Digital forensics: a fast algorithm for a digital sensor identification

Jarosław Bernacki and Rafał Scherer

ABSTRACT

We consider the identification of imaging devices by analysing images they produce. The problem is studied in the literature, yet the existing solutions are rather computationally demanding. We propose a high-speed algorithm for the identification of imaging devices. The aim is to provide additional security by identification of legitimate imaging devices or an identification for forensics. The experimental evaluation confirms efficient identification of devices models and brands by the proposed algorithm, compared with the state-of-the-art method. Moreover, our algorithm is approximately two orders of magnitude faster, which is very important in resource-constrained IoT ecosystems or very large databases.

1. Introduction

Distinguishing the model of the sensor that was used to generate a digital image is one of the tasks of digital forensics. In many cases, it is important to determine the source of a digital image, for example, for criminal or forensic investigation. Usually, the goal is to identify a digital camera or flatbed scanner. Digital camera identification is understood as the identification of the sensor that produced an image. Similarly, recognition of the flatbed scanner relies on the identification of the scanner sensor that captures scanned materials (Khanna et al., 2007b). The aim to provide additional security can be understood in the context of an authentication protocol. For example, the user wants to log into a certain system using a dedicated smartphone to take a photo of her or his iris. Then the system checks the iris and camera’s fingerprint in the base. If both things match, the user is considered legitimate, and the system grants access; otherwise, user login fails. For such a purpose, there is a need for a fast and accurate camera identification algorithm that checks the imaging device origin in real-time. Due to its highly connected nature, Internet of Things technology is vulnerable to many threats and breaches. Adequate security measures are crucial to ensuring computer network safety as IoT devices can be used to penetrate or attack their host computer networks. As the IoT idea is

CONTACT Jarosław Bernacki jaroslaw.bernacki@pcz.pl Department of Intelligent Computer Systems, Częstochowa University of Technology, Częstochowa, Poland

© 2022 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.
relatively new, and the market is very dynamic; thus, many devices are out without serious
security protection or with gaps. Moreover, many devices are low-end ones, not allowing
to implement of enough security. Similarly, legacy devices were sometimes not designed
to work online and can be insecure. IoT security breaches and attacks can hurt any indus-
try or even smart homes. An example of a fragile ecosystem can be healthcare. In the
paper, we propose a method of securing imaging devices by extracting features respon-
sible for the manufacturer and type of the imaging system. The method can be used to
detect external, not authorized imaging devices in the IoT network.

Recently, many algorithms for sensor identification have been proposed. A state-of-the-art algorithm for digital camera identification was proposed by Lukás et al. (2006). An analogous algorithm for scanner identification is described in Khanna et al. (2007b). Both approaches are based on searching for the so-called sensor pattern noise by denoi-
sing source material with a wavelet-based denoising filter. The main assumption is that
source materials are represented according to the RGB model by three data arrays that
define red, green and blue colour components (channels) for each individual pixel. It is
usually suggested to denoise all the channels. Algorithms based on wavelet denoising
are efficient in classification tasks; however, they are very time-consuming. The typical
time for processing an image of 6000 × 4000 pixel resolution by algorithms presented
in Khanna et al. (2007b) and Lukás et al. (2006) take at average 2–3 min (http://dde.
binghamton.edu/download/). According to Lukás et al. (2006), the optimal number of
processed images for representative calculation of the fingerprint is 45. It is then
clearly visible that such algorithms are inefficient in terms of processing time, which
makes them impractical for dense traffic in IoT networks or large datasets. Therefore,
these facts motivated us to propose faster solutions.

In this paper, we propose a very fast imaging device identification method for
resource-constrained IoT devices and high network traffic. We apply the denoising filter
only to one colour channel of an RGB image. Moreover, we process only fragments of
images of size 512 × 512 pixels instead of the whole image. As a result, a significant
increase in the speed of source materials processing time is observed; however, the pro-
posed algorithm is slightly less accurate than state-of-the-art deep learning methods. The
results are presented with the use of ROC (Receiver operating characteristic) curves and
confusion matrices. The classification performance is also discussed in terms of the stat-
istical analysis, which confirmed the reliability of the proposed algorithm.

The rest of the paper is organized as follows. Section 2 describes related works and
recalls a state-of-the-art algorithm. In Section 3, the proposed method is described.
Section 4 shows experimental results of the evaluation. Section 5 presents the statistical
analysis of obtained results. Section 6 concludes this work.

2. Related work

One of the most common algorithms in camera forensics is Lukás et al.’s approach, pre-
sented in Lukás et al. (2006). This algorithm is often used for digital camera identification
and even to distinguish different copies of the same camera. It gives precise classification
results but can be time-consuming, especially for high-resolution images. Note also that
the considered algorithm assumes that images should not be processed as fragments,
but in their full resolution, which also may have a negative impact on the performance.
We will recall this algorithm in detail in the next section. The idea of this algorithm has been further investigated and extended in many works, for example in Kang et al. (2012) and Jiang et al. (2016). There have been considered other approaches for camera identification, like entropy and image quality measures (Agarwal et al., 2016), analysis of camera’s white balance algorithm (Deng et al., 2011), JPEG compression (Goljan et al., 2016), clustering techniques (Baar et al., 2012; Tomioka & Kitazawa, 2011) or compact representation of the camera’s fingerprint (Li et al., 2018; Valsesia et al., 2015). Recently very popular are approaches based on deep learning (DL) and convolutional neural networks (CNN). A convolutional neural network with fully connected layers was described in Tuama et al. (2016). The convolutional layers are responsible for feature representation, and the fully connected layers were used for classification. The layer of feature representation collects the noise pattern \( \mathbf{N} \) which is extracted from images from the well-known formula \( \mathbf{N} = \mathbf{I} - \mathbf{F}(\mathbf{I}) \) (Fridrich & Goljan, 2011; Galdi et al., 2015, 2016; Kang et al., 2012; Lukás et al., 2006), where \( \mathbf{I} \) denote the input image and \( \mathbf{F} \) is the denoising filter. Noise patterns are treated with 64 kernels of size \( 3 \times 3 \), and the size of produced feature maps is \( 126 \times 126 \). The second layer produces feature maps of size \( 64 \times 64 \). The third layer applies convolutions with 32 kernels of size \( 3 \times 3 \). The Rectified Linear Unit (ReLU) is applied to the output of every convolutional layer as an activation function.

Nvidia GeForce Titan X (12GB) GPU is used as the hardware. Despite that classification accuracy achieves 98 %, time for learning the network takes 5 h and a half only for 12 camera models. Another CNN-based approach was discussed in Rafi et al. (2019), where the DenseNet convolutional network was used for camera identification. The classification accuracy reached 98%. In Tuama et al. (2016) a machine learning classifier was trained on the concatenation of the co-occurrences of colour band noise residuals with features computed with a Markovian model in discrete cosine transform (DCT) domain. The usage of deep models is also considered in Freire-Obregón et al. (2017), Chen et al. (2017), Yang et al. (2019), Bondi et al. (2017) and Yao et al. (2018). However, such approaches are time-consuming, and the classification accuracy is comparable to the previous algorithms.

To the best of our knowledge, the most popular algorithm for flatbed scanner identification is proposed by Khanna et al. (2007b). Further improvements were described by the authors in Khanna et al. (2008a, 2008b, 2009). This approach shows that flatbed scanner identification can be realized in the same spirit as the state-of-the-art algorithm for digital camera identification proposed by Lukás et al. (2006). Another approach for recognizing scanners is presented in Khanna et al. (2008a) (and its journal extension Khanna et al., 2009). The core of the algorithm is similar to Lukás et al. (2006). Image \( \mathbf{I} \) and its denoised version \( \mathbf{F}(\mathbf{I}) \) is taken and the residual \( \mathbf{R} \) is calculated in the following way: \( \mathbf{R} = \mathbf{I} - \mathbf{F}(\mathbf{I}) \). The experimental evaluation proved that for any scanner, matrix \( \mathbf{R} \) is unique for any scanner and therefore can serve as the scanner fingerprint. These approaches use the wavelet denoising filter for image processing. A somewhat similar approach is presented in Gou et al. (2007). In Khanna et al. (2007b) and Khanna et al. (2008b), there are shown different experiments for comparing scanner and camera identification by images. In Lyu and Farid (2005), there is proposed a technique for differentiating between computer-generated and photographic images. The base is the usage of wavelet statistics. Work (Khanna et al., 2007a) presents a method for the classification of images based on their sources. An SVM classifier is trained by the appropriate features of the sensor pattern noise.
Distinguishing if the image was taken by a camera or scanned is performed with an accuracy of 95%. In Gloe et al. (2007), a method for digital camera identification based on spatial noise is applied to flatbed scanner identification. Since scanners and digital cameras use similar technologies, the application of camera-related forensics methods for scanner identification can be successful. Work (Dirik et al., 2009) presents an approach based on the presence of dust and physical scratches over the scanner platen.

3. Imaging sensor identification

In this subsection we recall algorithms for digital imaging sensor identification and propose our MSE-DSI method.

3.1. State-of-the-art algorithms

3.1.1. Lukás / Khanna et al.’s algorithm

The most popular algorithm for digital imaging sensor identification, especially used for digital camera identification, seems to be proposed by Lukás et al. (2006). Although recently deep learning (DL) approach including convolutional neural networks (CNN) is very popular, this algorithm is still widely used for digital forensics purposes. Moreover, many modern approaches using DL like Tuama et al. (2016) are also based on this approach. This algorithm extracts a specific pattern called the Photo-Response Nonuniformity Noise (PRNU), which serves as a unique identification fingerprint. The idea of the algorithm is to extract the noise from the input image \( I \) by using a denoising filter \( F \). Using a wavelet-based denoising filter has been proposed (Goljan, 2008; Jiang et al., 2016; Kang et al., 2012). After denoising, the camera fingerprint is calculated as

\[
N = I - F(I)
\]

following the summary below.

**Input:** Image \( I \) in RGB of size \( M \times N \);

**Output:** Matrix \( N \) of noise residual, size \( M \times N \).

1. Using a wavelet-based denoising Calculate \( K = \frac{I_R + I_G + I_B}{3} \);
2. Denoise all colour channels of the input image \( I_R, I_G, I_B \) with filter \( F \);
3. Calculate mean \( D = \frac{F(I_R) + F(I_G) + F(I_B)}{3} \);
4. Calculate the matrix of noise residual: \( N = K - D \).

where \( I_R, I_G \) and \( I_B \) are matrices of each component of the RGB model of the input \( I \); \( F \) is the denoising filter. Matrix \( I \) comes from a single image of a particular camera. PRNU for each camera is calculated from at least 45 images and then \( N = \frac{N_1 + \cdots + N_{45}}{45} \). Afterwards, a correlation coefficient between the new image \( N_x \) and \( N \) is calculated. If the correlation coefficient exceeds some threshold, it is assumed that \( N_x \) comes from the same camera as \( N \). The efficiency of recognizing the sensor in this approach is very high (True Positive Rate greater than 90%). The algorithm also identifies different cameras of the same model with similar probability.

Khanna et al. (2007b) showed that flatbed scanner identification can be realized in the same spirit as Lukás et al.’s algorithm for camera identification. It also calculates the PRNU of each flatbed scanner based on scans it produced and the classification of a new scan is realized similarly as in Lukás et al.’s algorithm.
Wavelet-based denoising filters are often used for digital forensics. The most common filters used for calculation of the PRNU is sigma (Lukás et al., 2006) and Mihcak filters (Amerini et al., 2009; Mihçak et al., 1999). Here, we recall the formula for Mihcak denoising. This filter uses spatially adaptive statistical modelling of wavelet coefficients. The noisy coefficients $G(i)$ are considered as the addition of the noise-free image $I(i)$ and the noise component $N(i)$. The noise component $N(i)$ is the white Gaussian noise with known variance $\sigma_n^2$. The goal is to retrieve the original image coefficients as well as possible from the noisy observation by using a local Wiener filter described in Equation (1).

$$I'(i) = \frac{\sigma_X^2(i)}{\sigma_X^2(i) + \sigma_n^2} G(i)$$ (1)

### 3.1.2. Convolutional neural network-based methods

Recently camera identification is mostly realized with the use of convolutional neural networks (CNN). We consider two methods: one proposed by Mandelli et al.’s (2020) and the other by Kirchner and Johnson (2020).

Mandelli et al.’s CNN architecture can be described as follows:

1. The first convolutional layer of kernel $3 \times 3$, producing feature maps of size $16 \times 16$ pixels with Leaky ReLU as an activation method and max-pooling of $3 \times 3$;
2. The second convolutional layer of kernel $5 \times 5$, producing feature maps of size $64 \times 64$ pixels with Leaky ReLU as an activation method and max-pooling of $3 \times 3$;
3. The third convolutional layer of kernel $5 \times 5$ producing feature maps of size $64 \times 64$ pixels with Leaky ReLU as an activation method and max-pooling of $3 \times 3$;
4. A pairwise correlation pooling layer;
5. Fully connected layers.

Kirchner & Johnson proposed a CNN with the following structure:

1. 17 layers implementing 64 convolutional filters with $3 \times 3$ kernels;
2. ReLU as an activation method after each layer;
3. No pooling layers.

Both Mandelli and Kirchner & Johnson’ networks can be trained with the PRNU $\mathbf{N}$ calculated in the same manner as in Lukás et al.’s algorithm: $\mathbf{N} = \mathbf{I} - F(\mathbf{I})$, where $\mathbf{I}$ is the input image, and $F$ is the denoising filter.

### 3.2. Proposed approach

The Mean Square Error (MSE) is a quality metric that can be used for assessing the quality of images or videos. We propose the Mean Square Error-Digital Sensor Identification (MSE-DSI) algorithm which uses this metric for a scanner identification. To the best of our knowledge, it is the first attempt to use such an algorithm for scanner identification.
purposes. Calculating MSE-DSI is defined as in Equation (2).

\[ \text{MSE} - \text{DSI} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} [R(i, j) - D(i, j)]^2, \]  

where: \( M, N \) – image resolution (in pixels); \( R \) – pixel intensities of one of colour channels of the original image \( I \); \( D = F(I) \) – denoised a particular colour channel of image \( I \), and \( F \) is the denoising filter. A wavelet-based denoising filter is commonly used filter for forensics purposes (Khanna et al., 2008b, 2007a; Lukás et al., 2006), therefore we chose this filter as it was the most common choice in the literature (Fridrich & Goljan, 2011; Galdi et al., 2015, 2016) and our initial experiments confirmed the best suitability of this filter. It turned out to be the most discriminative of all denoising filters. We have observed that this MSE value seems to be unique for different scanners; thus, we consider it as a unique scanner fingerprint. The core of the algorithm is to calculate the MSE value on the difference of pixel intensities of only one colour channel of image \( I \) and its filtered version \( D \). Unlike in the Khanna et al.’s algorithm, we propose to process only one colour channel instead of all colour channels, which has a positive impact on the time of image processing. We also propose to process only small parts of scanned images, for example, of size \( 512 \times 512 \) pixels instead of the full resolution. This also speeds up the process of flatbed scanner identification. The MSE-DSI algorithm is repeated for each image from a particular scanner, and then the average value of MSE-DSIs is calculated and serves as the scanner fingerprint.

4. Experimental evaluation

4.1. MSE-DSI algorithm for digital camera identification

The MSE-DSI algorithm is tested for each colour channel, separately; algorithms presented by Mandelli et al.’s and Kirchner & Johnson are tested twofold: the same as the MSE-DSI algorithm (i.e. for each colour channel separately) and also as the original authors’ description (i.e. processing full images in RGB model in their original full resolution).

We have used a set of 45 modern digital single-lens reflex (DSLR) and mirrorless (DSLM) cameras, compact cameras and smartphones with built-in multiple sensors. This set includes the following models: Apple iPhone 8, Apple iPhone Xr, Canon 1D X Mark II, Canon 5D Mark IV, Canon 90D, Canon G9X Mark II, Canon M6 Mark II, Canon M10, Canon M100, Canon R, Canon RP, Huawei P9 Plus, Huawei P20 Pro, Nikon D3X, Nikon D5, Nikon D600, Nikon D610, Nikon D750, Nikon D810, Nikon D850, Nikon D7200, Nikon D7500, Nikon Z6, Nikon Z7, Panasonic Lumix GX800, Panasonic Lumix S1, Samsung S9 Plus, Samsung S10 Plus, Sony A7R III, Sony A7S, Sony A7S II, Sony A9, Sony A6500, Sony RX100 II, Sony Xperia 1, Sony Xperia XZ1, Xiaomi Mi 9. The number of all images is 1919, what means that at least 30 images were used per device.

The experiments were performed on an MSI notebook with Intel Core i5-7300HQ@2.5GHz CPU with 24 gigabytes of RAM and nVidia GTX 1050 GPU with 4 gigabytes of video memory.
4.1.1. Device identification

As evaluation, we use standard accuracy (ACC), true positive rate (TPR) and false positive rate (FPR) measures, defined as:

\[
\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN}, \quad \text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN}
\]

where TP/TN denotes ‘true positive/true negative’; FP/FN stands for ‘false positive/false negative’. TP denotes cases correctly classified to a specific class; TN are instances that are correctly rejected. FP denotes cases incorrectly classified to the specific class; FN are examples incorrectly rejected.

For the results of the classification, we present ROC curves and confusion matrices.

4.1.1.1. Results – ROC curves. For graphic visualization, we have used the ROC curves (receiver operating characteristic) with the AUC (area under the curve) measure. ROC curves show the relation between true positive rate (TPR) against the false positive rate (FPR). Results of classification are described in Figure 1.

The AUC for MSE-DSI equals 0.81 (red colour channel), 0.79 (green colour channel), 0.80 (blue colour channel), respectively. We have also checked, what is the impact of processing only one colour channel of images, just like in the proposed MSE-DSI algorithm. Results are presented as Figure 2.

The AUC for Mandelli et al.’s algorithm equals 0.87 (red colour channel), 0.87 (green colour channel), 0.85 (blue colour channel), respectively. Kirchner & Johnson obtained the AUC measure 0.75, 0.79 and 0.78 (red, green and blue colour channel, respectively).

Figure 1. ROC curves of camera model recognition of MSE-DSI (particular channels) vs. CNNs proposed by Mandelli et al.’s and Kirchner & Johnson (full image processing).

Figure 2. ROC curves of camera model recognition of MSE-DSI vs. CNNs proposed by Mandelli et al.’s and Kirchner & Johnson for particular channels.
Moreover, Mandelli and Kirchner & Johnson obtained the AUC of 0.95 and 0.88 (red, green and blue colour channel, respectively) for full image processing. This means that identification performed by CNN-based methods is slightly more accurate than the proposed MSE-DSI algorithm.

### Table 1. Confusion matrix of brand recognition, MSE-DSI algorithm, ACC = 0.80, red colour channel.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | 68.0| *   | *   | *   | 5.0 | 6.0 | *   | 8.0 |
| Canon | *   | 78.0| *   | *   | *   | *   | *   | *   |
| Huawei| Hua | 5.0 | 73.0| *   | *   | *   | *   | 5.0 |
| Nikon | Nik | *   | *   | 91.0| *   | *   | *   | *   |
| Panasonic| Pan | *   | *   | *   | 86.0| *   | *   | *   |
| Samsung| Sam | 5.0 | 5.0 | *   | *   | 72.0| *   | 6.0 |
| Sony | Son | *   | *   | *   | *   | *   | 82.0| 8.0 |
| Xiaomi| Xia | *   | *   | *   | *   | *   | *   | 91.0|

Note: The symbol * represents values smaller than 5%.

### Table 2. Confusion matrix of brand recognition, MSE-DSI algorithm, ACC = 0.74, green colour channel.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | 71.0| *   | *   | *   | *   | 5.0 | *   | 7.0 |
| Canon | *   | 79.0| *   | 9.0 | *   | *   | *   | *   |
| Huawei| Hua | 5.0 | 78.0| *   | *   | *   | *   | *   |
| Nikon | Nik | 10.0| *   | *   | 74.0| *   | *   | *   |
| Panasonic| Pan | *   | *   | 5.0 | 76.0| *   | 5.0 | *   |
| Samsung| Sam | *   | *   | 6.0 | *   | 72.0| 6.0 | 6.0 |
| Sony | Son | *   | *   | *   | 5.0 | *   | 72.0| 5.0 |
| Xiaomi| Xia | *   | *   | 5.0 | 5.0 | *   | 5.0 | 75.0|

Note: The symbol * represents values smaller than 5%.

### Table 3. Confusion matrix of brand recognition, MSE-DSI algorithm, ACC = 0.77, blue colour channel.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | 77.0| *   | *   | *   | *   | 6.0 | *   | *   |
| Canon | *   | 81.0| *   | 5.0 | *   | *   | *   | *   |
| Huawei| Hua | *   | 77.0| *   | *   | *   | *   | *   |
| Nikon | Nik | 9.0 | *   | 5.0 | 73.0| *   | *   | *   |
| Panasonic| Pan | *   | *   | *   | 74.0| *   | 8.0 | 5.0 |
| Samsung| Sam | *   | *   | *   | *   | 81.0| *   | *   |
| Sony | Son | 5.0 | *   | *   | *   | 74.0| *   | *   |
| Xiaomi| Xia | 5.0 | *   | *   | *   | *   | *   | 77.0|

Note: The symbol * represents values smaller than 5%.

### Table 4. Confusion matrix of brand recognition, Mandelli et al.’s algorithm, ACC = 0.89, red colour channel.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | 88.0| *   | *   | *   | *   | *   | *   | *   |
| Canon | Can | 91.0| *   | *   | *   | *   | *   | *   |
| Huawei| Hua | *   | 90.0| *   | *   | *   | *   | *   |
| Nikon | Nik | *   | *   | 89.0| *   | *   | *   | *   |
| Panasonic| Pan | *   | *   | *   | 87.0| *   | *   | *   |
| Samsung| Sam | *   | *   | *   | *   | 88.0| *   | *   |
| Sony | Son | *   | *   | *   | *   | 87.0| *   | *   |
| Xiaomi| Xia | *   | *   | *   | *   | *   | *   | 91.0|

Note: The symbol * represents values smaller than 5%.
Table 5. Confusion matrix of brand recognition, Mandelli et al.’s algorithm, ACC = 0.88, green colour channel.

|        | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple  | App | 87.0| *   | *   | *   | *   | *   | *   |
| Canon  | Can | *   | 92.0| *   | *   | *   | *   | *   |
| Huawei | Hua | *   | *   | 88.0| *   | *   | *   | *   |
| Nikon  | Nik | *   | *   | *   | 91.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 90.0| *   | *   |
| Samsung | Sam | *   | *   | *   | *   | *   | 85.0| *   |
| Sony   | Son | *   | *   | *   | *   | *   | 86.0| *   |
| Xiaomi | Xia | *   | *   | *   | *   | *   | *   | 87.0|

Note: The symbol * represents values smaller than 5%.

Table 6. Confusion matrix of brand recognition, Mandelli et al.’s algorithm, ACC = 0.87, blue colour channel.

|        | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple  | App | 85.0| *   | *   | *   | *   | *   | *   |
| Canon  | Can | *   | 86.0| *   | *   | *   | *   | *   |
| Huawei | Hua | *   | *   | 85.0| *   | *   | *   | *   |
| Nikon  | Nik | *   | *   | *   | 83.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 84.0| *   | *   |
| Samsung | Sam | 5.0 | *   | *   | *   | 83.0| *   | *   |
| Sony   | Son | *   | *   | *   | *   | *   | 84.0| *   |
| Xiaomi | Xia | *   | *   | *   | *   | *   | *   | 89.0|

Note: The symbol * represents values smaller than 5%.

Table 7. Confusion matrix of brand recognition, Kirchner & Johnson, ACC = 0.85, red colour channel.

|        | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple  | App | 86.0| *   | *   | *   | *   | *   | *   |
| Canon  | Can | *   | 85.0| *   | *   | *   | *   | *   |
| Huawei | Hua | *   | *   | 83.0| *   | *   | *   | *   |
| Nikon  | Nik | *   | *   | *   | 82.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 83.0| *   | *   |
| Samsung | Sam | 6.0 | *   | *   | *   | 82.0| *   | *   |
| Sony   | Son | *   | *   | *   | *   | *   | 84.0| *   |
| Xiaomi | Xia | *   | *   | *   | *   | *   | *   | 86.0|

Note: The symbol * represents values smaller than 5%.

Table 8. Confusion matrix of brand recognition, Kirchner & Johnson, ACC = 0.84, green colour channel.

|        | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple  | App | 86.0| *   | *   | *   | *   | *   | *   |
| Canon  | Can | *   | 85.0| *   | *   | *   | *   | *   |
| Huawei | Hua | *   | *   | 83.0| *   | *   | *   | *   |
| Nikon  | Nik | *   | *   | *   | 82.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 83.0| *   | *   |
| Samsung | Sam | 6.0 | *   | *   | *   | 82.0| *   | *   |
| Sony   | Son | *   | *   | *   | *   | *   | 84.0| *   |
| Xiaomi | Xia | *   | *   | *   | *   | *   | *   | 88.0|

Note: The symbol * represents values smaller than 5%.
4.1.1.2. Results – Confusion matrices. We have also used the confusion matrices for the presentation of classification results. The confusion matrices of brand recognition of MSE-DSI algorithm, Mandelli et al.’s algorithm and Kirchner & Johnson are presented in Tables 1–11.

Due to the large number of examined cameras (45 devices), we do not present for clarity confusion matrices of model recognition. Instead of this we provide an information that the ACC = 0.81 for model recognition in MSE-DSI algorithm, while Mandelli et al.’s algorithm obtained ACC = 0.97 and Kirchner & Johnson ACC = 0.94. The TPRs for Mandelli et al.’s algorithm algorithm are of 0.96, 0.97, 0.98 and 0.99 for the most cameras and only the Canon G9X Mark II obtained 0.94, Canon M100 – 0.95, Canon R – 0.96, Huawei P20 Pro artificial intelligence-based camera – 0.92, Nikon D750 – 0.95, Panasonic Lumix GX800 – 0.94, Samsung S9 plus main camera – 0.95 and Sony RX100 II – 0.95.

### Table 9. Confusion matrix of brand recognition, Kirchner & Johnson, ACC = 0.86, blue colour channel.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | App | 81.0| *   | *   | 7.0 | *   | *   | 5.0 |
| Canon | Can | *   | 89.0| *   | *   | *   | *   | *   |
| Huawei| Hua | *   | *   | 88.0| *   | *   | *   | *   |
| Nikon | Nik | *   | *   | *   | 87.0| *   | *   | 5.0 |
| Panasonic | Pan | *   | 5.0 | *   | *   | 85.0| *   | *   |
| Samsung | Sam | *   | *   | *   | *   | 85.0| *   | *   |
| Sony  | Son | *   | 7.0 | *   | *   | *   | *   | 84.0|
| Xiaomi| Xia | *   | 5.0 | *   | 5.0 | *   | *   | 85.0|

Note: The symbol * represents values smaller than 5%.

### Table 10. Confusion matrix of brand recognition, Mandelli et al.’s algorithm, ACC = 0.98, full image processing.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | App | 98.0| *   | *   | *   | *   | *   | *   |
| Canon | Can | *   | 98.0| *   | *   | *   | *   | *   |
| Huawei| Hua | *   | *   | 97.0| *   | *   | *   | *   |
| Nikon | Nik | *   | *   | *   | 98.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 97.0| *   | *   |
| Samsung | Sam | *   | *   | *   | *   | *   | 99.0| *   |
| Sony  | Son | *   | *   | *   | *   | *   | *   | 99.0|
| Xiaomi| Xia | *   | *   | *   | *   | *   | *   | 98.0|

Note: The symbol * represents values smaller than 5%.

### Table 11. Confusion matrix of brand recognition, Kirchner & Johnson, ACC = 0.95, full image processing.

|       | App | Can | Hua | Nik | Pan | Sam | Son | Xia |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| Apple | App | 96.0| *   | *   | *   | *   | *   | *   |
| Canon | Can | *   | 94.0| *   | *   | *   | *   | *   |
| Huawei| Hua | *   | *   | 95.0| *   | *   | *   | *   |
| Nikon | Nik | *   | *   | *   | 96.0| *   | *   | *   |
| Panasonic | Pan | *   | *   | *   | *   | 97.0| *   | *   |
| Samsung | Sam | *   | *   | *   | *   | *   | 95.0| *   |
| Sony  | Son | *   | *   | *   | *   | *   | *   | 95.0|
| Xiaomi| Xia | *   | *   | *   | *   | *   | *   | 93.0|

Note: The symbol * represents values smaller than 5%.
The TPR values for MSE-DSI algorithm for most cameras are not lower than 0.8, however, classification results for some cameras are not satisfactory. For example, the TPR for Apple iPhone 8 Plus tele was 0.52, Canon M10 – 0.51, Canon R – 0.47, Nikon D610 – 0.46, Sony A7S – 0.4, Sony Xperia XZ1 – 0.26. The following cameras were not recognized: Huawei P20 Pro AI – 0.0, Samsung S10 Plus tele camera – 0.0, Sony A6500 – 0.0, Sony Xperia 1 ultrawide – 0.0.

The state-of-the-art algorithms are better in classification because they process all image colour channels as well as the full image size. Therefore, more pixels are processed and the CNNs are more precisely trained. The proposed algorithm processes only one colour channel and a fragment of this channel instead of the whole matrix. That is why it is less accurate. Results also clearly indicate that one may process any colour channel, as well as the proposed MSE-DSI algorithm and both Mandelli et al.’s algorithm and Kirchner & Johnson obtained similar classification accuracy for particular colour channels.

To sum up, the classification results intuitively indicate that processing smaller amount of data is associated with smaller classification accuracy. Processing full resolution images give the highest classification accuracy, while reducing images to only one colour channel decreases the outcomes.

4.1.2. Time performance
The experiments showed that the CNN-based methods process images much longer than the proposed MSE-DSI algorithm. At average, one image of MSE-DSI algorithm is processed in about 6 s. The time for learning one epoch with considered CNNs is about 1.5 min. However, if we apply only one colour channel images to CNNs, their time for learning decreases by about 0.5 a minute. Therefore, calculating the MSE values of all

![Figure 3. Comparison of time performance of MSE-DSI algorithm vs. considered CNNs.](image-url)
tested 1919 images took about 3.5 h, while CNNs (particular colour channels and full image processing) needed 16.5 and 50 h, respectively.

Graphical interpretation of time performance comparison is presented in Figure 3.

The main reason for the long processing of the images is the usage of a denoising filter on all three colour channels of the image, which is computationally inefficient. In our method, we propose processing only a fragment of an image of size $512 \times 512$ pixels instead of the whole image what is faster, and we also apply it to only one colour channel of the image instead of all three colour channels.

### 4.2. MSE-DSI algorithm for flatbed scanner identification

In this section, we present the results of scanner identification by the MSE-DSI algorithm and compare them with state-of-the-art Khanna et al.’s algorithm (Khanna et al., 2007b). The following 10 scanners were used: Brother MFC 9970CDW, Canon C2020i, HP Deskjet F4180, HP Laser Jet M1005 MFP, HP ScanJet 3670, HP ScanJet PLS 2800, OKI MC562w, PLUSTEK, Ricoh SP 112SU and Samsung SCX-3205. Scanner classification was performed with a set of 290 JPEG photographs (29 images per device). Sample images can be seen in Figure 4. Scripts for the proposed method and Khanna et al.’s algorithm were implemented in MATLAB (http://dde.binghamton.edu/download/). All the tested scanners were connected to computers with installed Microsoft Windows 8.1 or 10 operating systems. We used the Fax & Scan application in Windows OS to manage the scanners. All the scanners were set to their default settings, and all the images were scanned at 300 dpi resolution.

#### 4.2.1. Device identification

Classification was performed with the $k$-nearest neighbours algorithm ($k$-NN) with experimentally picked $k=5$ as the best value.

Figure 4. Sample images used in experiments.
First, the MSE-DSI values for each device (based on 29 images) are calculated. A new image is acquired with a particular device and its MSE-DSI value is calculated. Then, the device is classified by a plurality vote, i.e. to the class most common among its $k$ nearest MSE-DSI values. All the experiments were performed with 10-fold cross-validation. The results of device classification by the proposed methods are presented in Figure 5 (Table 12–15).

The results of classification presented in confusion matrices point that the proposed method is less accurate in scanner classification compared with the state-of-the-art algorithm. The Khanna et al.’s algorithm obtained the $AUC = 0.94$. The proposed MSE-DSI algorithm gives the $AUC = 0.81$ accuracy for model recognition.

The proposed algorithm may even work as preprocessing, e.g. for pre-selection of photos according to the device model before subjecting these photos to further analysis.

**Table 12.** Confusion matrix of model recognition for the proposed MSE-DSI algorithm, $ACC = 0.82$.

|     | Bro  | Can | HP1 | HP2 | HP3 | HP4 | OKI | Plu | Ric | Sam |
|-----|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bro | 89.7 | 3.4 | *   | *   | *   | 6.9 | *   | *   | *   | 0.0 |
| Can | 10.3 | 41.4| *   | 10.3| 20.7| 17.2| *   | *   | *   | *   |
| HP1 | *   | *   | 41.4| *   | *   | *   | 10.3| *   | *   | 48.3|
| HP2 | *   | *   | *   | 93.1| *   | *   | *   | *   | *   | 6.9 |
| HP3 | *   | 17.2| *   | 6.9 | 75.9| *   | *   | *   | *   | *   |
| HP4 | *   | 13.8| *   | *   | 86.2| *   | *   | *   | *   | *   |
| OKI | *   | *   | *   | *   | *   | 100.0| *   | *   | *   | *   |
| Plu | *   | *   | *   | *   | *   | *   | 100.0| *   | *   | *   |
| Ric | *   | *   | *   | *   | *   | *   | *   | 100.0| *   | *   |
| Sam | *   | *   | 24.1| *   | *   | *   | *   | *   | *   | 75.9|

**Figure 5.** ROC curve of scanner model recognition of MSE-DSI vs. Khanna et al.’s algorithm.
by a more sophisticated or accurate algorithm. The advantage of the proposed algorithm is processing a small amount of data, which makes the algorithm fast. However, at the same time, this is a potential weakness because the smaller amount of data makes the algorithm slightly less accurate.

4.2.2. Time performance
The experiments showed that Khanna et al.’s algorithm processes images much longer than the proposed MSE-DSI algorithm. On average, one image in the MSE-DSI algorithm is processed in about 2 s. Therefore, calculating the MSE values of all tested 290 images took nearly 10 min. Khanna et al.’s algorithm processes one image on average in 120 s, resulting in total time for processing 290 images at nearly 10 h. Graphical interpretation of time performance comparison is presented in Figure 6.

4.2.3. Remark about image histograms
We have also observed some interesting features in histograms of scanned images. Images scanned with Brother MFC 9970CDW and Samsung SCX-3205 have visible dark

| Table 13. Confusion matrix of model recognition for the proposed MSE-DSI algorithm, ACC = 0.89. |
|-----------|---|---|---|---|---|---|---|
| Bro | Can | HP | OKI | Plu | Ric | Sam |
| Brother | 71.4 | * | 14.3 | * | * | 14.3 |
| Canon | * | 100.0 | * | * | * | * |
| HP | * | 71.4 | 14.3 | * | * | 14.3 |
| OKI | * | * | 100.0 | * | * | 0.0 |
| Plustek | * | * | * | * | 100.0 | * |
| Ricoh | * | * | * | * | * | 100.0 |
| Samsung | * | * | 20.0 | * | * | 80.0 |

| Table 14. Confusion matrix of brand recognition for the Khanna et al.’s algorithm, ACC = 0.78. |
|-----------|---|---|---|---|---|---|---|
| Bro | Can | HP1 | HP2 | HP3 | HP4 | OKI | Plu | Ric | Sam |
| Bro | 75.0 | * | 5.0 | * | * | 10.0 | * | 8.0 | * |
| Canon | * | 76.0 | * | 20.0 | * | * | 7.0 | * | * |
| HP1 | * | * | 80.0 | * | * | * | 5.0 | * | 11.0 |
| HP2 | * | * | 8.0 | 87.0 | 5.0 | * | * | * | * |
| HP3 | * | * | * | 6.0 | 60 | 81.0 | * | * | * |
| HP4 | * | * | * | 11.0 | * | * | 82.0 | * | * |
| OKI | 15.0 | * | 5.0 | * | * | 70.0 | * | 7.0 | * |
| Plu | * | * | * | * | * | * | 7.0 | 83.0 | * |
| Ric | * | * | * | * | * | * | * | 82.0 | 10.0 |
| Sam | * | * | * | * | * | 15.0 | * | * | 77.0 |

| Table 15. Confusion matrix of brand recognition for the Khanna et al.’s algorithm, ACC = 0.80. |
|-----------|---|---|---|---|---|---|---|
| Bro | Can | HP | OKI | Plu | Ric | Sam |
| Brother | 75.0 | * | 5.0 | 10.0 | * | 8.0 | * |
| Canon | * | 76.0 | 10.0 | * | 7.0 | * | * |
| HP | * | * | 82.7 | * | * | * | 11.8 |
| OKI | 15.0 | * | 8.0 | 70.0 | * | 7.0 | 0.0 |
| Plustek | * | * | * | 7.0 | 83.0 | * | * |
| Ricoh | * | * | * | * | * | 82.0 | 10.0 |
| Samsung | * | * | 5.0 | 15.0 | * | * | 77.0 |
Figure 6. Comparison of time performance of Khanna et al. and MSE-DSI algorithm.

Table 16. The average percentage of 0, 1 and 2 pixel intensities occurring in the original image and scanned by Brother and Samsung scanners.

| Scanner Type          | Average Percentage |
|-----------------------|--------------------|
| Original image        | 0.1%               |
| Brother MFC 9970CDW   | 6.4%               |
| Samsung SCX-3205      | 0.5%               |

Figure 7. (1): original (digital) photo; (2): image scanned by Brother MFC 9970CDW, (3): image scanned by Samsung SCX-3205. The dark parts of the images are clearly visible in images’ histograms.
peaks’ in its histograms compared with the original images and also other scanners. What is more, comparing Brother MFC 9970CDW with Samsung SCX-3205 in all scanned images, dark peaks are higher in case of Brother’s scanner. Of course, such images are visually darker and their contrast is higher. We have analysed the percentage occurrence of the darkest pixel intensities (0, 1 and 2) of original images (from camera) and images scanned with Brother and Samsung devices. Results are presented in Table 16. The highest percentage is observed in case of Brother scanner (6.4 percent of 0, 1 and 2 pixel intensities of overall number of pixels). Smaller peaks occur in Samsung scanner, but simultaneously higher than in original images. Some example images are presented in Figure 7.

We compared percentage values of 0, 1 and 2 pixel intensities of original (digital) images and scanned by Brother and Samsung devices in order to determine if they differ statistically. These values were grouped as sequences, corresponding to each device. Due to lack of normality performed by Shapiro-Wilk tests (Chen, 1971), the Ott and Longnecker (2006) non-parametric Anova Cowan (1998) was conducted with POST-HOC analysis (Johnson & Wichern, 2002; Levine & Ensom, 2001). The value \( p=2.89 \times 10^{-11} \) means that mean ranks of tested sequences differ statistically. Graphical interpretation of POST-HOC analysis is presented in Figure 8.

Results of the statistical analysis confirm that dark peaks in histograms of scanned images are statistically different than in original digital images. The highest dark peaks are observed for Brother’s scanner, a bit smaller for Samsung. Both scanners leave dark peaks statistically higher than in original images; therefore, one may consider them as a specific device trace. Although such observation cannot be used for a reasonable scanner identification.

5. Statistical analysis of results

We have conducted a statistical analysis of whether the classification results of the proposed and state-of-the-art algorithms differ significantly. For this purpose, we took the values of True Positive Rates from the confusion matrices and compared them to each other.
5.1. Flatbed scanner classification

In this test, we statistically compared the results of model classification of MSE-DSI with Khanna et al.’s algorithm. First of all, normality tests were performed. The hypotheses are defined as follows:

- \( H_0 \): The sample comes from the normal distribution;
- \( H_1 \): The sample does not come from the normal distribution.

The Kolmogorov-Smirnov test of normality showed that both analysed datasets come from the normal distribution. The \( p \)-value of data from the proposed algorithm and Khanna et al.’s is 0.62 and 0.93, so the hypothesis of normality was fulfilled. The values of test statistics are 0.22 for the MSE-DSI algorithm and 0.15 for Khanna et al’s. Thus, for further analysis, the parametric test of Student’s \( t \)-distribution was used. Hypotheses relate to equality of the averages \( \mu \) of examined samples:

- \( H_0 \): \( \mu_1 = \mu_2 \);
- \( H_1 \): \( \mu_1 \neq \mu_2 \).

The Student’s \( t \)-test is performed with 18 degrees of freedom. The test statistic equals 0.14 and the two-sided (with Cochran-Cox correction) (Azzalini & Capitanio, 2003) \( p \)-value is 0.89. It means that the considered samples do not differ significantly in terms of statistical analysis. Therefore, the proposed algorithm provides statistically the same performance as the Khanna et al.’s approach.

5.2. Digital camera classification

Similar reasoning was also conducted for digital camera identification, where classification results of the proposed algorithm were compared with the CNNs proposed by Mandelli et al.’s and Kirchner & Johnson. Obviously, analysis begins with normality tests, and the hypotheses are defined as follows:

- \( H_0 \): The sample comes from the normal distribution;
- \( H_1 \): The sample does not come from the normal distribution.

The Shapiro-Wilk and Lilliefors tests (Iman, 1982) of normality showed that none of analysed data came from the normal distribution. In all cases, \( p \)-value was < 0.001 so the hypothesis of normality was rejected. Thus, for further analysis the non-parametric test (Kruskal-Wallis ANOVA) was used. Hypotheses relate to equality of the average rank for next samples or are simplified to the median \( \theta \):

- \( H_0 \): \( \theta_1 = \theta_2 = \ldots = \theta_j \);
- \( H_1 \): Not all \( \theta_j \) are equal (\( j = 1, 2, \ldots, k \)).

The Kruskal-Wallis ANOVA test indicated that the test statistic was 26.05 and the \( p \)-value was < 0.001. This means that there exists a statistical difference between the tested samples. The average group rank for MSE-DSI algorithm is 45.93; for Mandelli
et al.’s algorithm it is 86.02, Kirchner & Johnson: 76.54. Therefore, the CNNs proposed by Mandelli et al.’s and Kirchner & Johnson achieve better classification results in terms of statistical analysis. However, the MSE-DSI method is much faster, and this fact can be considered as the trade-off between classification accuracy and speed which is important in, e.g. IoT applications.

6. Conclusions and future work

We proposed the MSE-DSI algorithm for image device identification and compared it with the state-of-the-art algorithms for flatbed scanners and digital camera identification. It can be used, for example, in IoT network security by detecting imaging device compromise or in forensics. Experimental results showed a slightly better classification performance of the proposed method. Moreover, the MSE-DSI algorithm is more efficient than the state-of-the-art algorithms in terms of the processing time (about two orders of magnitude). Khanna et al.’s algorithm and CNNs proposed by Mandelli et al.’s and Kirchner & Johnson calculate the image device fingerprint by denoising all colour channels of the scanned images at their full resolution. We proposed to process only one colour channel, which significantly speeds up the process of image processing. Furthermore, we process only a fragment of the scanned image of size 512 × 512 pixels. Calculating device fingerprints for 290 images by the proposed method takes about 10 min, while the Khanna/CNNs need almost 10 h. The processing speed is crucial in IoT networks, where we usually deal with energy-efficient devices, low computational requirements, and rather dense traffic. Future work will concern further analysis of the proposed MSE-DSI algorithm in order to increase its classification accuracy. We are going to extend the experiments in order to check if our method works for several flatbed scanners of the same model.

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes on contributors

Jarosław Bernacki received the B.Sc. degree from the Faculty of Fundamental Problems of Technology and the M.Sc. degree from the Faculty of Computer Science and Management in computer science, Wrocław University of Science and Technology. He is currently pursuing the Ph.D. degree in computer science. His main research interests include image processing, anonymity, privacy, cryptography and computer networks.

Rafał Scherer (Member, IEEE) received the M.S. degree in electrical engineering from the Department of Electrical Engineering, Czestochowa University of Technology, and the Ph.D. degree in computer science (methods of classification using neuro-fuzzy systems) from the Department of Mechanical Engineering and Computer Science, Czestochowa University of Technology. He is currently an Associate Professor with the Institute of Computational Intelligence, Czestochowa University of Technology. He is also a co-coordinator of the Microsoft Dynamics Academic Alliance Program, Czestochowa University of Technology. He was a Principal Investigator of the Polish Ministry of Science and Higher Education Project “Computational Intelligence Methods in Data Mining” and a Researcher of the Polish-Singapore Research Project “Development of Intelligent Techniques for Modeling, Controlling and Optimizing Complex Manufacturing Systems.” He has authored a book on multiple classification techniques published in Springer. He has authored more than 80 research articles. His research interests include developing new methods in computational
intelligence and data mining, ensembling methods in machine learning, and content-based image indexing. He was a reviewer for major computational intelligence journals. He is also a Co-Editor of the Journal of Artificial Intelligence and Soft Computing Research. He co-organizes every year or two years the International Conference on Artificial Intelligence and Soft Computing in Zakopane, which is one of the major events on computational intelligence.

**ORCID**

Jarosław Bernacki [http://orcid.org/0000-0002-4488-3488](http://orcid.org/0000-0002-4488-3488)

Rafał Scherer [http://orcid.org/0000-0001-9592-262X](http://orcid.org/0000-0001-9592-262X)

**References**

Agarwal, A., Singh, R., & Vatsa, M. (2016, December 4–8). Fingerprint sensor classification via mélange of handcrafted features. In 23rd International Conference on Pattern Recognition, ICPR 2016 (pp. 3001–3006). IEEE.

Amerini, I., Caldelli, R., Cappellini, V., Picchioni, F., & Piva, A. (2009). Analysis of denoising filters for photo response non uniformity noise extraction in source camera identification. In 2009 16th International Conference on Digital Signal Processing (pp. 1–7). IEEE.

Azzalini, A., & Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew t-distribution. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2), 367–389. [https://doi.org/10.1111/rssb.2003.65.issue-2](https://doi.org/10.1111/rssb.2003.65.issue-2)

Baar, T., van Houten, W., & Geradts, Z. J. M. H. (2012). Camera identification by grouping images from database, based on shared noise patterns. CoRR, abs/1207.2641.

Bondi, L., Baroffio, L., Guera, D., Bestagini, P., Delp, E. J., & Tubaro, S. (2017). First steps toward camera model identification with convolutional neural networks. *IEEE Signal Processing Letters*, 24(3), 259–263. [https://doi.org/10.1109/LSP.2016.2641006](https://doi.org/10.1109/LSP.2016.2641006)

Chen, E. H. (1971). The power of the shapiro-wilk w test for normality in samples from contaminated normal distributions. *Journal of the American Statistical Association*, 66(336), 760–762.

Chen, Y., Huang, Y., & Ding, X. (2017, September 17–20). Camera model identification with residual neural network. In 2017 IEEE International Conference on Image Processing, ICIP 2017 (pp. 4337–4341). IEEE.

Cowan, G. (1998). *Statistical data analysis*. Oxford University Press.

Deng, Z., Gijssenij, A., & Zhang, J. (2011, November 6–13). Source camera identification using auto-white balance approximation. In *IEEE International Conference on Computer Vision, ICCV 2011* (pp. 57–64). IEEE.

Dirik, A. E., Sencar, H. T., & Memon, N. D. (2009, April 19–24). Flatbed scanner identification based on dust and scratches over scanner platen. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2009* (pp. 1385–1388). IEEE.

Freire-Obregón, D., Narducci, F., Barra, S., & Santana, M. C. (2017). Deep learning for source camera identification on mobile devices. CoRR, abs/1710.01257.

Fridrich, J. J., & Goljan, M. (2011, January 24–26). Determining approximate age of digital images using sensor defects. In *Media Forensics and Security III, Proceedings* (p. 788006).

Galdi, C., Nappi, M., & Dugelay, J. (2015, September 7–11). Combining hardwaremetry and biometry for human authentication via smartphones. In *Image Analysis and Processing –ICIAP 2015 – 18th International Conference, Proceedings, Part II* (pp. 406–416). ACM Digital Library.

Galdi, C., Nappi, M., & Dugelay, J. (2016). Multimodal authentication on smartphones: Combining iris and sensor recognition for a double check of user identity. *Pattern Recognition Letters*, 82(12), 144–153. [https://doi.org/10.1016/j.patrec.2015.09.009](https://doi.org/10.1016/j.patrec.2015.09.009)

Gloe, T., Franz, E., & Winkler, A. (2007, January 28). Forensics for flatbed scanners. In *Security, Steganography, and Watermarking of Multimedia Contents IX* (p. 65051I). SPIE Digital Library.

Goljan, M. (2008, November 10–12). Digital camera identification from images – estimating false acceptance probability. In *Digital Watermarking, 7th International Workshop, IWDW 2008. Selected Papers* (pp. 454–468). Springer.
Goljan, M., Chen, M., Comesaña, P., & Fridrich, J. J. (2016, February 14–18). Effect of compression on sensor-fingerprint based camera identification. In Media Watermarking, Security, and Forensics 2016 (pp. 1–10).

Gou, H., Swaminathan, A., & Wu, M. (2007, January 28). Robust scanner identification based on noise features. In Security, Steganography, and Watermarking of Multimedia Contents IX (p. 65050S). SPIE Digital Library.

Iman, R. L. (1982). Graphs for use with the Lilliefors test for normal and exponential distributions. The American Statistician, 36(2), 109–112.

Jiang, X., Wei, S., Zhao, R., Zhao, Y., & Wu, X. (2016). Camera fingerprint: A new perspective for identifying user's identity. CoRR, abs/1610.07728.

Johnson, R., & Wichern, D. (2002). Applied multivariate statistical analysis (5th ed). Prentice Hall.

Khanna, N., Chiu, G. T., Allebach, J. P., & Delp, E. J. (2007a, January 28). Forensic classification of imaging sensor types. In Security, Steganography, and Watermarking of Multimedia Contents IX (p. 65050U). SPIE Digital Library.

Khanna, N., Mikkilineni, A. K., & Delp, E. J. (2009). Scanner identification using feature-based processing and analysis. IEEE Transactions on Information Forensics and Security, 4(1), 123–139. https://doi.org/10.1109/TIFS.2008.2009604

Kirchner, M., & Johnson, C. (2020). SPN-CNN: boosting sensor-based source camera attribution with deep learning. CoRR, abs/2002.02927.

Levine, M., & Ensom, M. H. H. (2001). Post hoc power analysis: An idea whose time has passed?. Pharmacotherapy: The Journal of Human Pharmacology and Drug Therapy, 21(4), 405–409. https://doi.org/10.1592/phco.2001.21.issue-4

Li, R., Li, C., & Guan, Y. (2018). Inference of a compact representation of sensor fingerprint for source camera identification. Pattern Recognition, 74(2), 556–567. https://doi.org/10.1016/j.patcog.2017.09.027

Lukás, J., Fridrich, J. J., & Goljan, M. (2006). Digital camera identification from sensor pattern noise. IEEE Trans. Information Forensics and Security, 1(2), 205–214. https://doi.org/10.1109/TIFS.2006.873602

Lyu, S., & Farid, H. (2005). How realistic is photorealistic?. IEEE Transactions on Signal Processing, 53(2–2), 845–850. https://doi.org/10.1109/TSP.2004.839896

Mandelli, S., Cozzolino, D., Bestagini, P., Verdoliva, L., & Tubaro, S. (2020). Cnn-based fast source device identification. IEEE Signal Processing Letters, 27, 1285–1289. https://doi.org/10.1109/LSP.97

Mihçak, M. K., Kozintsev, I., & Ramchandran, K. (1999, March 15–19). Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising. In Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP ’99 (pp. 3253–3256). IEEE.

Ott, R. L., & Longnecker, M. T. (2006). Introduction to statistical methods and data analysis (with CD-ROM). Duxbury Press.

Rafi, A. M., Kamal, U., Hoque, R., Abrar, A., Das, S., Laganière, R., & Hasan, M. K. (2019, June 16–20). Application of densenet in camera model identification and post-processing detection. In IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019 (pp. 19–28). IEEE.
Tomioka, Y., & Kitazawa, H. (2011, July 11–15). Digital camera identification based on the clustered pattern noise of image sensors. In Proceedings of the 2011 IEEE International Conference on Multimedia and Expo, ICME 2011 (pp. 1–4). IEEE.

Tuama, A., Comby, F., & Chaumont, M. (2016). 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). IEEE.

Tuama, A., Comby, F., & Chaumont, M. (2016, August 29–September 2). Camera model identification based machine learning approach with high order statistics features. In 24th European Signal Processing Conference, EUSIPCO 2016 (pp. 1183–1187). IEEE.

Valsesia, D., Coluccia, G., Bianchi, T., & Magli, E. (2015). Compressed fingerprint matching and camera identification via random projections. IEEE Transactions on Information Forensics and Security, 10 (7), 1472–1485. https://doi.org/10.1109/TIFS.2015.2415461

Yang, P., Ni, R., Zhao, Y., & Zhao, W. (2019). Source camera identification based on content-adaptive fusion residual networks. Pattern Recognition Letters, 119, 195–204. https://doi.org/10.1016/j.patrec.2017.10.016

Yao, H., Qiao, T., Xu, M., & Zheng, N. (2018). Robust multi-classifier for camera model identification based on convolution neural network. IEEE Access, 6, 24973–24982. https://doi.org/10.1109/ACCESS.2018.2832066