On Blind Source Camera Identification

G. M. Farinella[1], M. V. Giuffrida[1], V. Digiacomo[1], and S. Battiato[1]

[1]Image Processing Laboratory,
Dipartimento di Matematica e Informatica,
University of Catania, Italy
{gfarinella, battiato}@dmi.unict.it
{valerio.giuffrida88, vinc.digiacomo}@gmail.com

Abstract. An interesting and challenging problem in digital image forensics is the identification of the device used to acquire an image. Although the source imaging device can be retrieved exploiting the file's header (e.g., EXIF), this information can be easily tampered. This lead to the necessity of blind techniques to infer the acquisition device, by processing the content of a given image. Recent studies are concentrated on exploiting sensor pattern noise, or extracting a signature from the set of pictures. In this paper we compare two popular algorithms for the blind camera identification. The first approach extracts a fingerprint from a training set of images, by exploiting the camera sensor's defects. The second one is based on image features extraction and it assumes that images can be affected by color processing and transformations operated by the camera prior to the storage. For the comparison we used two representative dataset of images acquired, using consumer and mobile cameras respectively. Considering both type of cameras this study is useful to understand whether the theories designed for classic consumer cameras maintain their performances on mobile domain.

Keywords: Blind Source Camera Identification

1 Introduction

Since the increasing use of low cost imaging devices embedded in different consumer products (e.g., digital cameras, smartphones, tablet, etc.), thousands of pictures are shot everyday and most of them are posted on the Internet through social networks. Among the questions, Image Forensics aim to answer the following one during investigation: is the image under consideration generated by the device being claimed to be acquired with? In examining the history of a picture, the identification of the device used for its acquisition is a key ingredient. Indeed, in a court of law, the origin of a particular image may represent a crucial evidence; the validity of this evidence might be compromised by the (reasonable) doubt that the image has not been captured from the claimed device [1]. Figuring out what devices was used to take a particular picture could be so important to overturn the court's decision on a trial. Sometimes, one can be lucky to find EXIF metadata inside an image file and can trivially detect the model of camera used to take a picture [2]. However, EXIFs cannot be used during a trial, because can be easily manipulated. Therefore, particular attention was made by the research community to design algorithms able to infer the camera device using the only available visual information: the input image itself.

Blind source camera identification methods attempt to infer the device using the information extracted from the images. In literature, different solutions were proposed for this purpose. The methods can be grouped in two main categories: sensor's defects based and pipeline based. In building the fingerprint, the first kind of algorithms usually exploits the noise generated by camera sensor, whereas the latter one is based on features extracted from the image. For a survey of the different techniques the reader can refer [1, 3].

We tested two popular source camera identification algorithms, belonging to the aforementioned categories. Specifically, we have considered the camera identification through the sensor noise [4] and the feature based method [5, 6]. The contribution of this study is to test those approaches on images acquired by mobile phone devices. This allows to understand the performances of the involved methods (which were designed for images acquired with consumer digital cameras), perform well in the mobile domain. We aim to prove how those methods are able to solve two source camera identification's scenarios. The former one detects which camera device shot a particular picture, whereas the latter one discriminates among different camera models.

The remainder of this paper is organized as follows: Section 2 discusses the camera identification based on sensor noise, whereas Section 3 introduces the approach based on feature extraction. Section 4 reports experimental settings and the results. Finally, Section 5 concludes the paper.

2 Camera identification based on sensor noise

Any image can contain different kinds of noise, which can be classified how they are generated from. The *shot noise* is a random electronic signal perturbation produced by the integrated circuits. Another noise source are due to faulty pixels (dead or saturated), which alter significantly the RGB value of a cell in the camera sensor. The remaining part of the noise is almost a regular signal and it is imprinted at each camera shot, called *pattern noise*
[7, 8]. And in fact, it is the *pattern noise* the signal we look for to generate the camera fingerprint. Figure 1 shows the two components included in the *patter noise*: *Fixed Pattern Noise* (FPN), and *Photo Response Non-Uniformity* (PRNU). A small amount of the pattern noise is given by the FPN and it is caused by dark currents in the circuit and also depends on exposure and temperature. Most of the pattern noise is due to the *Photo Response Non-Uniformity*, which is given in part by the *Pixel Non-Uniformity* (PNU) noise, and in part by *Low Frequency Defects* (LFD).

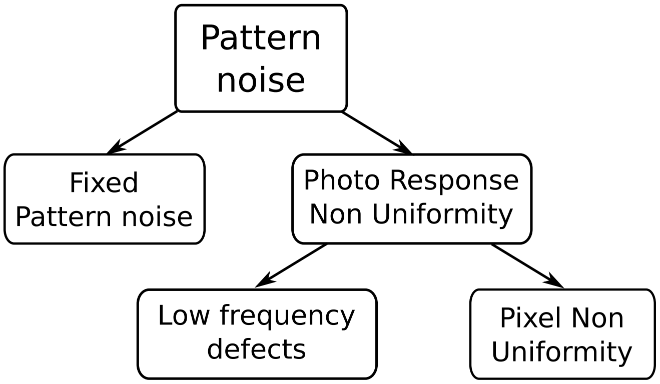

Fig. 1: Pattern noise hierarchy.

Color Features	**IQM features**
Pixel intensity mean value	Minkowsky measures
R-G, R-B, G-B channel correlation	Correlation measures
Pixel neighbor center of mass	Spectral distances
Frequency-domain statistics	

Table 1: List of features used in [12].

To extract the PNU signature component of a specific camera, N pictures have to be considered [4, 9–11]. The residual noise of each image is obtained by the following relation:

$$n_c^{(k)} = p_c^{(k)} - F(p_c^{(k)}) \tag{1}$$

where $p_c^{(k)}$ denotes the k^{th} image acquired with a camera c and $F(\cdot)$ is a denoising filter. Given a camera c, the PNU fingerprint P_c is obtained by averaging all the $n_c^{(k)}$'s. When the fingerprints for each camera P_c, $c = 1 \ldots, C$ are extracted, the identification of an image is done correlating the pattern noise, extracted from the query image, and all of the fingerprints. The correlation is computed as follows:

$$\rho_c(p) = \frac{(n - \overline{n})(P_c - \overline{P_c})}{\|n - \overline{n}\| \|P_c - \overline{P_c}\|} \tag{2}$$

where the $\overline{n}$ and $\overline{P_c}$ denotes mean values respectively of the picture residual noise and fingerprint of the camera c. Refinements of this method employ the *Maximum-Likelihood* approach to estimate the fingerprint [9], or specifically designed correlation method [10] to improve the classification performances.

Model	**Symb.**	**# Devices**	**# Pictures**
Canon Ixus 70	I70	3	567
Casio EX-Z150	EX	3	555
Kodak M1063	M	3	603
Nikon Coolpix S710	S710	3	566
Olympus μ1050SW	μ	3	631
Pratika DCZ 5.9	DCZ	3	614
Rollei RCP-7325XS	RCP	3	589
Samsung NV15	NV	3	645
	TOTAL	**24**	**4770**

Table 2: Selected camera model from the *Dresden database*.

3 Camera identification based on features

Another family of source camera identification methods is based on the extraction of a set of features to build up a descriptor for the specific camera. The discrimination is performed by analyzing differences on model-dependent characteristics. The basic idea resides in looking at differences in the *Image Generation Pipeline* (IGP), where the image is processes by different algorithms (e.g., demosaicing, white balancing, color correction, etc.) [13]. Since many types of camera use different algorithms/parameters in the IGP, features on the image can be properly extracted to set up a descriptor for the camera model.

The employed method considers a set of features [5, 6, 12, 14, 15], which can be grouped into two families, as it is shown in Table 1: *features based on color* and *image quality metrics (IQM) features*. The set of features can be improved by adding new measurement for similarity between two images [16]. Other features include the dependencies between the average values of the colour channels [16]; additional features characterising white point correction have been also included considering twelve new features belonging to two groups [5, 6]: *white balancing* and *wavelets measurement*. Classification is performed by training a *Support Vector Machine* [17], using the radial basis kernel function.

4 Experimental setup and results

The methods discussed in Section 2 and Section 3 have been considered and tested on two different datasets: the well-known *Dresden database* [18] and a cellular mobile phone dataset [16]. The *Dresden database* collects more than 14,000 pictures of 47 different scenes, using 73 consumer camera devices of 25 different models. The second dataset [16] collects more than 3,000 pictures, using 17 mobile phones of 15 different models. Table 2 and Table 3 show in detail the two datasets respectively.

Model	Symb.	# Pictures
LG 5600	L	200
Motorola V3	V3	200
Motorola V500	V5	231
Nokia 5140i	N5	200
Nokia 6230i	N62	216
Nokia 6600 (1)	A66	235
Nokia 6600 (2)	B66	200
Nokia 7270	N7	219
Samsung D500	S5	200
Samsung D600	S6	200
Samsung E720	S7	200
Sony K700i (1)	AK7	275
Sony K700i (2)	BK7	200
Sony K750	K75	204
Sony P800	P8	209
Sony P910	P9	200
PalmOne Treo	PO	200
	TOTAL	**3589**

Table 3: List of mobile phones in [16] dataset.

4.1 First test on Dresden database

For this test we considered the problem of recognizing the camera model. Hence, we are not interested to detect the exactly device, but its model. We selected a subset of the *Dresden dataset* such that the same scene was taken from all the devices. With this setup, testing dataset contains 4,470 images coming from 24 camera devices of 8 different models (see Table 2). We performed the test, using picture from 2 devices as training set and the last one as validation set. Final results are obtained by averaging the outcomes of each test. In Table 4 we report the results using the sensor noise method, whereas in Table 5 we report the results for the feature based method.

Results show that the algorithm based on sensor noise do not perform well in this case. This is due to the fact that fingerprints cannot be generalized for different devices, even if they belong to the same model. Specifically, an average accuracy of 45.37% (Table 4) was obtained for the method based on sensor noise, whereas the feature based method obtained an average accuracy of 79.19% (Table 5).

4.2 Second test on Dresden database

As a follow-up of the previous test, we selected randomly 2/3 of the pictures for each of camera data models, independently from the acquisition device. With this

	Inferred as							
	I70	EX	M	S710	μ	DCZ	RCP	NV
I70	14.65	10.21	9.89	**20.16**	11.07	7.58	12.51	13.92
EX	1.08	**86.88**	1.79	2.52	2.70	1.43	1.81	1.79
M	8.45	8.79	**28.18**	9.44	11.77	10.82	10.80	11.76
S710	3.37	2.32	2.69	**79.69**	2.47	4.63	2.147	2.69
μ	6.70	9.10	9.76	10.84	**40.22**	8.13	7.60	7.67
DCZ	7.48	9.16	12.72	17.27	11.06	**22.78**	10.10	9.43
RCP	2.72	3.56	3.05	5.61	5.27	1.70	**73.67**	4.41
NV	9.90	9.79	13.00	**18.30**	10.40	9.31	12.39	16.91

Table 4: Confusion matrix from the first test on Dresden database (cf. Section 4.1), for the algorithm based on sensor noise.

	Inferred as							
	I70	EX	M	S710	μ	DCZ	RCP	NV
I70	**82.25**	7.15	0.18	0	1.24	2.13	0	7.06
EX	14.62	**80.93**	0	0.53	2.14	0.18	0.36	1.25
M	0	0	**99.83**	0	0.17	0	0	0
S710	0.34	0.53	0	**94.20**	4.08	0	0	0.85
μ	0.96	1.80	1.28	1.46	**66.28**	0	3.11	25.12
DCZ	2.93	0.64	0	0	0	**78.71**	17.06	0.65
RCP	0	1.68	0	0	10.27	19.70	**67.34**	1.01
NV	4.63	1.69	0	1.85	17.29	0.61	9.97	**63.96**

Table 5: Confusion matrix from the first test on Dresden database (cf. Section 4.1), for the algorithm based on features extraction.

setup, we obtained an average accuracy of 98.59% (Table 6) for the method based on sensor noise. Instead, feature based method we obtained an average accuracy of 55.87%. Considering sensor noise from many pictures, selected randomly from three different devices of the same model, makes the resulting fingerprint more sensitive for the blind camera identification. Nevertheless, the drop in accuracy for features based methods is due to cameras parameters variability. Even if we tested among the same camera model, each device had different photometric setting (focus, white balancing, and so forth), which makes the classification task harder. To confirm our theory, we performed another test with feature based algorithm, using leave-one-out cross validation [19]. In this case, accuracy of the feature based method was 81.97%, as it is shown in Table 8.

4.3 Third test on Dresden database

The third test is devoted to assess the performances of the identification of a specific camera device. Differently than before, we tested how those algorithms perform in identifying the device that took a specific picture. In this case, the

				Inferred as				
	I70	EX	M	S710	μ	DCZ	RCP	NV
I70	**100**	0	0	0	0	0	0	0
EX	0	**99.86**	0	0	0.14	0	0	0
M	0.28	0	**99.17**	0.28	0	0	0	0.28
S710	0	0	0	**100**	0	0	0	0
μ	1.03	1.16	0.90	1.55	**90.48**	1.67	1.54	1.67
DCZ	0	0	0	0.13	0	**99.73**	0	0.13
RCP	0	0	0	0	0	0	**100**	0
NV	0	0.26	0	0.26	0	0	0	**99.48**

Table 6: Results for the second test, using sensor noise based algorithm.

				Inferred as				
	I70	EX	M	S710	μ	DCZ	RCP	NV
I70	**54.90**	0.96 0	0	0.55	14.80	2.73	21.55	4.51
EX	13.34	**51.31**	0	0.84	7.78	4.33	21.97	0.42
M	1.11	7.21	**90.57**	0.28	0.69	0	0.14	0
S710	0	2.66	0	**81.32**	3.11	0	12.16	0.75
μ	0.39	0	1.81	4.65	**58.25**	0.51	31.31	3.08
DCZ	17.57	4.78	0	0	9.07	**47.02**	11.48	10.08
RCP	0.40	0.71	0	0	**54.16**	0.42	34.12	10.18
NV	2.42	0.26	0	2.77	**35.07**	0.65	29.31	29.52

Table 7: Results for the second test, using features extraction algorithm.

Dresden dataset was randomly sampled, in such a way 29 scenes were used as training set and the remaining 18 as validation set. This experiment was repeated three times as before, and we reported the average performances. In Table 9 we shows that sensor based method outperforms the feature one, with an overall accuracy of about 99%.

4.4 Test on mobile phones database

The tests presented so far have been performed considering a dataset composed by images acquired with consumer digital camera. However, nowadays most of the images are acquired with mobile phones and are becoming more and more recurring in digital investigation. We used the dataset in [16], that is listed in Table 3. Because of the data we have, we considered the device detection task, as it was done in Section 4.3 for the *Dresden dataset*. Devices belonging to the same model (e.g., Nokia 6600) are treated as different cameras.

We perfomed three experiments, splitting each time the dataset in three groups. Table 10 shows the results obtained from the sensor noise based method, whereas Table 11 we report the results obtained from feature based approach.

				Inferred as				
	I70	EX	M	S710	μ	DCZ	RCP	NV
I70	**78.99**	1.91	0	0	0.70	16.68	0.77	0.95
EX	6.28	**81.79**	0.11	3.14	0.45	5.84	1.90	0.47
M	0	0.99	**99.01**	0	0	0	0	0
S710	0	1.07	0	**96.92**	0.47	0	0.47	1.06
μ	7.06	0.71	2.01	0	**40.94**	0	10.57	38.70
DCZ	0.57	0.11	0	0	0.51	**91.69**	6.21	0.91
RCP	0	0	0	0	0.83	0.35	**98.82**	0
NV	5.58	0.83	0	0.19	18.13	1.55	6.11	**67.62**

Table 8: Results for the second test for the algorithm based on features extraction (Section 3) by using leave-one-out cross-validation.

	Device 0		**Device 1**		**Device 2**	
Models	*Sensor Noise*	*Features*	*Sensor Noise*	*Features*	*Sensor Noise*	*Features*
I70	**100%**	60.73%	**100%**	41.45%	**100%**	16.86%
EX	**99.12%**	30.51%	**99.58%**	33.10%	**99.18%**	45.57%
M	**99.16%**	30.91%	**98.27%**	59.18%	**99.60%**	28.69%
S710	**100%**	27.08%	**100%**	61.30%	**100%**	24.11%
μ	**93.03%**	9.98%	**95.34%**	54.04%	**93.43 %**	41.29%
DCZ	**100%**	52.27%	**100%**	37.25%	**100%**	23.30%
RCP	**100%**	51.86%	**100%**	20.35%	**100%**	36.22%
NV	**100%**	33.41%	**100%**	35.99%	**100%**	33.72%

Table 9: Third test results.

Experimental results show that sensor noise method outperforms the feature based on. Cameras found in a mobile phone has a worse optic than the cameras used in the *Dresden database*. This means that the amount of noise released by the devices in the pictures is higher, resulting in a stronger fingerprint able to discriminate better among different devices.

5 Conclusions

In this paper we have presented a comparative study of two popular methods for source camera identification: sensor noise extraction method [4] and features extraction method [6, 18]. We tested the performances of those two approaches with two dataset: the *Dresden dataset* [20] and the mobile phone dataset proposed by [16]. Our tests show the sensor noise approach outperforms the feature one. The reason in that the pattern noise imprinted on the pictures is more discriminative than the extracted features. Our finding are more evident in the mobile phones dataset. Optic in these devices are worse than the devices used to make the *Dresden Dataset*, resulting in a more evident and sensible fingerprint for the classification task. The only case the features based method outperformed

	Inferred as																
	L	V3	V5	N5	N62	A66	B66	N7	S5	S6	S7	AK7	BK7	K75	P8	P9	PO
L	**100**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V3	0	**99.67**	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0
V5	0.87	2.33	**84.30**	0.58	0.58	1.17	1.45	0.29	0.29	0.87	2.33	0	1.17	0	0.58	0.58	2.61
N5	0	0	0	**100**	0	0	0	0	0	0	0	0	0	0	0	0	0
N62	0	0	0	0	**100**	0	0	0	0	0	0	0	0	0	0	0	0
A66	0	0.28	0.57	0.28	0	**98.01**	0	0	0	0	0.29	0	0.57	0	0	0	0
B66	0	0	0	0	0	0	**99.67**	0	0	0	0	0	0	0	0	0.33	0
N7	0	0	0	0	0	0.31	0.31	**99.38**	0	0	0	0	0	0	0	0	0
S5	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0	0	0	0
S6	0	0	0	0	0	0	0	0	0	**99.33**	0	0.66	0	0	0	0	0
S7	0	0	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0	0
AK7	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0
BK7	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0	0	0
K75	1.96	0	0.33	0.65	0	0.33	0.33	1.30	0.33	0.65	0.98	0	0	**91.19**	0.65	0.65	0.65
P8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0	0
P9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**100**	0
PO	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0	**99.67**

Table 10: Confusion matrix for the algorithm based on sensor noise, tested with the mobile phone dataset [16].

	Inferred as																
	L	V3	V5	N5	N62	A66	B66	N7	S5	S6	S7	AK7	BK7	K75	P8	P9	PO
L	**72.34**	0	0.33	0	0	0	27.33	0	0	0	0	0	0	0	0	0	0
V3	0	**66.67**	31.33	0	0	1	1	0	0	0	0	0	0	0	0	0	0
V5	0	18.02	**80.82**	0	0	1.16	0	0	0	0	0	0	0	0	0	0	0
N5	21.67	1	0.33	**22.67**	21	9.67	18	1	0.33	0.33	2	0	0	2	0	0	0
N62	0.93	9.24	5.87	0	**73.78**	2.16	7.71	0	0.31	0	0	0	0	0	0	0	0
A66	0	15.10	7.98	0	0	**72.08**	4.84	0	0	0	0	0	0	0	0	0	0
B66	0	4.33	0.67	0	0	11.67	**83.33**	0	0	0	0	0	0	0	0	0	0
N7	1.26	6.92	2.83	3.77	11.95	4.09	0	**65.10**	0.63	0	0.31	0	0	0.31	1.57	0.63	0.63
S5	0.67	19.67	0.67	0.33	20.67	0.33	0.33	3.67	**49.66**	0	2.33	0	0	0.33	0.67	0	0.67
S6	0	0	0	0	0.34	0	0.67	0	0	**99**	0	0	0	0	0	0	0
S7	3.33	4.67	1.33	0.33	**42.67**	0.33	2	0	5	0	40.33	0	0	0	0	0	0
AK7	0	0	0	0	0	0.49	0	0	0	0	0.49	**97.81**	1.21	0	0	0	0
BK7	1	0.67	0	1	0	2	0	0	0	0	0	**80**	15.33	0	0	0	0
K75	1.31	3.92	0	7.52	7.19	4.25	3.92	2.29	0.98	0.98	0.98	0	0	**66.33**	0	0.33	0
P8	0	27.24	8.01	0	0.32	16.67	7.05	0	0	0	0.64	0	0	0	**30.45**	4.49	5.13
P9	0	4.67	1.67	0	0.33	10	2	0	0	0	0.33	0	0	0	20.67	**51**	9.33
PO	0	**43**	9	0	0	16.34	1	0	0	0	0	0	0	0	0	0.33	30.33

Table 11: Confusion matrix for the algorithm based on features, tested with the mobile phone dataset [16].

the sensor noise one is discussed in Section 4.1, where the camera model has to be recognized. In the performed experiments, feature based methods are not able to classify the devices correctly, when the images taken for the same camera devices are grouped altogether. Morever, for the device identification problem, the feature methods was not able to provide a reliable response. Future works could focus on combining the methods to improve the results, as well as in extending these methods in order to perform blind source camera identification in video domain.

Acknowledgments

We would like to thank the authors of [12] for providing us their original implementation for feature extraction, as well as the authors of [16] who allowed us to use their database to perform the experimental comparison on mobile images.

References

1. Judith Alice Redi, Wiem Taktak, and Jean-Luc Dugelay. Digital image forensics: a booklet for beginners. *Multimedia Tools and Applications*, 51(1):133–162, 2010.
2. E. Kee, M.K. Johnson, and H. Farid. Digital image authentication from JPEG headers. *Information Forensics and Security, IEEE Transactions on*, 6(3):1066–1075, 2011.
3. Alessandro Piva. An overview on image forensics. *ISRN Signal Processing, Article ID 496701*, 2013:22 pages, 2013.
4. J. Lukáš, J. Fridrich, and M. Goljan. Digital camera identification from sensor pattern noise. *Information Forensics and Security, IEEE Transactions on*, 1(2):205–214, June 2006.
5. Thomas Gloe, Karsten Borowka, and Antje Winkler. Information hiding. chapter Feature-Based Camera Model Identification Works in Practice, pages 262–276. Springer-Verlag, Berlin, Heidelberg, 2009.
6. Thomas Gloe. Feature-based forensic camera model identification. *Transactions on Data Hiding and Multimedia Security*, 8:42–62, 2012.
7. Gerald C Holst. *CCD arrays, cameras, and displays, 2nd edn.* JCD Publishing & SPIE Press, USA, 1998.
8. James R. Janesick. *Scientic Charge-Coupled Devices.* SPIE Press, USA, 2001.
9. Mo Chen, Jessica Fridrich, and Miroslav Goljan. Digital imaging sensor identification (further study). In *In Security, Steganography, and Watermarking of Multimedia Contents IX. Edited by Delp, Edward J., III; Wong, Ping Wah. Proceedings of the SPIE, Volume 6505*, 2007.
10. Miroslav Goljan, Jessica Fridrich, and Tom Filler. Large scale test of sensor fingerprint camera identification. In *Proc. SPIE, Electronic Imaging, Security and Forensics of Multimedia Contents XI*, pages 18–22.
11. Alan J Cooper. Improved photo response non-uniformity (PRNU) based source camera identification. *Forensic Science International*, 226(13):132–141, 2013.
12. M. Kharrazi, H.T. Sencar, and N. Memon. Blind source camera identification. In *Image Processing, 2004. ICIP '04. 2004 International Conference on*, volume 1, pages 709–712 Vol. 1, 2004.

13. S. Battiato, A.R. Bruna, G. Messina, and G. Puglisi. *Image Processing for Embedded Devices*. Bentham Science Publisher, 2010.
14. I. Avcibas, N. Memon, and B. Sankur. Steganalysis using image quality metrics. *Transaction on Image Processing*, 12(2):221–229, 2003.
15. Avcibas Ismail, Sankur Bülent, and Sayood Khalid. Statistical evaluation of image quality measures. *Journal of Electronic Imaging*, 12(2):221–229, 2003.
16. O. Celiktutan, B. Sankur, and I. Avcibas. Blind identification of source cell-phone model. *Information Forensics and Security, IEEE Transactions on*, 3(3):553–566, 2008.
17. Nello Cristianini and John Shawe-Taylor. *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, New York, NY, USA, 2000.
18. Thomas Gloe and Rainer Böhme. The Dresden Image Database for benchmarking digital image forensics. In *Proceedings of the 25th Symposium On Applied Computing (ACM SAC 2010)*, volume 2, pages 1585–1591, 2010.
19. Andrew R. Webb. *Statistical Pattern Recognition (2nd Edition)*. John Wiley & Sons, Ltd., November 2002.
20. Thomas Gloe and Rainer Böhme. The dresden image database for benchmarking digital image forensics. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1584–1590, 2010.