION MATRIX OF IMAGE PATCH CLASSIFICATION FOR 14 CAMERA MODELS FROM TABLE I

Accuracy	True Model														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
Identified Model	1	96.25	1.83	0.11	0.04	0.04	0.60	0	0.02	0.06	0.06	0.86	0.11	0	0
	2	2.27	96.68	0.08	0	0.03	0.24	0.08	0	0.03	0.25	0.27	0.03	0	0.02
	3	0.01	0	99.48	0	0	0.42	0.05	0	0	0	0	0.02	0	0.01
	4	0	0	0	98.87	0	0.59	0	0.34	0.08	0	0.04	0.08	0	0
	5	0	0.06	0	0.09	99.65	0.06	0	0	0	0.09	0.06	0.00	0	0
	6	0.06	0.01	0.04	0	0.03	99.74	0.02	0.04	0.01	0.02	0	0	0.01	0.01
	7	0.01	0	0.23	0	0.01	0.10	99.50	0.05	0	0.01	0.05	0.04	0.01	0
	8	0	0	0	0.02	0	0	0.02	99.88	0	0.02	0.02	0.02	0	0
	9	0	0.02	0	0	0.04	0	0.00	0.16	99.75	0	0.02	0	0.02	0
	10	0.18	0.71	0	0	0.04	0.18	0.09	0	0	98.26	0.45	0.09	0	0
	11	0.26	0.03	0	0.03	0	0.03	0.13	0	0	0.05	99.44	0	0	0.03
	12	0.17	0.04	0.02	0	0.06	0.28	0.11	0.02	0.04	0.15	0	99.09	0	0
	13	0.05	0.07	0.04	0.04	0.05	0.05	0	0.02	0.09	0.02	0	0.02	83.85	15.71
	14	0.07	0.02	0	0	0.05	0.02	0.02	0	0.05	0	0	0.05	19.24	80.48

TABLE IV. MODEL CLASSIFICATION ACCURACY OF IMAGES PATCHES

	Method	HFC	HFC+FFT	WRN	WRN+FFT
		[17]	ours	[17]	ours
12	AlexNet	100%	100%	100%	100%
	GoogleNet	99.81%	100%	100%	100%
	Tuama's Net	100%	100%	99.81%	100%
14	AlexNet	97.80%	99.15%	98.30%	98.65%
	GoogleNet	97.97%	99.32%	98.65%	98.81%
	Tuama's Net	97.29%	98.81%	96.79%	98.14%

The experiments were run on the single Nvidia GPU card of type GeForce GTX 1080Ti. We used NVIDIA DIGITS training system and Caffe as deep learning frameworks. In the following, we report details about each experiment.

A. Camera Model Classification for Patches

In this experiment, we compare the method proposed in [17] which trains a CNN model in a spatial domain with the proposed approach using a partial data set from the Dresden database (see Table I). For each CNN model, we randomly selected 15% of the images for the test data set, 15% for the validation data set and 70% for the training data set.

To demonstrate that the proposed approach using the frequency domain feature of image noise is effective, we compare four types of CNN inputs. The first two inputs are *HFC* and *WRN* used in [17]. Another two inputs are the magnitudes of the Fourier Spectrum of *HFC* and *WRN*, which are referred to as *HFC+FFT* and *WRN+FFT*, respectively. AlexNet, GoogleNet, and Tuama's Net are trained with each type of input.

First, we use the first 12 camera models given in Table I. Then, the experiment is re-performed on all the 14 camera models of Table I. The classification accuracy of image patches for the test data are summarized in Table II. It can be seen from the numerical results that *HFC+FFT* and *WRN+FFT* outperform *HFC* and *WRN*, respectively. *HFC+FFT* achieves the highest accuracy for all CNN structures. In the experiments with 14 camera models, *HFC+FFT* achieves 96.75%, 97.35%, and 96.48% as total classification accuracy for AlexNet, GoogleNet, and Tuama's Net, respectively. The accuracy for *HFC+FFT* is 1.79% higher than that of *HFC* on average. GoogleNet with *HFC+FFT* achieves the highest accuracy, which gains 1.84% improvement compared with GoogleNet with *HFC* proposed in [17]. In addition, Tuama's Net with *HFC+FFT* outperforms GoogleNet with *HFC* for 14-model classification even though Tuama's Net is smaller than

GoogleNet. These results show that it is more appropriate to extract features in the frequency domain than in the spatial domain for camera model classification.

The confusion matrix for AlexNet using *HFC+FFT* is illustrated in Table III. In the confusion matrix, each row of the matrix represents the predictive label of the camera model, and each column represents the true model. The value along the main diagonal is the probability that the label is correctly classified.

B. Camera Model Classification for Full Images

We validate the model classification performance for full images. We use the same dataset and image preprocessing method as Section IV.A. The difference is that a majority voting is used for patches of full images as described in Section III. The classification accuracy of 12 models and 14 models are shown in Table IV.

As can be seen from Table IV, our proposed method using *HFC+FFT* or *WRN+FFT* achieves 100% accuracy for all CNN structures in the classification of 12 camera models. Moreover, *HFC+FFT* achieves the highest accuracy for all CNN structures in the classification of 14 camera models. The accuracy for *HFC+FFT* is 1.41% higher than that of *HFC* on average. As in patch classification, GoogleNet with *HFC+FFT* achieves the highest accuracy, which gains 1.35% improvement compared with GoogleNet with *HFC*. Our approach is effective for not only patch classification but also image classification.

The confusion matrix for AlexNet using *HFC+FFT* is illustrated in Table V. We can see that the classification accuracy per model is 100% except for *SonyDSC-H50* (13) and *SonyDSC-W170* (14). Moreover, even for these two models, the accuracy is over 90%, which is significantly improved compared to the accuracy for patch classification. This is because the characteristics of some patches are not prominent, which leads to classification errors to some extent. When we use the majority voting for image classification, we can reduce errors caused by the above reasons.

C. Camera Brand Classification

We also compare the proposed approach with that proposed in camera brand classification [14], using the Dresden image database, where 10507 images, taken with 10 camera brands. We use 13669 images from 22 camera instances from the same 10 camera brands. We show the dataset information in Table VI.

TABLE V. CONFUSION MATRIX OF IMAGE CLASSIFICATION FOR 14 CAMERA MODELS FROM TABLE I

Accuracy		True Model													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Identified Model	1	100	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	100	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	100	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	100	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	100	0	0	0	0	0	0	0	0	0
	6	0	0	0	0	0	100	0	0	0	0	0	0	0	0
	7	0	0	0	0	0	0	100	0	0	0	0	0	0	0
	8	0	0	0	0	0	0	0	100	0	0	0	0	0	0
	9	0	0	0	0	0	0	0	0	100	0	0	0	0	0
	10	0	0	0	0	0	0	0	0	0	100	0	0	0	0
	11	0	0	0	0	0	0	0	0	0	0	100	0	0	0
	12	0	0	0	0	0	0	0	0	0	0	0	100	0	0
	13	0	0	0	0	0	0	0	0	0	0	0	0	93.02	6.97
	14	0	0	0	0	0	0	0	0	0	0	0	0	9.67	90.32

TABLE VI. EXPERIMENTAL DATASET FOR CAMERA BRAND CLASSIFICATION

	Brand	Model	#images in our dataset	#images in [14]				
1	AgfaPhoto	DC-504	169	1357	1000			
2		DC-733s	281					
3		DC-830i	363					
4		Sensor505-x	172					
5		Sensor530s	372					
6	Canon	Ixus55	224	979	979			
7		Ixus70	567					
8		PowerShotA640	188					
9	Casio	EX-Z150	925	925	925			
10	Fujifilm	FinePixJ50	630	630	630			
11	Kodak	M1063	2391	2391	1000			
12		CoolPix710	925					
13		D200	752					
14		D70	369					
15	Nikon	D70s	367	2413	1000			
16		Olympus	M1050SW			1040	1040	1040
17		Panasonic	DMC-FZ50			931	931	931
18	Samsung	L74wide	687	1332	1331			
19		NV15	645					
20	Sony	DSC-H50	541	1671	1671			
21		DSC-W170	725					
22		DSC-T77	405					

In [14], they randomly divide the dataset into 10 parts, then use 9 parts for training and the rest for testing, and this process is repeated 10 times for different parts. The camera brand classification accuracy achieves 99.1% using the method they proposed.

In our experiment, we also randomly select 90% of the images as the training dataset and 10% as the test dataset. From the experimental results for patch/image classification, we can see that *HFC+FFT* is the best. Therefore, in this experiment, we use *HFC+FFT*. As in the previous

experiments, we trained AlexNet, GoogleNet, and Tuama's net to verify that our method is suitable for various neural networks. Table VII shows the confusion matrix of the test accuracy based on AlexNet. Our method achieved 100% accuracy for each camera model except *Nikon D70* (14) and *Nikon D70s* (15). The model numbers of *Nikon D70* and *Nikon D70s* are similar. Therefore, they may have an image sensor and lens of the same or similar design.

The total classification accuracy of 10 camera brands is summarized in Table VIII for patches and images. All models achieve about 100% accuracy for the camera brand classification¹. It is 0.9% higher than the accuracy reported in [14]. CNN models trained in a frequency domain have high potential to achieve more reliable brand classification compared with that of [14].

V. CONCLUSION

In this paper, we proposed a CNN-based model classification approach that uses frequency domain features of the extracted image noise as input. We evaluated the efficiency of using the frequency domain features, for the dataset from Table I. Our method effectively classified the camera model (see Table II). The results demonstrate that features in a frequency domain are suitable for CNN-based model classification. The full images were classified using majority voting of image patches, and the average accuracy achieved 100% for 12 models and 99.32% for 14 models (see Table IV). Moreover, we achieved 100% accuracy for 10-brand classification. Our proposed approach is effective not only for model classification but also for brand

TABLE VII. CONFUSION MATRIX OF IMAGE CLASSIFICATION ACCURACY FOR 22 CAMERA MODELS FROM TABLE VI

Accuracy	True Model																					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Identified Model	1	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	6	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	7	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
	9	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0
	10	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0
	11	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0
	12	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0
	13	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0
	14	0	0	0	0	0	0	0	0	0	0	0	0	0	48.65	51.35	0	0	0	0	0	0
	15	0	0	0	0	0	0	0	0	0	0	0	0	0	70.27	29.73	0	0	0	0	0	0
	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0
	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0
	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0
	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE VIII. BLAND CLASSIFICATION ACCURACY USING HFC+FFT FOR 10 BRANDS OF TABLE VI

Method	Patch classification	Image classification
AlexNet	99.82%	100%
GoogleNet	99.85%	100%
Tuama's Net	99.77%	100%

¹ This accuracy is rounded off to one decimal place.

classification.

We ignored phase information and used only magnitude information from the Fourier spectrum. From the experimental results, we can see that the magnitude has significant information for camera model classification. It is still an open question how to best extract valuable information from the phase if it exists.

ACKNOWLEDGMENT

This work was partially supported by Nature Science Foundation of China with Grant No. U1609216.

REFERENCES

- [1] H. T. Sencar, and N. Memon, "Digital Image Forensics: There is More to a Picture than Meets the Eye." New York, NY, USA: Springer, 2012.
- [2] P. Korus, "Digital image integrity—A survey of protection and verification techniques," *Digit. Signal Proc.*, vol. 71, pp. 1–26, Dec. 2017.
- [3] M. Kharrazi, H. T. Sencar, and N. Memon, "Blind source camera identification," in *Proc. IEEE Int. Conf. Image Proc.*, 2004, vol. 1, pp. 709–712.
- [4] D. Cozzolino, D. Gragnaniello, and L. Verdoliva, "Image forgery localization through the fusion of camera-based, feature-based and pixel-based techniques," in *Proc. IEEE Int. Conf. Image Proc.*, 2014, pp. 5302–5306.
- [5] L. Bondi, L. Baroffio, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "First steps toward camera model identification with convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 259–263, Mar. 2017.
- [6] C.-T. Li, "Source camera identification using enhanced sensor pattern noise," in *Proc. 16th IEEE Int. Conf. Image Process.*, pp. 7–11, 2009–Aug.
- [7] K. S. Choi, E. Y. Lam, and K. K. Y. Wong, "Source camera identification using footprints from lens aberration," in *Proc. SPIE*, vol. 6069, p. 60690J, Feb. 2006.
- [8] T. Qiao, F. Retraint, R. Coganne, and T. H. Thai, "Source camera device identification based on raw images," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 3812–3816.
- [9] J. Lukáš, J. Fridrich, M. Goljan, "Determining digital image origin using sensor imperfections," in *Proc. SPIE*, vol. 5685, pp. 249–260, 2005–Apr.
- [10] J. Lukáš, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.
- [11] M. Chen, J. Fridrich, M. Goljan, J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.
- [12] Y. Tomioka, Y. Ito, and H. Kitazawa, "Robust digital camera identification based on pairwise magnitude relations of clustered sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 8, pp. 1986–1995, Dec. 2013.
- [13] C. Chen, and M. C. Stamm, "Camera model identification framework using an ensemble of demosaicing features," in *Proc. IEEE Int. Workshop Inf. Forensics Security (WIFS)*, 2015, pp. 1–6.
- [14] A. Roy, R. S. Chakraborty, U. Sameer, and R. Naskar, "Camera Source Identification Using Discrete Cosine Transform Residue Features and Ensemble Classifier." *IEEE Conf. CVPRW*, 2017.
- [15] Y. Hu, C. Li, C. Zhou, "Selecting forensic features for robust source camera identification," *Int. Comp. Symp.*, Dec. 2010, pp. 506–511.
- [16] B. Wang, Y. Guo, X. Kong, "Source Camera Identification Forensics Based on Wavelet Features," in *5th Int. Conf. on IHH-MSP*, 2009, pp. 702–705.
- [17] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification with the use of deep convolutional neural networks," in *Proc. IEEE Int. Workshop Inf. Forensics Security (WIFS)*, Dec. 2016, pp. 1–6.
- [18] H. Yao, T. Qiao, M. Xu, N. Zheng, "Robust Multi-Classifer for Camera Model Identification Based on Convolution Neural Network." *IEEE Access*, vol. 6, pp. 24973–24982, 2018.
- [19] G. Thomas, S. Pfennig, and M. Kirchner, "Unexpected Artefacts in PRNU-Based Camera Identification: A 'Dresden Image Database' Case-Study," in *Proc. Multimedia Sec.*, pp. 109–114, 2012.
- [20] M. K. Mihcak, "Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Mar. 1999.
- [21] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," in *Proc. SPIE*, vol. 9409, pp. 94090J–94090J–10, 2015.
- [22] T. Tang, and P. Hu, "Quantitative standard of promotion strategy and analysis on the influence of consumer purchase behavior," *Cluster. Comput.*, 2018.
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. in NIPS 25. Curran Associates Inc.*, pp. 1097–1105, 2012.
- [24] T. Gole, and R. Böhme, "The Dresden Image Database for Benchmarking Digital Image Forensics," in *Proc. of ACM Symp. Appl. Comput.*, vol. 2, pp. 1585–1591, 2010.
- [25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR)*, Boston, USA, pp. 1–9, 2015.