AFRL-SR-AR-TR-09-0051

## REPORT DOCUMENTATION PAGE

| | | |
|---|---|---|
| 1. REPORT DATE *(DD-MM-YYYY)*<br>06-02-2009 | 2. REPORT TYPE<br>Final Performance Report | 3. DATES COVERED *(From - To)*<br>01-01-2006 to 30-11-2008 |

4. TITLE AND SUBTITLE
Advanced Steganographic and Digital Forensic Methods

5a. CONTRACT NUMBER
FA9559-06-1-0046

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Dr. Jessica Fridrich

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The Research Foundation of State University of New York
SUNY at Binghamton
4400 Vestal Parkway East
Binghamton, NY 13902-6000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
USAF, AFRL
AF Office of Scientific Research
875 N. Randolph Street RM 3112
Arlington, VA
Dr Herklotz/NL

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR/PKA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
All data delivered is approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The author developed and advanced a new class of digital forensic techniques to verify origin and integrity of digital imagery based on systematic artifacts of imaging sensors called photo-response non-uniformity (PRNU), which is caused by slight variations in physical dimensions of pixels and inhomogeneity of silicon. PRNU thus forms a unique fingerprint that characterizes an imaging device, such as a camera, camcorder, or scanner. Both CCD and CMOS technologies exhibit this type of irregularity. The specific achievements include perfecting the methodology for estimating the fingerprint from images, extending to cases when the image under investigation is simultaneously cropped, scaled, and processed, extending the technology when the digital image is printed, developing technology capable of determining the camera model from the fingerprint, and a large scale validation on millions of images from 6900 cameras of 150 models. The techniques have dual purpose and are important for information validation in military intelligence gathering, law enforcement, and forensic investigation.

15. SUBJECT TERMS
Digital Forensics, sensor, fingerprint, identification, forgery detection, authentication, information assurance, media security

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Jessica Fridrich |
| Unclassified | Unclassified | Unclassified | Unlimited | 43 | 19b. TELEPHONE NUMBER *(Include area code)*<br>607 777 6177 |

# Final Report

**Effort title:** Advanced Digital Forensic and Steganalysis Methods

**Principal Investigator:** Jessica Fridrich, Professor
Department of Electrical and Computer Engineering
Binghamton University
T. J. Watson School
Binghamton, NY 13902-6000
Ph: 607-777-6177, Fax: 607-777-4464

**Agreement Number:** FA95500610046

## OBJECTIVES

1. Develop new digital forensic methods for identifying camera from images or video clips.
2. Develop digital forensic methods for detection and localization of tampering.
3. Implement the methods in Matlab and deliver the code to US Air Force and FBI.
1. Assist in technology transfer to law enforcement and government.

## PERSONNEL SUPPORTED

Dr. Jessica Fridrich, PI
Dr. Miroslav Goljan, Postdoctoral Assistant
Dr. Mo Chen, Postdoctoral Assistant
Mr. Tomáš Filler, PhD student
Dr. Paul Blythe, DDE Lab Manager

## LIST OF PUBLISHED PAPERS AND DISSERTATIONS

1. J. Lukáš, J. Fridrich, and M. Goljan: "Digital Camera Identification from Sensor Pattern Noise." *IEEE Transactions on Information Security and Forensics*, vol. 1(2), pp. 205–214, June 2006.
2. M. Goljan and J. Fridrich: "Camera Identification from Cropped and Scaled Images." *Proc. SPIE Electronic Imaging, Forensics, Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, San Jose, California, January 28 – 30, pp. 0E-1–0E-13 2008.
3. M. Goljan and J. Fridrich: "Camera Identification from Printed Images." *Proc. SPIE Electronic Imaging, Forensics, Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, San Jose, California, January 28 – 30, pp. OI-1–OI-12, 2008.
4. M. Goljan, Mo Chen, and J. Fridrich: "Identifying Common Source Digital Camera from Image Pairs." *Proc. IEEE ICIP 07*, San Antonio, TX, 2007.
5. Mo Chen, J. Fridrich, and M. Goljan: "Source Digital Camcorder Identification Using CCD Photo Response Non-uniformity." *Proc. SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, San Jose, California, January 28 – February 1, pp. 1G–1H, 2007.
6. J. Fridrich, Mo Chen, M. Goljan, and J. Lukáš, "Digital Imaging Sensor Identification (Further Study)." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, San Jose, CA, January 28–February 2, pp. 0P–0Q, 2007.
7. Lukáš, J., Fridrich, J., and Goljan, M.: "Detecting Digital Image Forgeries Using Sensor Pattern Noise." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII*, vol. 6072. San Jose, California, pp. 0Y1–0Y11, 2006.
8. Mo Chen, J. Fridrich, M. Goljan, and J. Lukáš: "Determining Image Origin and Integrity Using Sensor Noise." *IEEE Transactions on Information Security and Forensics*, vol. 1(1), pp. 74–90, March 2008.

9. J. Fridrich, Mo Chen, J. Lukáš, and M. Goljan, "Imaging Sensor Noise as Digital X-Ray for Revealing Forgeries." 9th Information Hiding Workshop, Saint Malo, France, June 9–11, LNCS, 4567, Springer-Verlag, pp. 342–358, 2007.
10. T. Filler, J. Fridrich, and M. Goljan: "Using Sensor Pattern Noise for Camera Model Identification." *Proc. IEEE ICIP 08*, San Diego, CA, September 2008
11. J. Fridrich, "Digital Image Forensics using Sensor Noise," to appear in Special Issue Signal processing Magazine, 2009 (to appear).
12. M. Goljan, J. Fridrich, and T. Filler: "Camera Identification – Large Scale Test." *Proc. SPIE, Electronic Imaging, Security and Forensics of Multimedia Contents XI,* San Jose, CA, January 18–22, 2009 (to appear).

All papers were presented at the above noted conferences or published in respective journals.

PhD Dissertations

J. Lukáš: *Digital Image Forensics using Sensor Pattern Noise,* Ph.D. Dissertation, SUNY Binghamton, Department of Electrical and Computer Engineering, June 2006.

## INTERACTIONS/TRANSITIONS

The research findings were presented to peers at professional meetings, such as the Annual AFOSR PI meetings, IEEE International Conference on Image Processing (ICIP) in San Antonio, Texas, in September 2007, IEEE International Conference on Image Processing (ICIP) in San Diego, California, in September 2008, 9th Information Hiding Workshop in 2007, and multiple times at SPIE Electronic Imaging in San Jose in January 2006–2009. The research was also briefed to FBI forensic investigators and to AFRL in July 2008.

The PI and the co-PI, Miroslav Goljan, provided expert service to Scottish law enforcement agency in connection with a child pornography case. The issue was whether a given image was taken with the exact same camera as other images. For the first time, the camera ID technology has been used in the court of law.

The research results are continuously being incorporated into a software application through a separate project "FIND Camera" funded by the FBI. This application is intended to be used by forensic investigators for fighting crime and collecting intelligence as part of Homeland Security

## PATENTS

The achievements on this project are closely related to two patents that were previously filed. Below are included the patent disclosures.

PROJECT: RB-218 (internal Research Foundation filing number)

TITLE: METHOD AND APPARATUS FOR IDENTIFYING AN IMAGING DEVICE

INVENTORS: Jessica Fridrich, Miroslav Goljan, Jan Lukáš

DESCRIPTION

This invention concerns the problem of authenticating digital images. In particular, the method enables to decisively answer the following questions: Given a digital image and the imaging device that took it, was some specific area of the image tampered? Is there a tampered area in the image and where?

As digital images and video continue to replace their analog counterparts, reliable and inexpensive authentication of digital images increases on importance. It would especially prove useful in the court. For example, the integrity verification could be used for establishing the authenticity of images presented as evidence, or, in a child pornography case, one could prove that certain imagery, or at least its critical part, has been obtained using a specific camera and is not a computer-generated image.

The process of image authentication has been approached from several different directions, but to the best of our knowledge none so far led to a generally usable reliable method. The technology in this disclosure uses as an identification pattern a certain component of the pattern noise of imaging sensors (e.g., CCD or CMOS) caused by pixel non-uniformity. This pattern is extracted using a specially designed denoising filter. The presence of the pattern in a given region of interest is established using a mathematical operation called correlation. This approach is computationally simple and relatively reliable. It is also possible to verify integrity of processed images (e.g., using JPEG or other common processing operations).

TECHNOLOGY APPLICATIONS

1. Discovering digital forgeries
2. Establishing integrity of digital images

TECHNOLOGY ADVANTAGES

1. Reliability and accuracy of this technology unmatched by other competing methods
2. Simplicity and computational efficiency
3. Applicability to all sensor types and image acquisition devices

---

PROJECT: RB-222 (internal Research Foundation filing number)

TITLE: USING PATTERN NOISE OF IMAGING SENSORS FOR IMAGING HARDWARE IDENTIFICATION

INVENTORS: Jessica Fridrich, Miroslav Goljan, Jan Lukáš

DESCRIPTION

This patent concerns the problem of identification of the imaging digital device (e.g., a digital camera or scanner) from a digital image. In particular, the method enables to decisively answer the following questions: Given a digital camera, was a specific image taken with that camera (or scanned with a given scanner)? Did the same camera take two images?

As digital images and video continue to replace their analog counterparts, reliable, inexpensive, and fast identification of digital image origin increases on importance. It would especially prove useful in the court. For example, the identification could be used for establishing the origin of images presented as evidence, or, in a child pornography case, one could prove that certain imagery has been obtained using a specific camera and is not a computer-generated image.

The process of image identification has been approached from several different directions, but to the best of our knowledge none so far led to a reliable method. The technology in this disclosure uses as an identification pattern a certain component of the pattern noise of imaging sensors (e.g., CCD or CMOS) caused by pixel non-uniformity. This pattern is extracted using a specially designed denoising filter. The presence of the pattern in a given image is established using a mathematical operation called correlation. This approach is computationally simple and is able to distinguish between cameras of the exact same model. It is also possible to identify the camera from processed images (e.g., using JPEG or other common processing operations).

TECNOLOGY APPLICATIONS

1. Determining the origin of digital images
2. Matching an image to a camera

TECHNOLOGY ADVANTAGES

1. Reliability and accuracy of this technology unmatched by other competing methods
2. Simplicity and computational efficiency
3. Applicability to all sensor types and image acquisition devices
4. Ability to distinguish between cameras of the exact same brand

POTENTIAL LICENSEES (for both patents)

Law enforcement and forensic analysts, Government contractors and consulting companies, imaging companies and companies manufacturing imaging sensors, companies dealing with data embedding and security, such as Digimarc (http//www.digimarc.com), Verance (http://www.verance.com/), Blue Spike, Inc.

(http://www.bluespike.com), Signum Technologies, (http://www.signumtech.com), or Wetstone, Inc. (http://www.wetstonestech.com).

COMPETING PRODUCTS: None known at this time.

STAGE OF DEVELOPMENT: A working prototype is available for demonstration.

STATUS OF INTELLECTUAL PROPERTY PROTECTION: U.S. patent applications submitted in 2006.

RIGHTS AVAILABLE: Due to the potential broad applicability of the technology, non-exclusive licensing and distribution is anticipated.

CONTACT FOR FURTHER INFORMATION:

Dr. Eugene B. Krentsel
Director, Technology Transfer and Innovation Partnerships
Division of Research
P.O. Box 6000
State University of New York
Binghamton, NY 13902-6000
Phone: 607-777-5871
Fax: 607-777-4354
E-mail: krentsel@binghamton.edu

# Advanced Digital Forensics and Steganalysis Methods

## Executive Summary

Despite its unquestionable advantages, it is highly non-trivial to establish integrity and origin of digitally represented visual data. This issue of trust increases on importance with widespread use of digital imagery for reconnaissance, remote sensing, intelligence gathering, command, control, and communication. Digital images and video are also increasingly more often produced as silent witness in court in connection with child pornography and movie piracy cases, or insurance claims.

The goal of digital forensics is to investigate the origin, integrity, and meaning of evidence in digital form. The fundamental tasks of digital forensic can be clustered into the following six types:

**Source Classification** with the objective to assign a given image to several broad classes based on their origin, such as scan vs. digital camera, or Canon vs. Kodak.

**Device Identification** focuses on proving that a given image was obtained by a specific device that is available (prove that a given camera took a certain image or video).

**Device Linking**, whose task is to group images according to their common source. For example, given a set of images, we would like to find out which images were obtained using the exact same camera.

**Processing History Recovery** with the objective to recover the processing chain applied to a given image. Here, we are interested in non-malicious processing, e.g., lossy compression, filtering, recoloring, contrast/brightness adjustment, etc.

**Integrity Verification** or forgery detection is a procedure aimed at discovering malicious processing, such as object removal or adding.

**Anomaly Investigation** deals with explaining anomalies found in images that may be a product of digital processing or other phenomena specific to digital cameras.

The research presented in this report concerns virtually all of the above forensic tasks. The crucial idea is to use pixel imperfections of digital imaging sensors as a unique fingerprint whose form, integrity, or presence can be used to reach high-certainty conclusions about image processing history, integrity, and origin. The sensor fingerprint is an intrinsic property of all digital imaging sensors due to slight variations among individual pixels in their ability to convert photons to electrons. Consequently, every sensor casts a weak noise-like pattern onto every image it takes. This pattern, or a sensor fingerprint, is essentially an unintentional stochastic spread-spectrum watermark that survives processing, such as lossy compression or filtering. This report explains in detail how this fingerprint can be estimated from images taken by the camera and later detected in a given image to establish image origin and integrity. Extensive experimental evaluation confirms the usability of the proposed methods in practice.

All forensic techniques developed under this project have been peer reviewed and published. The methods were also implemented in Matlab, tested, and made available to the US Government. A forensic software product with all reported methods is currently being developed by PAR, Inc., for use by the FBI and US Air Force. The technology is covered by two US patents.

# 1. MAIN ACHIEVEMENTS

In this section, the investigator summarizes the main research achievements. Some topics are then detailed in individual sections, while the remaining material uncovered in this report is cited with appropriate references.

## 1.1 INTRODUCTION

There exist two types of imaging sensors commonly found in digital cameras, camcorders, and scanners—CCD (Charge-Coupled Device) and CMOS (Complementary Metal-Oxide Semiconductor). Both consist of a large number of photo detectors also called pixels. Pixels are made of silicon and capture light by converting photons into electrons using the photoelectric effect. The accumulated charge is transferred out of the sensor, amplified, and then converted to a digital signal in an AD converter and further processed before the data is stored in an image format, such as JPEG.

The pixels are usually rectangular, several microns across. The amount of electrons generated by the incident light at a pixel depends on the physical dimensions of the pixel photosensitive area and on the homogeneity of silicon. The pixels' physical dimensions slightly vary due to imperfections in the manufacturing process. Also, the inhomogeneity naturally present in silicon contributes to variations in quantum efficiency among pixels (the ability to convert photons to electrons). The differences among pixels can be captured with a matrix **K** of the same dimensions as the sensor. When the imaging sensor is illuminated with ideally uniform light intensity $Y$, in the absence of other noise sources, the sensor would register a noise-like signal $Y+Y\mathbf{K}$ instead. The term $Y\mathbf{K}$ is usually referred to as the pixel-to-pixel non-uniformity or PRNU.

The matrix **K** is responsible for a major part of what is called the camera fingerprint. The fingerprint can be estimated experimentally, for example by taking many images of a uniformly illuminated surface and averaging the images to isolate the systematic component of all images. At the same time, the averaging suppresses random noise components, such as the shot noise (random variations in the number of photons reaching the pixel caused by quantum properties of light) or the readout noise (random noise introduced during the sensor readout), etc [1,2]. Fig. 1 shows a magnified portion of a fingerprint from a 4 megapixel Canon G2 camera obtained by averaging 120 8-bit grayscale images with average grayscale 128 across each image. Bright dots correspond to pixels that consistently generate more electrons, while dark dots mark pixels whose response is consistently lower. The variance in pixel values across the averaged image (before adjusting its range for visualization) was 0.5 or 51 dB. Although the strength of the fingerprint strongly depends on the camera model, the sensor fingerprint is typically quite a weak signal.

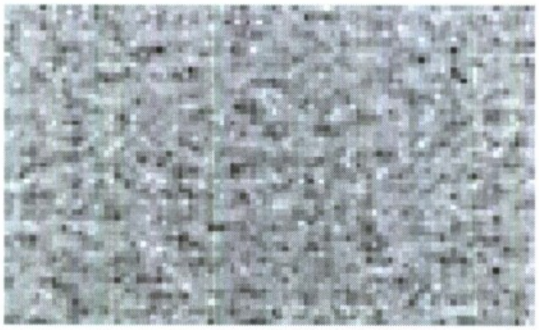


**Fig. 1:** Magnified portion of the sensor fingerprint from Canon G2. The dynamic range was scaled to the interval [0,255] for visualization.

Fig. 2 shows the magnitude of the Fourier transform of one pixel row in the averaged image. The signal resembles white noise with an attenuated high frequency band.

Besides the PRNU, the camera fingerprint essentially contains all systematic defects of the sensor, including hot and dead pixels (pixels that consistently produce high and low output independently of illumination) and the so called dark current (a noise-like pattern that the camera would take with its objective covered). The most important component of the fingerprint is the PRNU. The PRNU term $Y\mathbf{K}$ is only weakly present in dark areas where $Y \approx 0$. Also, completely saturated areas of an image, where the pixels were filled to their full capacity, producing a constant signal, do not carry any traces of PRNU or any other noise for that matter.

It should be noted that essentially all imaging sensors (CCD, CMOS, JFET, or CMOS-Foveon™ X3) are built from semiconductors and their manufacturing techniques are similar. Therefore, these sensors will likely exhibit fingerprints with similar properties.

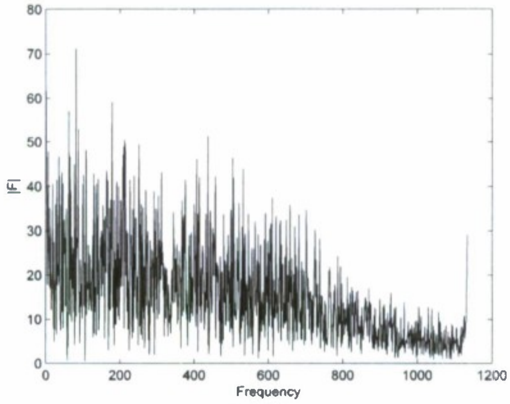


**Fig. 2:** Magnitude of Fourier transform of one row of the sensor fingerprint.

Even though the PRNU term is stochastic in nature, it is a relatively stable component of the sensor over its life span. The factor **K** is thus a very useful forensic quantity responsible for a unique sensor fingerprint with the following important properties:

1. **Dimensionality**. The fingerprint is stochastic in nature and has a large information content, which makes it unique to each sensor.

2. **Universality**. All imaging sensors exhibit PRNU.

3. **Generality**. The fingerprint is present in every picture independently of the camera optics, camera settings, or scene content, with the exception of completely dark images.

4. **Stability**. It is stable in time and under wide range of environmental conditions (temperature, humidity).

5. **Robustness**. It survives lossy compression, filtering, gamma correction, and many other typical processing.

The fingerprint can be used for many forensic tasks:

• By testing the *presence* of a specific fingerprint in the image, one can achieve reliable device identification (e.g., prove that a certain camera took a given image) or prove that two images were taken by the same device (device linking). The presence of camera fingerprint in an image is also indicative of the fact that the image under investigation is natural and not a computer rendering.

• By establishing the *absence* of the fingerprint in individual image regions, it is possible to discover maliciously replaced parts of the image. This task pertains to integrity verification.

• By detecting the *strength or form* of the fingerprint, it is possible to reconstruct some of the processing history. For example, one can use the fingerprint as a template to estimate geometrical processing, such as scaling, cropping, or rotation. Non-geometrical operations are also going to influence the strength of the fingerprint in the image and thus can be potentially detected.

• The spectral and spatial characteristics of the fingerprint can be used to identify the camera model or distinguish between a scan and a digital camera image (the scan will exhibit spatial anisotropy).

This section is organized as follows. In Section 1.2, the author describes a simplified sensor output model and uses it to derive a maximum likelihood estimator for the fingerprint. At the same time, the author points out the need to preprocess the estimated signal to remove certain systematic patterns that might increase false alarms in device identification and missed detections when using the fingerprint for image integrity verification. Starting again with the sensor model in Section 1.3, the task of detecting the PRNU is formulated as a two-channel problem and approached using the generalized likelihood ratio test in Neyman-Pearson setting. First, the detector for device identification is derived and then adapted for device linking and fingerprint matching. Section 1.4 shows how the fingerprint can be used for integrity verification by detecting the fingerprint in individual image blocks. The reliability of camera identification and forgery detection using sensor fingerprint is illustrated on real imagery in Section 1.5.

Everywhere in this report, boldface font will denote vectors (or matrices) of length specified in the text, e.g., **X** and **Y** are vectors of length $n$ and $\mathbf{X}[i]$ denotes the $i$th component of **X**. Sometimes, pixels will be indexed using a two-dimensional index formed by the row and column index. Unless mentioned otherwise, all operations among vectors or matrices, such as product, ratio, raising to a power, etc., are *elementwise*. The dot product of vectors is denoted as $\mathbf{X}\Box\,\mathbf{Y}=\sum_{i=1}^{n}\mathbf{X}[i]\mathbf{Y}[i]$ with $\|\mathbf{X}\|=\sqrt{\mathbf{X}\Box\,\mathbf{X}}$ being the $L_2$ norm of **X**. Denoting the sample mean with a bar, the normalized correlation is

$$corr(\mathbf{X},\mathbf{Y})=\frac{(\mathbf{X}-\bar{\mathbf{X}})\Box\,(\mathbf{Y}-\bar{\mathbf{Y}})}{\|\mathbf{X}-\bar{\mathbf{X}}\|\cdot\|\mathbf{Y}-\bar{\mathbf{Y}}\|}.$$

## 1.2 SENSOR FINGERPRINT ESTIMATION

The PRNU is injected into the image during acquisition before the signal is quantized or processed in any other manner. In order to derive an estimator of the fingerprint, we need to formulate a model of the sensor output.

### 1.2.1 Sensor Output Model

Even though the process of acquiring a digital image is quite complex and varies greatly across different camera models, some basic elements are common to most cameras. The light cast by the camera optics is projected onto the pixel grid of the imaging sensor. The charge generated through interaction of photons with silicon is amplified and quantized. Then, the signal from each color channel is adjusted for gain (scaled) to achieve proper white balance. Because most sensors cannot register color, the pixels are typically equipped with a color filter that lets only light of one specific color (red, green, or blue) enter the pixel. The array of filters is called the color filter array (CFA). To obtain a color image, the signal is interpolated or demosaicked. Finally, the colors are further adjusted to correctly display on a computer monitor through color correction and gamma correction. Cameras may also employ filtering, such as denoising or sharpening. At the very end of this processing chain, the image is stored in the JPEG or some other format, which may involve quantization.

Let us denote by $\mathbf{I}[i]$ the quantized signal registered at pixel $i$, $i = 1, \ldots, m\times n$, before demosaicking. Here, $m\times n$ are image dimensions. Let $\mathbf{Y}[i]$ be the incident light intensity at pixel $i$. Dropping the pixel indices for better readability, the following vector form of the sensor output model is used

$$\mathbf{I}=g^{\gamma}\cdot\left[(1+\mathbf{K})\mathbf{Y}+\mathbf{\Omega}\right]^{\gamma}+\mathbf{Q}. \qquad (1)$$

All operations in (1) (and everywhere else in this report) are element-wise. In (1), $g$ is the gain factor (different for each color channel) and $\gamma$ is the gamma correction factor (typically, $\gamma\approx 0.45$). The matrix **K** is a zero-mean noise-like signal responsible for the PRNU (the sensor fingerprint). Denoted by $\mathbf{\Omega}$ is a combination of the other noise sources, such as the dark current, shot noise, and read-out noise [2]; **Q** is the combined distortion due to quantization and/or JPEG compression.

In parts of the image that are not dark, the dominant term in the square bracket in (1) is the scene light intensity **Y**. By factoring it out and keeping the first two terms in the Taylor expansion of $(1 + x)^{\gamma} = 1 + \gamma x + O(x^2)$ at $x = 0$, one obtains

$$\begin{aligned} \mathbf{I} = (g\mathbf{Y})^{\gamma} \cdot [\mathbf{1} + \mathbf{K} + \mathbf{\Omega}/\mathbf{Y}]^{\gamma} + \mathbf{Q} \square \\ (g\mathbf{Y})^{\gamma} \cdot (\mathbf{1} + \gamma\mathbf{K} + \gamma\mathbf{\Omega}/\mathbf{Y}) + \mathbf{Q} = \mathbf{I}^{(0)} + \mathbf{I}^{(0)}\mathbf{K} + \mathbf{\Theta}. \end{aligned} \tag{2}$$

In (2), $\mathbf{I}^{(0)} = (g\mathbf{Y})^{\gamma}$ denotes the ideal sensor output in the absence of any noise or imperfections. Note that $\mathbf{I}^{(0)}\mathbf{K}$ is the PRNU term and $\mathbf{\Theta} = \gamma\mathbf{I}^{(0)}\mathbf{\Lambda}/\mathbf{Y} + \mathbf{\Theta}_q$ is the modeling noise. In the last expression in (2), the scalar factor $\gamma$ was absorbed into the PRNU factor **K** to simplify the notation.

### 1.2.2 Sensor Fingerprint Estimation

The sensor output model is now used to derive an estimator of the PRNU factor **K**. A good introductory text on signal estimation and detection is [3,4].

The SNR between the signal of interest $\mathbf{I}^{(0)}\mathbf{K}$ and observed data **I** can be improved by suppressing the noiseless image $\mathbf{I}^{(0)}$ by subtracting from both sides of (2) a denoised version of **I**, $\hat{\mathbf{I}}^{(0)} = F(\mathbf{I})$, obtained using a denoising filter $F$ (Section 1.6 describes the filter used in all experiments in this report):

$$\begin{aligned} \mathbf{W} = \mathbf{I} - \hat{\mathbf{I}}^{(0)} = \mathbf{IK} + \mathbf{I}^{(0)} - \hat{\mathbf{I}}^{(0)} + (\mathbf{I}^{(0)} - \mathbf{I})\mathbf{K} + \mathbf{\Theta} \\ = \mathbf{IK} + \mathbf{\Xi}\ . \end{aligned} \tag{3}$$

It is easier to estimate the PRNU term from **W** than from **I** because the filter suppresses the image content. Here, $\mathbf{\Xi}$ is the sum of $\mathbf{\Theta}$ and two additional terms introduced by the denoising filter.

It will be assumed that a database of $d \geq 1$ images, $\mathbf{I}_1, \ldots, \mathbf{I}_d$, obtained by the camera, is available. For each pixel $i$, the sequence $\mathbf{\Xi}_1[i], \ldots, \mathbf{\Xi}_d[i]$ is modeled as white Gaussian noise (WGN) with variance $\sigma^2$. Even though the noise term is technically not independent of the PRNU signal **IK** due to the term $(\mathbf{I}^{(0)} - \mathbf{I})\mathbf{K}$, because the energy of this term is small compared to **IK**, the assumption that $\mathbf{\Xi}$ is independent of **IK** is reasonable.

From (3), one can write for each $k = 1, \ldots, d$

$$\frac{\mathbf{W}_k}{\mathbf{I}_k} = \mathbf{K} + \frac{\mathbf{\Xi}_k}{\mathbf{I}_k}, \quad \mathbf{W}_k = \mathbf{I}_k - \hat{\mathbf{I}}_k^{(0)}, \ \hat{\mathbf{I}}_k^{(0)} = F(\mathbf{I}_k)\,. \tag{4}$$

Under the assumption about the noise term, the log-likelihood of observing $\mathbf{W}_k/\mathbf{I}_k$ given **K** is

$$L(\mathbf{K}) = -\frac{d}{2}\sum_{k=1}^{d} \log(2\pi\sigma^2/(\mathbf{I}_k)^2) - \sum_{k=1}^{d} \frac{(\mathbf{W}_k/\mathbf{I}_k - \mathbf{K})^2}{2\sigma^2/(\mathbf{I}_k)^2}\,. \tag{5}$$

By taking partial derivatives of (5) with respect to individual elements of **K** and solving for **K**, one obtains the maximum likelihood estimate $\hat{\mathbf{K}}$

$$\frac{\partial L(\mathbf{K})}{\partial \mathbf{K}} = \sum_{k=1}^{d} \frac{\mathbf{W}_k/\mathbf{I}_k - \mathbf{K}}{\sigma^2/(\mathbf{I}_k)^2} = 0 \Rightarrow \hat{\mathbf{K}} = \frac{\sum_{k=1}^{d} \mathbf{W}_k\mathbf{I}_k}{\sum_{k=1}^{d} (\mathbf{I}_k)^2}\,. \tag{6}$$

The Cramer-Rao Lower Bound (CRLB) gives the bound on the variance of $\hat{\mathbf{K}}$

$$\frac{\partial^2 L(\mathbf{K})}{\partial \mathbf{K}^2} = -\frac{\sum_{k=1}^{d}(\mathbf{I}_k)^2}{\sigma^2} \Rightarrow var(\hat{\mathbf{K}}) \ge \frac{1}{-E\left[\frac{\partial^2 L(\mathbf{K})}{\partial \mathbf{K}^2}\right]} = \frac{\sigma^2}{\sum_{k=1}^{d}(\mathbf{I}_k)^2}. \tag{7}$$

Because the sensor model (3) is linear, the CRLB says that the maximum likelihood estimator is minimum variance unbiased and its variance $\mathrm{var}(\hat{\mathbf{K}}) \sim 1/d$. From (7), one can see that the best images for estimation of $\mathbf{K}$ are those with high luminance (but not saturated) and small $\sigma^2$ (which means smooth content). If the camera under investigation is in our possession, out-of-focus images of bright cloudy sky would be the best. In practice, good estimates of the fingerprint may be obtained from 20–50 natural images depending on the camera. If sky images are used instead of natural images, only approximately one half of them would be enough to obtain an estimate of the same accuracy.

The estimate $\hat{\mathbf{K}}$ contains all components that are *systematically present* in every image, including artifacts introduced by color interpolation, JPEG compression, on-sensor signal transfer [5], and sensor design. While the PRNU is unique to the sensor, the other artifacts are shared among cameras of the same model or sensor design. Consequently, PRNU factors estimated from two different cameras may be slightly correlated, which undesirably increases the false identification rate. Fortunately, the artifacts manifest themselves mainly as periodic signals in row and column averages of $\hat{\mathbf{K}}$ and can be suppressed simply by subtracting the averages from each row and column. For a PRNU estimate $\hat{\mathbf{K}}$ with $m$ rows and $n$ columns, the processing is described using the following pseudo-code

```
$r_i = 1/n\sum_{j=1}^{n}\hat{\mathbf{K}}[i,j]$
for $i$ = 1 to $m$ { $\hat{\mathbf{K}}'[i,j] = \hat{\mathbf{K}}[i,j] - r_i$ for $j$ = 1, …, $n$}
$c_j = 1/m\sum_{i=1}^{m}\hat{\mathbf{K}}'[i,j]$
for $j$ = 1 to $n$ { $\hat{\mathbf{K}}''[i,j] = \hat{\mathbf{K}}'[i,j] - c_j$ for $i$ = 1, …, $m$}.
```

The difference $\hat{\mathbf{K}} - \hat{\mathbf{K}}''$ is called the linear pattern (see Fig. 3) and it is a useful forensic entity by itself – it can be used to classify a camera fingerprint to a camera model or brand. More details of this preprocessing step are contained in [6,28].

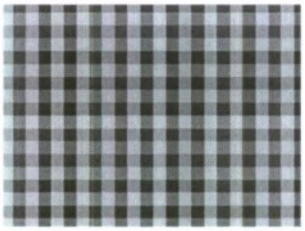


**Fig. 3:** Detail of the linear pattern for Canon S40.

To avoid cluttering the text with too many symbols, in the rest of this report, the *processed* fingerprint $\hat{\mathbf{K}}''$ will be denoted with the same symbol $\hat{\mathbf{K}}$.

For color images, the PRNU factor can be estimated for each color channel separately, obtaining thus three fingerprints of the same dimensions $\hat{\mathbf{K}}_R$, $\hat{\mathbf{K}}_G$, and $\hat{\mathbf{K}}_B$. Since these three fingerprints are highly correlated due to in-camera processing, in all forensic methods in this report, before analyzing a color image under investigation it is converted to grayscale and correspondingly the three fingerprints are combined into one fingerprint using the usual conversion from RGB to grayscale

$$\hat{\mathbf{K}} = 0.2989\hat{\mathbf{K}}_R + 0.587\hat{\mathbf{K}}_G + 0.114\hat{\mathbf{K}}_B. \tag{8}$$

## 1.3 CAMERA IDENTIFICATION USING SENSOR FINGERPRINT

This section introduces general methodology for determining the origin of images or video using sensor fingerprint. The author starts with what is generally considered as the most frequently occurring situation in practice, which is **camera identification** from images. Here, the task is to determine if an image under investigation was taken with a given camera. This is achieved by testing whether the image noise residual contains the camera fingerprint. Anticipating the next two closely related forensic tasks, the author formulates the hypothesis testing problem for camera identification in a setting that is general enough to essentially cover the remaining tasks, which are **device linking** and **fingerprint matching**. In device linking, two images are tested if they came from the same camera (the camera itself may not be available). The task of matching two estimated fingerprints occurs in matching two video-clips because individual video frames from each clip can be used as a sequence of images from which an estimate of the camcorder fingerprint can be obtained (here, again, the cameras/camcorders may not be available to the analyst).

### 1.3.1 Device identification

A general scenario will be considered here, in which the image under investigation has possibly undergone a geometrical transformation, such as scaling or rotation. Let us assume that before applying any geometrical transformation the image was in grayscale represented with an $m\times n$ matrix $\mathbf{I}[i, j]$, $i = 1, \ldots, m$, $j = 1, \ldots, n$. Let us denote as $\mathbf{u}$ the (unknown) vector of parameters describing the geometrical transformation, $T_{\mathbf{u}}$. For example, $\mathbf{u}$ could be a scaling ratio or a two-dimensional vector consisting of the scaling parameter and unknown angle of rotation. In device identification, we wish to determine whether or not the transformed image

$$\mathbf{Z} = T_{\mathbf{u}}(\mathbf{I})$$

was taken with a camera with a known fingerprint estimate $\hat{\mathbf{K}}$. Thus, one can assume that the geometrical transformation is downgrading (such as downsampling) and thus it will be more advantageous to match the inverse transform $T_{\mathbf{u}}^{-1}(\mathbf{Z})$ with the fingerprint rather than matching $\mathbf{Z}$ with a downgraded version of $\hat{\mathbf{K}}$.

The detection problem will now be formulated in a slightly more general form to cover all three forensic tasks mentioned above within one framework. The fingerprint detection is the following two-channel hypothesis testing problem

$$\begin{aligned} \mathrm{H}_0 &: \mathbf{K}_1 \neq \mathbf{K}_2 \\ \mathrm{H}_1 &: \mathbf{K}_1 = \mathbf{K}_2 \end{aligned} \tag{9}$$

where

$$\begin{aligned} \mathbf{W}_1 &= \mathbf{I}_1\mathbf{K}_1 + \Xi_1 \\ T_{\mathbf{u}}^{-1}(\mathbf{W}_2) &= T_{\mathbf{u}}^{-1}(\mathbf{Z})\mathbf{K}_2 + \Xi_2. \end{aligned} \tag{10}$$

In (10), all signals are observed with the exception of the noise terms $\Xi_1$, $\Xi_2$ and the fingerprints $\mathbf{K}_1$ and $\mathbf{K}_2$. Specifically, for the device identification problem, $\mathbf{I}_1 \equiv 1$, $\mathbf{W}_1 = \hat{\mathbf{K}}$ estimated in the previous section, and $\Xi_1$ is the estimation error of the PRNU. $\mathbf{K}_2$ is the PRNU from the camera that took the image, $\mathbf{W}_2$ is the geometrically transformed noise residual, and $\Xi_2$ is a noise term. In general, $\mathbf{u}$ is an unknown parameter. Note that since $T_{\mathbf{u}}^{-1}(\mathbf{W}_2)$ and $\mathbf{W}_1$ may have different dimensions, the formulation (10) involves an unknown spatial shift between both signals, s.

Modeling the noise terms $\Xi_1$ and $\Xi_2$ as white Gaussian noise with known variances $\sigma_1^2, \sigma_2^2$, the generalized likelihood ratio test for this two-channel problem was derived in [7]. The test statistics is a sum of three terms: two energy-like quantities and a cross-correlation term

$$t = \max_{\mathbf{u},s}\{E_1(\mathbf{u},\mathrm{s}) + E_2(\mathbf{u},\mathrm{s}) + C(\mathbf{u},\mathrm{s})\}\,, \tag{11}$$

$$E_1(\mathbf{u},\mathbf{s}) = \sum_{i,j} \frac{\mathbf{I}_1^2[i,j](\mathbf{W}_1[i+s_1,j+s_2])^2}{\sigma_1^2\mathbf{I}_1^2[i,j] + \sigma_1^4\sigma_2^{-2}(T_{\mathbf{u}}^{-1}(\mathbf{Z})[i+s_1,j+s_2])^2}$$

$$E_2(\mathbf{u},\mathbf{s}) = \sum_{i,j} \frac{(T_{\mathbf{u}}^{-1}(\mathbf{Z})[i+s_1,j+s_2])^2(T_{\mathbf{u}}^{-1}(\mathbf{W}_2)[i+s_1,j+s_2])^2}{\sigma_2^2(T_{\mathbf{u}}^{-1}(\mathbf{Z})[i+s_1,j+s_2])^2 + \sigma_2^4\sigma_1^{-2}\mathbf{I}_1^2[i,j]}$$

$$C(\mathbf{u},\mathbf{s}) = \sum_{i,j} \frac{\mathbf{I}_1[i,j]\mathbf{W}_1[i,j](T_{\mathbf{u}}^{-1}(\mathbf{Z})[i+s_1,j+s_2])(T_{\mathbf{u}}^{-1}(\mathbf{W}_2)[i+s_1,j+s_2])}{\sigma_2^2\mathbf{I}_1^2[i,j] + \sigma_1^2(T_{\mathbf{u}}^{-1}(\mathbf{Z})[i+s_1,j+s_2])^2}\,.$$

The complexity of evaluating these three expressions is proportional to the square of the number of pixels, $(m\times n)^2$, which makes this detector unusable in practice. Thus, this detector is simplified to a normalized cross-correlation (NCC) that can be evaluated using fast Fourier transform. Under $\mathrm{H}_1$, the maximum in (11) is mainly due to the contribution of the cross-correlation term, $C(\mathbf{u}, \mathbf{s})$, that exhibits a sharp peak for the proper values of the geometrical transformation. Thus, a much faster suboptimal detector is the NCC between **X** and **Y** maximized over all shifts $s_1$, $s_2$, and **u**

$$\mathbf{NCC}[s_1,s_2;\mathbf{u}] = \frac{\sum_{k=1}^{m}\sum_{l=1}^{n}\left(\mathbf{X}[k,l]-\bar{\mathbf{X}}\right)\left(\mathbf{Y}[k+s_1,l+s_2]-\bar{\mathbf{Y}}\right)}{\left\|\mathbf{X}-\bar{\mathbf{X}}\right\|\left\|\mathbf{Y}-\bar{\mathbf{Y}}\right\|}\,, \tag{12}$$

which we view as an $m\times n$ matrix parameterized by **u**, where

$$\mathbf{X} = \frac{\mathbf{I}_1\mathbf{W}_1}{\sqrt{\sigma_2^2\mathbf{I}_1^2 + \sigma_1^2\left(T_{\mathbf{u}}^{-1}(\mathbf{Z})\right)^2}}\,,\quad \mathbf{Y} = \frac{T_{\mathbf{u}}^{-1}(\mathbf{Z})T_{\mathbf{u}}^{-1}(\mathbf{W}_2)}{\sqrt{\sigma_2^2\mathbf{I}_1^2 + \sigma_1^2\left(T_{\mathbf{u}}^{-1}(\mathbf{Z})\right)^2}}\,. \tag{13}$$

A more stable detection statistics, whose meaning will become apparent from error analysis later in this section, that is strongly advocated to use for all camera identification tasks, is the Peak to Correlation Energy measure (PCE) defined as

$$\mathrm{PCE}(\mathbf{u}) = \frac{\mathbf{NCC}[\mathrm{s}_{\mathrm{peak}};\mathbf{u}]^2}{\frac{1}{mn-|\mathcal{N}|}\sum_{\mathbf{s},\mathbf{s}\notin\mathcal{N}}\mathbf{NCC}[\mathbf{s};\mathbf{u}]^2}\,, \tag{14}$$

where for each fixed **u**, $\mathcal{N}$ is a small region surrounding the peak value of NCC $\mathbf{s}_{\mathrm{peak}}$ across all shifts $s_1$, $s_2$.

For device identification from a single image, the fingerprint estimation noise $\Xi_1$ is much weaker compared to $\Xi_2$ for the noise residual of the image under investigation. Thus, $\sigma_1^2 = \mathrm{var}(\Xi_1)\,\Box\,\ \mathrm{var}(\Xi_2) = \sigma_2^2$ and (12) can be further simplified to a NCC between

$$\mathbf{X} = \mathbf{W}_1 = \hat{\mathbf{K}} \ \text{ and } \ \mathbf{Y} = T_{\mathbf{u}}^{-1}(\mathbf{Z})T_{\mathbf{u}}^{-1}(\mathbf{W}_2)\,.$$

Recall that $\mathbf{I}_1 = 1$ for device identification when its fingerprint is known.

In practice, the maximum PCE value can be found by a search on a grid obtained by discretizing the range of **u**. Because the statistics is noise-like for incorrect values of **u** and only exhibits a sharp peak in a small neighborhood of the correct value of **u**, unfortunately, gradient methods do not apply and one is left with a potentially expensive grid search. The grid has to be sufficiently dense in order not to miss the peak. As an example, the author now provides additional details how one can carry out the search when $\mathbf{u} = r$ is an unknown scaling ratio. More details are given in Section 2.

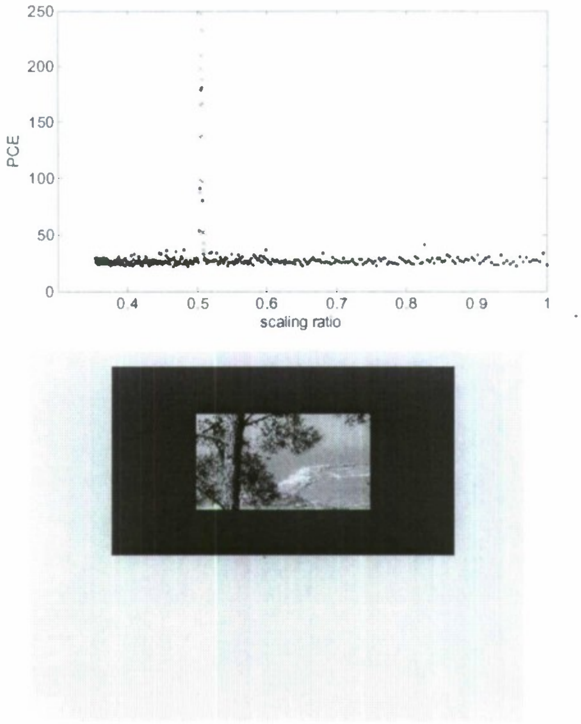


**Fig. 4:** Top: Detected peak in PCE($r_i$). Bottom: Visual representation of the detected cropping and scaling parameters $r_{\text{peak}}$, $s_{\text{peak}}$. The gray frame shows the original image size, while the black frame shows the image size after cropping before resizing.

Assuming the image under investigation has dimensions $M \times N$, one searches for the scaling parameter at discrete values $r_i \leq 1$, $i = 0, 1, \ldots, R$, from $r_0 = 1$ (no scaling, just cropping) down to $r_{\min} = \max\{M/m, N/n\} < 1$

$$r_i = \frac{1}{1+0.005i}, \quad i = 0,1,2,\ldots. \tag{15}$$

For a fixed scaling parameter $r_i$, the cross-correlation (12) does not have to be computed for all shifts **s** but only for those that move the upsampled image $T_{r_i}^{-1}(\mathbf{Z})$ *within* the dimensions of $\hat{\mathbf{K}}$ because only such shifts can be generated by cropping. Given that the dimensions of the upsampled image $T_{r_i}^{-1}(\mathbf{Z})$ are $M/r_i \times N/r_i$, one has the following range for the spatial shift $\mathbf{s} = (s_1, s_2)$

$$0 \leq s_1 \leq m - M/r_i \text{ and } 0 \leq s_2 \leq n - N/r_i. \tag{16}$$

The peak of the two-dimensional NCC across all spatial shifts **s** is evaluated for each $r_i$ using PCE($r_i$) (14). If $\max_i$ PCE($r_i$) > $\tau$, the decision is $\text{H}_1$ (camera and image are matched). Moreover, the value of the scaling parameter at which the PCE attains this maximum determines the scaling ratio $r_{\text{peak}}$. The location of the peak $\mathbf{s}_{\text{peak}}$ in the normalized cross-correlation determines the cropping parameters. Thus, as a by-product of this algorithm, one can determine the processing history of the image under investigation (see Fig. 4). The fingerprint can thus play the role of a synchronizing template similar to templates used in digital watermarking. It can also be used for reverse-engineering in-camera processing, such as digital zoom [9].

In any forensic application, it is important to keep the false alarm rate low. For camera identification tasks, this means that the probability, $P_{\text{FA}}$, that a camera that did not take the image is falsely identified must be below a certain user-defined threshold (Neyman-Pearson setting). Thus, it is necessary to obtain a relationship between $P_{\text{FA}}$ and the threshold on the PCE. Note that the threshold will depend on the size of the search space, which is in turn determined by the dimensions of the image under investigation.

Under hypothesis $\text{H}_0$ for a fixed scaling ratio $r_i$, the values of the normalized cross-correlation $\mathbf{NCC}[\mathbf{s}; r_i]$ as a function of s are well-modeled as white Gaussian noise $\zeta^{(i)} \sim N(0, \sigma_i^2)$ (see Fig. 5) with variance that may depend on $i$. Estimating the variance of the Gaussian model using the sample variance $\hat{\sigma}_i^2$ of **NCC**[**s**; $r_i$] over **s** after excluding a small central region $\mathcal{N}$ surrounding the peak

$$\hat{\sigma}_i^2 = \frac{1}{mn - |\mathcal{N}|} \sum_{\mathbf{s}, \mathbf{s} \notin \mathcal{N}} \mathbf{NCC}[\mathbf{s}; r_i]^2 , \tag{17}$$

one can now calculate the probability $p_i$ that $\zeta^{(i)}$ would attain the peak value **NCC**[$\mathbf{s}_{\text{peak}}$; $r_{\text{peak}}$] or larger by chance:

$$p_i = \int_{\mathrm{NCC}[\mathbf{s}_{\text{peak}}; r_{\text{peak}}]}^{\infty} \frac{1}{\sqrt{2\pi}\hat{\sigma}_i} e^{-\frac{x^2}{2\hat{\sigma}_i^2}} dx = \int_{\hat{\sigma}_{\text{peak}}\sqrt{\mathrm{PCE}_{\text{peak}}}}^{\infty} \frac{1}{\sqrt{2\pi}\hat{\sigma}_i} e^{-\frac{x^2}{2\hat{\sigma}_i^2}} dx = Q\left(\frac{\hat{\sigma}_{\text{peak}}}{\hat{\sigma}_i}\sqrt{\mathrm{PCE}_{\text{peak}}}\right),$$

where $Q(x) = 1 - \Phi(x)$ with $\Phi(x)$ denoting the cumulative distribution function of a standard normal variable $N(0,1)$ and $\mathrm{PCE}_{\text{peak}} = \mathrm{PCE}(r_{\text{peak}})$. As explained above, during the search for the cropping vector **s**, one only needs to search in the range (16), which means that the maximum is taken over $k_i = (m - M/r_i + 1)\times(n - N/r_i + 1)$ samples of $\zeta^{(i)}$. Thus, the probability that the maximum value of $\zeta^{(i)}$ would not exceed **NCC**[$\mathbf{s}_{\text{peak}}$; $r_{\text{peak}}$] is $(1-p_i)^{k_i}$. After $R$ steps in the search, the probability of false alarm is

$$P_{\text{FA}} = 1 - \prod_{i=1}^{R} (1-p_i)^{k_i} . \tag{18}$$

Since the search can be stopped after the PCE reaches a certain threshold, it must be $r_i \le r_{\text{peak}}$. Because $\hat{\sigma}_i^2$ is non-decreasing in $i$, $\hat{\sigma}_{\text{peak}} / \hat{\sigma}_i \ge 1$. Because $Q(x)$ is decreasing, $p_i \le Q\left(\sqrt{\mathrm{PCE}_{\text{peak}}}\right) = p$. Thus, because $k_i \le mn$, one obtains an upper bound on $P_{\text{FA}}$

$$P_{\text{FA}} \le 1 - (1-p)^{k_{\max}} , \tag{19}$$

where $k_{\max} = \sum_{i=0}^{R-1} k_i$ is the maximal number of values of the parameters $r$ and **s** over which the maximum of (11) could be taken. Equation (19), together with $p = Q\left(\sqrt{\tau}\right)$, determines the threshold for PCE, $\tau = \tau$ ($P_{\text{FA}}$, $M$, $N$, $m$, $n$).

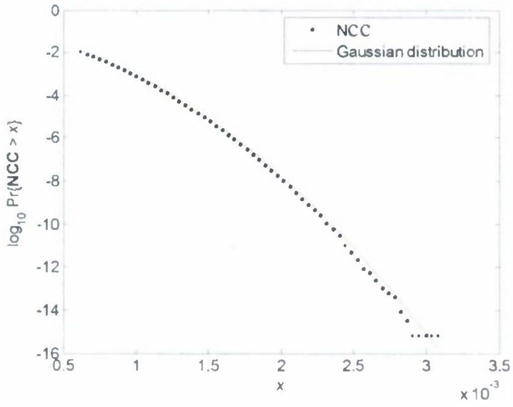


**Fig. 5:** Log-tail plot for the right tail of the sample distribution of **NCC**[s; $r_i$] for an unmatched case.

This finishes the technical formulation and solution of the camera identification algorithm from a single image if the camera fingerprint is known. To demonstrate how reliable this algorithm is, Section 1.5 shows the results of experiments on real images. This algorithm can be used with small modifications for the other two forensic tasks formulated in the beginning of this section, which are device linking and fingerprint matching.

**Pseudo-code for camera identification from cropped and scaled images**

```
1.  Read      True color image $\mathbf{Z}$, with $M$ rows and $N$ columns of pixels.
              $\mathrm{PCE}_{\mathrm{peak}} = 0$; $\hat{\mathbf{K}}$ is the estimated PRNU with $m$ rows and $n$ columns.
2.  Set       $r_{\min} = \max\{M/m, N/n\}$
              $\mathbf{r} = (r_0, r_1, r_2, \ldots, r_{\min})$, $r_i = \frac{1}{1+0.005i}$, $i = 0,1,2,\ldots,R-1$;
              $\tau$      {detection threshold for PCE for a given $P_{\mathrm{FA}}$ (see Equation (18) in Section
      1.3.1)}
3.  Extract noise $\mathbf{W}$ from $\mathbf{Z}$ in each color channel and combine the matrices using the linear transform (8):
              $\mathbf{W} = 0.2989 \times \mathbf{W}_{\mathrm{R}} + 0.5870 \times \mathbf{W}_{\mathrm{G}} + 0.1140 \times \mathbf{W}_{\mathrm{B}}$.
4.  Convert $\mathbf{Z}$ to grayscale.
5.  For       $i = (R-3, R-2, R-1, 0, 1, \ldots, R-4)$
    begin{phase 1}
        6.    Up-sample noise $\mathbf{W}$ by factor $1/r_i$ to obtain $T_{1/r_i}(\mathbf{W})$ (nearest neighbor algorithm used).
        7.    Calculate the NCC matrix (12) with $\mathbf{X} = \hat{\mathbf{K}}$ and $\mathbf{Y} = T_{1/r_i}(\mathbf{Z})T_{1/r_i}(\mathbf{W})$.
        8.    Obtain PCE($r_i$) according to (14).
        9.    If PCE($r_i$) > $\mathrm{PCE}_{\mathrm{peak}}$ , then $\mathrm{PCE}_{\mathrm{peak}}$ = PCE($r_i$); $j = i$;
              elseif $\mathrm{PCE}_{\mathrm{peak}} > \tau$ then go to Step 10.
    end{phase 1}

10. Set       $r_{\mathrm{step}} = 1/\max\{m, n\}$; $\mathbf{r}' = (r_{j-1}-r_{\mathrm{step}}, r_{j-1}-2r_{\mathrm{step}}, \ldots, r_{j+1}+r_{\mathrm{step}}) = (r'_1, r'_2, \ldots, r'_{R'})$;
11. For       $i = (1, \ldots, R')$
    begin{phase 2}
        12.   Up-sample noise $\mathbf{W}$ by factor $1/r'_i$ to obtain $T_{1/r'_i}(\mathbf{W})$.
        13.   Calculate the NCC matrix (12) with $\mathbf{X} = \hat{\mathbf{K}}$ and $\mathbf{Y} = T_{1/r'_i}(\mathbf{Z})T_{1/r'_i}(\mathbf{W})$.
        14.   Obtain PCE($r'_i$) according to (14).
        15.   If PCE($r'_i$) > $\mathrm{PCE}_{\mathrm{peak}}$ , then $\mathrm{PCE}_{\mathrm{peak}}$ = PCE($r'_i$); $r_{\mathrm{peak}} = r'_i$;
    end{phase 2}
16. If        $\mathrm{PCE}_{\mathrm{peak}} > \tau$,
        then  begin
                  Declare the image source being identified.
                  Locate the maximum (= $\mathrm{PCE}_{\mathrm{peak}}$) in NCC to determine
                  the cropping parameters $(u_l, u_t) = \mathbf{s}_{\mathrm{peak}}$.
                  Output $\mathbf{s}_{\mathrm{peak}}$, $r_{\mathrm{peak}}$.
              end
        else  Declare the image source unknown.
    end
```

### 1.3.2 Device linking

The detector derived in the previous section can be readily used with only a few changes for device linking or determining whether two images, $\mathbf{I}_1$ and $\mathbf{Z}$, were taken by the exact same camera [11]. Note that in this problem the camera or its fingerprint is not necessarily available.

The device linking problem corresponds exactly to the two-channel formulation (9) and (10) with the GLRT detector (11). Its faster, suboptimal version is the PCE (14) obtained from the maximum value of $\mathbf{NCC}[s_1, s_2; \mathbf{u}]$ over all $s_1, s_2; \mathbf{u}$ (see (12) and (13)). In contrary to the camera identification problem, now the power of both noise terms, $\Xi_1$ and $\Xi_2$, is comparable and needs to be estimated from observations. Fortunately, because the PRNU term **IK** is much weaker than the modeling noise $\Xi$, reasonable estimates of the noise variances are simply $\hat{\sigma}_1^2 = \mathrm{var}(\mathbf{W}_1)$, $\hat{\sigma}_2^2 = \mathrm{var}(\mathbf{W}_2)$.

Unlike in the camera identification problem, the search for unknown scaling must now be enlarged to scalings $r_i > 1$ (upsampling) because the combined effect of unknown cropping and scaling for both images prevents us from easily identifying which image has been downscaled with respect to the other one. The error analysis carries over from Section 1.3.1. Experimental verification of the device linking algorithm appears in Section 3 and in the original publication [11].

### 1.3.3 Matching fingerprints

The third, fingerprint matching scenario corresponds to the situation when one desires to decide whether or not two estimates of potentially two different fingerprints are identical. This happens, for example, in video-clip linking because the fingerprint can be estimated from all frames forming the clip [12].

The detector derived in Section III.A applies to this scenario, as well. It can be further simplified because for matching fingerprints, $\mathbf{I}_1 = \mathbf{Z} = 1$ and (12) simply becomes the normalized cross-correlation between $\mathbf{X} = \hat{\mathbf{K}}_1$ and $\mathbf{Y} = T_{\mathbf{u}}^{-1}(\hat{\mathbf{K}}_2)$. Experimental verification of the fingerprint matching algorithm for video clips is in Section 4 and in the original publication [12].

## 1.4 FORGERY DETECTION USING CAMERA FINGERPRINT

Another important use of the sensor fingerprint is verification of image integrity. Certain types of tampering can be identified by detecting the fingerprint presence in smaller regions. The assumption is that if a region was copied from another part of the image (or an entirely different image), it will not have the correct fingerprint on it. Some malicious changes in the image may preserve the PRNU and will not be detected using this approach. A good example is changing the color of a stain to a blood stain.

The forgery detection algorithm tests for the presence of the fingerprint in each $B\times B$ sliding block separately and then fuses all local decisions. For simplicity, it will be assumed that the image under investigation did not undergo any geometrical processing. For each block, $\mathcal{B}_b$, the detection problem is formulated as hypothesis testing

$$\mathrm{H}_0\text{:}\ \mathbf{W}_b = \Xi_b$$

$$\mathrm{H}_1\text{:}\ \mathbf{W}_b = \mathbf{I}_b\hat{\mathbf{K}}_b + \Xi_b\,. \tag{20}$$

Here, $\mathbf{W}_b$ is the block noise residual, $\hat{\mathbf{K}}_b$ is the corresponding block of the fingerprint, $\mathbf{I}_b$ is the block intensity, and $\Xi_b$ is the modeling noise assumed to be a white Gaussian noise with an unknown variance $\sigma_\Xi^2$. The likelihood ratio test is the normalized correlation

$$\rho_b = corr\left(\mathbf{I}_b\hat{\mathbf{K}}_b, \mathbf{W}_b\right). \tag{21}$$

In forgery detection, one is likely to desire to control both types of error – failing to identify a tampered block as tampered and falsely marking a region as tampered. To this end, the distribution of the test statistic under both hypotheses must be estimated.

The probability density under $\mathrm{H}_0$, $p(x|\mathrm{H}_0)$, can be estimated by correlating the known signal $\mathbf{I}_b\hat{\mathbf{K}}_b$ with noise residuals from other cameras. The distribution of $\rho_b$ under $\mathrm{H}_1$, $p(x|\mathrm{H}_1)$, is much harder to obtain because it is heavily influenced by the block content. Dark blocks will have lower value of correlation due to the multiplicative character of the PRNU. The fingerprint may also be absent from flat areas due to strong JPEG compression or saturation. Finally, textured areas will have a lower value of the correlation due to stronger modeling noise. This problem can be resolved by building a predictor of the correlation that will tell us what the value of the test statistics $\rho_b$ and its distribution would be if the block $b$ was not tampered and indeed came from the camera.

The predictor is a mapping that needs to be constructed for each camera. The mapping assigns an estimate of the correlation $\rho_b$ to each triple $(i_b, f_b, t_b)$, where the individual elements of the triple stand for a measure of intensity, saturation, and texture in block $b$. The mapping can be constructed for example using regression or machine learning techniques by training them on a database of image blocks coming from images taken by the camera. The block size cannot be too small (because then the correlation $\rho_b$ has too large a variance). On the other hand, large blocks would compromise the ability of the forgery detection algorithm to localize. Blocks of 64×64 or 128×128 pixels work well for most cameras.

A reasonable measure of intensity is the average intensity in the block

$$i_b = \frac{1}{|\mathcal{B}_b|}\sum_{i\in\mathcal{B}_b}\mathbf{I}[i]. \tag{22}$$

Among possible measures of flatness, in this report the author selected the relative number of pixels, $i$, in the block whose sample intensity variance $\sigma_\mathbf{I}[i]$ estimated from the local 3×3 neighborhood of $i$ is below a certain threshold

$$f_b = \frac{1}{|\mathcal{B}_b|}\left|\{i\in\mathcal{B}_b \mid \sigma_\mathbf{I}[i] < c\mathbf{I}[i]\}\right|, \tag{23}$$

where $c \approx 0.03$ (for Canon G2 camera). The best values of $c$ vary with the camera model.

A good texture measure should somehow evaluate the amount of edges in the block. Among many available options, the following example gives satisfactory performance

$$t_b = \frac{1}{|\mathcal{B}_b|}\sum_{i\in\mathcal{B}_b}\frac{1}{1+\mathrm{var}_5(\mathbf{F}[i])}, \tag{24}$$

where $\mathrm{var}_5(\mathbf{F}[i])$ is the sample variance computed from a local 5×5 neighborhood of pixel $i$ for a high-pass filtered version of the block, $\mathbf{F}[i]$, such as one obtained using an edge map or a noise residual in a transform domain.

Since one can obtain potentially hundreds of blocks from a single image, only a small number of images (e.g., ten) are needed to train (construct) the predictor. The data used for its construction can also be used to estimate the distribution of the prediction error $\nu_b$

$$\rho_b = \hat{\rho}_b + \nu_b, \tag{25}$$

where $\hat{\rho}_b$ is the predicted value of the correlation.

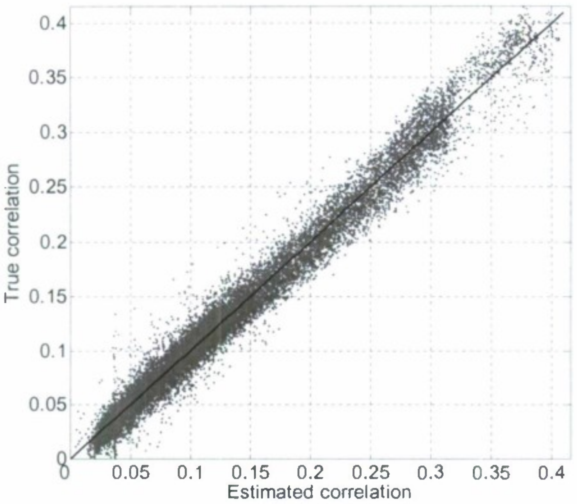


**Fig. 6:** Scatter plot of correlation $\rho_b$ vs. $\hat{\rho}_b$ for 30,000 128×128 blocks from 300 TIF images for Canon G2.

Fig. 6 shows the performance of the predictor constructed using second order polynomial regression for a Canon G2 camera. Say that for a given block under investigation, one applies the predictor and obtains the estimated value $\hat{\rho}_b$. The distribution $p(x|\mathrm{H}_1)$ is obtained by fitting a parametric pdf to all points in Fig. 7 whose estimated correlation is in a small neighborhood of $\hat{\rho}_b$, ($\hat{\rho}_b-\varepsilon$, $\hat{\rho}_b+\varepsilon$). A sufficiently flexible model for the distribution that allows thin and thick tails is the generalized Gaussian model with pdf $\alpha/(2\sigma\Gamma(1/\alpha))e^{-(|x-\mu|/\sigma)^{\alpha}}$ with variance $\sigma^2\Gamma(3/\alpha)/\Gamma(1/\alpha)$, mean $\mu$, and shape parameter $\alpha$.

The description of the forgery detection algorithm using sensor fingerprint now continues. The algorithm proceeds by sliding a block across the image and evaluates the test statistics $\rho_b$ for each block $b$. The decision threshold $t$ for the test statistics $\rho_b$ was set to obtain the probability of misidentifying a tampered block as non-tampered, $\Pr(\rho_b > t|\ \mathrm{H}_0) = 0.01$.

Block $b$ is marked as *potentially tampered* if $\rho_b < t$ but this decision is attributed only to the central pixel $i$ of the block. Through this process, for an $m$×$n$ image one obtains an $(m-B+1)\times(n-B+1)$ binary array $\mathbf{Z}[i] = \rho_b < t$ indicating the potentially tampered pixels with $\mathbf{Z}[i] = 1$.

The above Neyman-Pearson criterion decides 'tampered' whenever $\rho_b < t$ even though $\rho_b$ may be "more compatible" with $p(x|\mathrm{H}_1)$, which is more likely to occur when $\rho_b$ is small, such as for highly textured blocks. To control the amount of pixels falsely identified as tampered, one computes for each pixel $i$ the probability of falsely labeling the pixel as tampered when it was not

$$p[i] = \int_{-\infty} p(x\,|\,\mathrm{H}_1)dx\,. \tag{26}$$

Pixel $i$ is labeled as non-tampered (we reset $\mathbf{Z}[i] = 0$) if $p[i]>\beta$, where $\beta$ is a user-defined threshold (in experiments in this report, $\beta = 0.01$). The resulting binary map $\mathbf{Z}$ identifies the forged regions in their raw form. The final map $\mathbf{Z}$ is obtained by further post-processing $\mathbf{Z}$.

The block size imposes a lower bound on the size of tampered regions that the algorithm can identify. Thus, the author proposes to remove from $\mathbf{Z}$ all simply connected tampered regions that contain fewer than 64×64 pixels. The final map of forged regions is obtained by dilating $\mathbf{Z}$ with a square 20×20 kernel. The purpose of this step is to compensate for the fact that the decision about the whole block is attributed only to its central pixel and we may miss portions of the tampered boundary region.

## 1.5 EXPERIMENTAL VERIFICATION

In this section, the performance of the proposed forensic methods is evaluated and examples of how these techniques may be implemented is also given. References [9,13] contain more extensive tests and [11] and [12] contain experimental verification of device linking and fingerprint matching for video-clips. Camera identification from printed images appears in [10].

### 1.5.1 Camera identification

A Canon G2 camera with a 4 megapixel CCD sensor was used in all experiments in this section. The camera fingerprint was estimated for each color channel separately using the maximum likelihood estimator (6) from 30 blue sky images acquired in the TIFF format. The estimated fingerprints were preprocessed using the column and row zero-meaning explained in Section 1.2 to remove any residual patterns not unique to the sensor. This step is very important because these artifacts would cause unwanted interference at certain spatial shifts, **s**, and scaling factors, and thus decrease the PCE and substantially increase the false alarm rate.

The fingerprints estimated from all three color channels were combined into a single fingerprint using the linear conversion rule (8). All other images involved in this test were also converted to grayscale before applying the detectors described in Section 1.3.

The camera was further used to acquire 720 images containing snapshots or various indoor and outdoor scenes under a wide spectrum of light conditions and zoom settings spanning the period of four years. All images were taken at the full CCD resolution and with a high JPEG quality setting. Each image was first cropped by a random amount up to 50% in each dimension. The upper left corner of the cropped region was also chosen randomly with uniform distribution within the upper left quarter of the image. The cropped part was subsequently downsampled by a randomly chosen scaling ratio $r \in [0.5, 1]$. Finally, the images were converted to grayscale and compressed with 85% quality JPEG.

The detection threshold $\tau$ was chosen to obtain the probability of false alarm $P_{FA} = 10^{-5}$. The camera identification algorithm was run with $r_{min} = 0.5$ on all images. Only two missed detections were encountered (Fig. 7). In the figure, the PCE is displayed as a function of the randomly chosen scaling ratio. The missed detections occurred for two highly textured images. In all successful detections, the cropping and scaling parameters were detected with accuracy better than 2 pixels in either dimension.

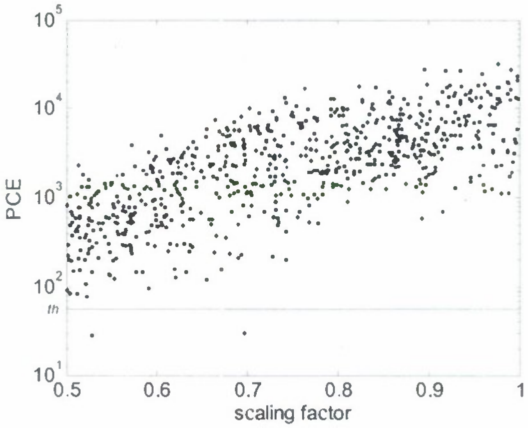


**Fig. 7:** $PCE_{peak}$ as a function of the scaling ratio for 720 images matching the camera. The detection threshold $\tau$, which is outlined with a horizontal line, corresponds to $P_{FA} = 10^{-5}$.

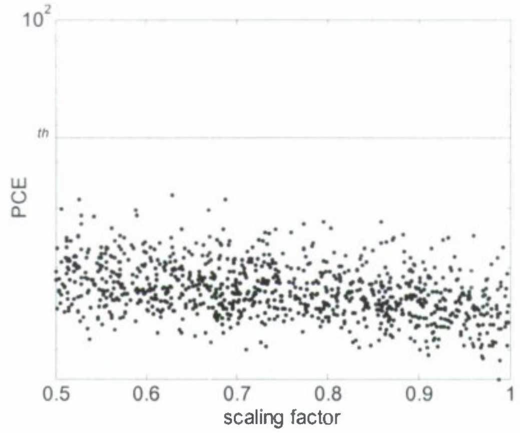


**Fig. 8:** $\mathrm{PCE}_{\mathrm{peak}}$ for 915 images not matching the camera. The detection threshold $\tau$ is again outlined with a horizontal line and corresponds to $P_{\mathrm{FA}} = 10^{-5}$.

To test the false identification rate, 915 images from more than 100 different cameras downloaded from the Internet in native resolution were used. The images were cropped to 4 megapixels (the size of Canon G2 images) and subjected to the same random cropping, scaling, and JPEG compression as the 720 images before. The threshold for the camera identification algorithm was set to the same value as in the previous experiment. All images were correctly classified as not coming from the tested camera (Fig. 8). To experimentally verify the theoretical false alarm rate, millions of images would have to be taken, which is, unfortunately, not feasible.

### 1.5.2 Forgery detection

Fig. 9a shows the original image taken in the raw format by an Olympus C765 digital camera equipped with a 4 megapixel CCD sensor. Using Photoshop, the girl in the middle was covered by pieces of the house siding from the background (Fig. 9b). The forged image was then stored in the TIFF and JPEG 75 formats. The corresponding output of the forgery detection algorithm, shown in Figs. 9c and d, is the binary map **Z** highlighted using a square grid. The last two figures show the map **Z** after the forgery was subjected to denoising using a 3×3 Wiener filter (Fig. 9e) followed by 90% quality JPEG and when the forged image was processed using gamma correction with $\gamma = 0.5$ and again saved as JPEG 90 (Fig. 9f). In all cases, the forged region was accurately detected.

More examples of forgery detection using this algorithm, including the results of tests on a large number automatically created forgeries as well as non-forged images, can be found in the original publication [13] and in [44], which presents an older version of the forgery detection algorithm.

Alternative approaches to detection of digital forgeries were described by other researchers in [33–45].

## 1.6 DENOISING FILTER

The denoising filter used in the experimental sections of this report is constructed in the wavelet domain. It was originally described in [22].

Assume that the image is a grayscale 512×512 image. Larger images can be processed by blocks and color images are denoised for each color channel separately. The high-frequency wavelet coefficients of the noisy image are modeled as an additive mixture of a locally stationary i.i.d. signal with zero mean (the noise-free image) and a stationary white Gaussian noise $N(0,\sigma_0^2)$ (the noise component). The denoising filter is built in two stages. In the first stage, one estimates the local image variance, while in the second stage the local Wiener filter is used to obtain an estimate of the denoised image in the wavelet domain. The individual steps are now described.

**Step 1.** Calculate the fourth-level wavelet decomposition of the noisy image with the 8-tap Daubechies quadrature mirror filters. The author describes the procedure for one fixed level (it is executed for the high-frequency bands for all four levels). Denote the vertical, horizontal, and diagonal subbands as $\mathbf{h}[i,j]$, $\mathbf{v}[i,j]$, $\mathbf{d}[i,j]$, where $(i,j)$ runs through an index set $\mathcal{J}$ that depends on the decomposition level.

**Step 2.** In each subband, estimate the local variance of the original noise-free image for each wavelet coefficient using the MAP estimation for 4 sizes of a square $W\times W$ neighborhood $\mathcal{N}$, for $W\in\{3, 5, 7, 9\}$.

$$\hat{\boldsymbol{\sigma}}_W^2[i,j] = \max\left(0, \frac{1}{W^2}\sum_{(i,j)\in\mathcal{N}} \mathbf{h}^2[i,j]-\sigma_0^2\right), (i,j)\in \mathcal{J}.$$

Take the minimum of the 4 variances as the final estimate,

$$\hat{\boldsymbol{\sigma}}^2(i,j) = \min\left(\boldsymbol{\sigma}_3^2[i,j], \boldsymbol{\sigma}_5^2[i,j], \boldsymbol{\sigma}_7^2[i,j], \boldsymbol{\sigma}_9^2[i,j]\right), (i,j)\in \mathcal{J}.$$

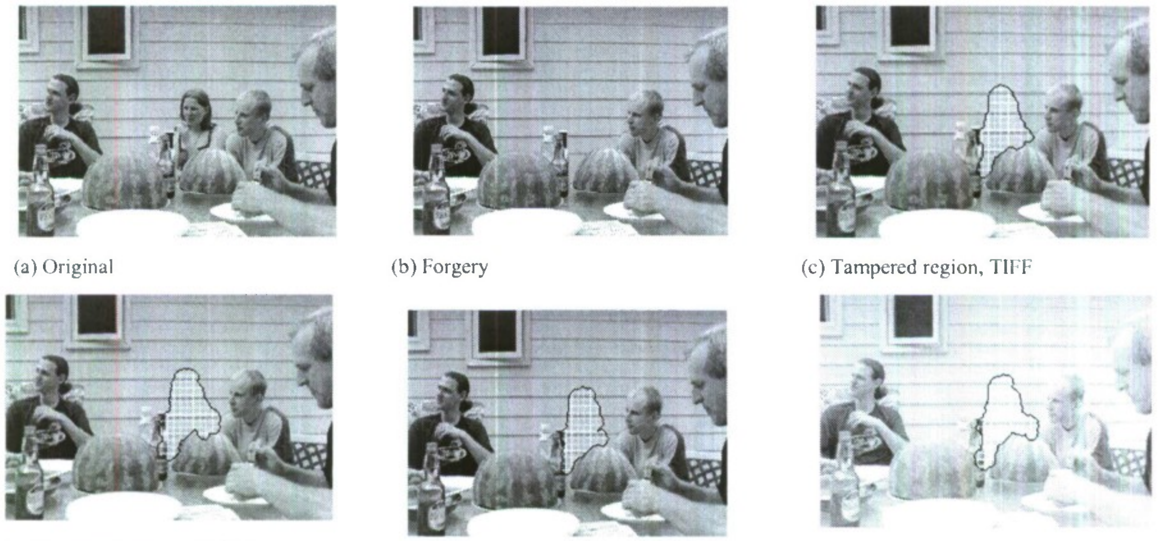
**Step 3.** The denoised wavelet coefficients are obtained using the Wiener filter

$$\mathbf{h}_{\text{den}}[i,j] = \mathbf{h}[i,j]\frac{\hat{\boldsymbol{\sigma}}^2[i,j]}{\hat{\boldsymbol{\sigma}}^2[i,j]+\sigma_0^2}$$

and similarly for $\mathrm{v}[i,j]$, and $\mathbf{d}[i,j]$, $(i,j)\in \mathcal{J}$.

**Step 4.** Repeat Steps 1–3 for each level and each color channel. The denoised image is obtained by applying the inverse wavelet transform to the denoised wavelet coefficients.

In all experiments in this report, $\sigma_0 = 2$ (for dynamic range of images 0, …, 255) to be conservative and to make sure that the filter extracts substantial part of the PRNU noise even for cameras with a large noise component.



(a) Original

(b) Forgery

(c) Tampered region, TIFF

(d) Tampered region, JPEG 75

(e) Tampered region, Wiener 3×3 , JPEG 90

(f) Tampered region, $\gamma = 0.5$ and JPEG 90

**Fig. 9:** An original (a) and forged (b) Olympus C765 image and its detection from a forgery stored as TIFF (c), JPEG 75 (d), denoised using a 3×3 Wiener filter and saved as 90% quality JPEG (e), gamma corrected with $\gamma = 0.5$ and stored as 90% quality JPEG.

## 1.7 SUMMARY

This section introduces several digital forensic methods that capitalize on the fact that each imaging sensor casts a noise-like fingerprint on every picture it takes. The main component of the fingerprint is the photo-response non-uniformity (PRNU), which is caused by pixels' varying capability to convert light to electrons. Because the differences among pixels are due to imperfections in the manufacturing process and silicon inhomogeneity, the fingerprint is essentially a stochastic, spread-spectrum signal and thus robust to distortion.

Since the dimensionality of the fingerprint is equal to the number of pixels, the fingerprint is unique for each camera and the probability of two cameras sharing similar fingerprints is extremely small. The fingerprint is also stable over time. All these properties make it an excellent forensic quantity suitable for many tasks, such as device identification, device linking, and tampering detection.

This section provides a summary of the main results and methods for estimating the fingerprint from images taken by the camera and methods for fingerprint detection. The estimator is derived using maximum likelihood principle from a simplified sensor output model. The model is then used to formulate fingerprint detection as two-channel hypothesis testing problem for which the generalized likelihood detector is derived. Due to its complexity, the GLRT detector is replaced with a simplified but substantially faster detector computable using fast Fourier transform.

The performance of the introduced forensic methods is briefly demonstrated on real images. The following sections contain more details and more extensive experimental verification.

For completeness, we note that there exist approaches combining sensor noise detection with machine-learning classification [14–16]. References [14,17,18] extend the sensor-based forensic methods to scanners. An older version of this forensic method was tested for cell phone cameras in [16] and in [19] where the authors show that combination of sensor-based forensic methods with methods that identify camera brand can decrease false alarms. The improvement reported in [19], however, is unlikely to hold for the newer version of the sensor noise forensic method presented in this report as the results appear to be heavily influenced by uncorrected effects discussed in Section II.B. The problem of pairing of a large number of images was studied in [20] using an ad hoc approach. Anisotropy of image noise for classification of images into scans, digital camera images, and computer art appeared in [21].

Digital forensic methods based on other principles than imaging sensor photo-response non-uniformity include the following work. Artifacts due to color filter interpolation can be used for classification of images to camera models or manufacturers [23–25,30]. Dust present on the protective glass of Single Lens Reflex cameras can also be used for forensic purposes [46].

# 2. CAMERA ID FROM CROPPED AND SCALED IMAGES

This section of the report provides more details about the algorithm for camera identification from images that underwent simultaneous cropping scaling. Extensive experimental results are provided to demonstrate the performance of the techniques in real life conditions.

## 2.2 EXPERIMENTAL RESULTS

Three types of experiments are presented in this section. Tests of the camera ID algorithm for the scaling only case and the cropping only case were performed on 5 different test images along with (and without) JPEG compression (see Section 2.2.1). Section 2.2.2 contains random cropping and random scaling tests with JPEG compression on a single image. This test follows the most likely "real life" scenario and reveals how each processing step affects camera identification. Section 3.3 discusses a special case of cropping and scaling which occurs when digital zoom is engaged in the camera.

### 2.2.1 Scaling only and cropping only

Fig. 10 shows five test images from Canon G2 with a 4 Mp CCD sensor. These images cover a wide range of difficulties from the point of view of camera identification with the first one being the easiest because it contains large flat and bright areas and the last one the most difficult due to its rich texture. The camera fingerprint **K** was estimated from 30 blue sky images in the TIFF format. It was also preprocessed using the column and row zero-meaning (as explained in Section 1.2.2) to remove any residual patterns not unique to the sensor. This step is important because periodicities in demosaicking errors would cause unwanted interference at certain translations and scaling factors, consequently decreasing the PCE (14) and increasing the false alarm rate. The author found that this effect can be quite substantial.

Several different tests were performed to first gain insight into how robust the camera ID algorithm is. In the Scaling Only Test, the test images were subjected to scaling with progressively smaller scaling parameter $r$. The results are displayed in Table 1 showing the PCE($r$) for $0.3 \leq r \leq 0.9$, with no lossy compression and with JPEG compression with quality factors 90%, 75%, and 60%. The downsampling method was bicubic resampling. The upsampling used in the search algorithm was the nearest neighbor algorithm. Here, the author intentionally used a different resampling algorithm because in reality we will not know the resampling algorithm and the author wants the tests to reflect real life conditions.

In the Cropping Only Test, all images were only subjected to cropping with an increasing amount of the cropped out region. The cropped part was always the lower-right corner of the images. Note that while scaling by the ratio $r$ means that the image dimensions were scaled by the factor $r$, cropping by a factor $r$ means that the size of the cropped image is $r$ times the original dimension. In particular, scaling and cropping by the same factor produces images with the same number of pixels, $r^2 \times mn$.

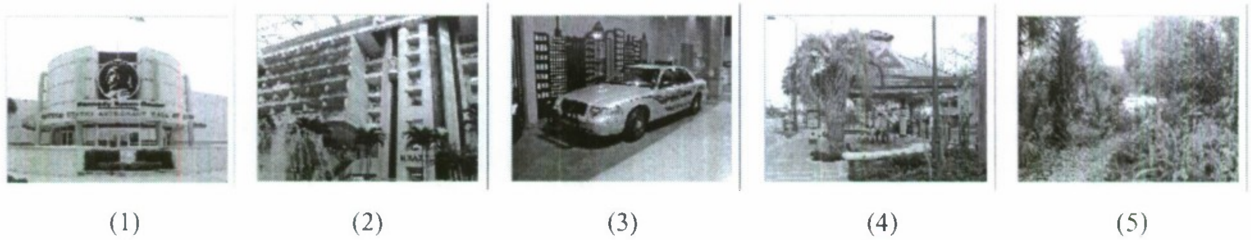


**Fig. 10:** Five test images from a 4 MP Canon G2 camera ordered by the richness of their texture (their difficulty to be identified).

The image identification in the Scaling Only Test (left half of Table 1) was successful for all 5 images and JPEG compression factors when the scaling factor was not smaller than 0.5. It started failing when the scaling ratio was 0.4 or lower and the JPEG quality was 75% or lower. Image #5 was correctly identified at ratios 0.5 and above although its content is difficult to suppress for the denoising filter. The largest PCE that did not determine the correct parameters [$\mathbf{s}_{\text{peak}}$; $r_{\text{peak}}$] was 38.502 (image #1). On the other hand, the lowest PCE for which the parameters were correctly determined was 35.463 (also for image #1). In some cases, the maximum value of NCC did occur for the correct cropping and scaling parameters but the identification algorithm failed because the PCE was below the threshold set to achieve $P_{FA} \leq 10^{-5}$.

Image cropping has a much smaller effect on image identification (the Cropping Only Test part of Table 1). It was possible to correctly determine the exact position of the cropped image within the (unknown) original in all tested cases. The PCE was consistently above 130 even when the images were cropped to the small size $0.3m \times 0.3n$ and compressed with JPEG quality factor 60.

| | | Scaling Only Test | | | | | Cropping Only Test | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Im. #1 | Im. #2 | Im. #3 | Im. #4 | Im. #5 | Im. #1 | Im. #2 | Im. #3 | Im. #4 | Im. #5 |
| | TIFF | 68003 | 32325 | 28570 | 20092 | 2964 | 14624 | 9157 | 7789 | 8961 | 3106 |
| $r = 0.9$ | Q 90 | 28951 | 12834 | 11347 | 6971 | 1515 | 20905 | 11475 | 8223 | 7348 | 2637 |
| $\tau=45.1$ | Q 75 | 6131 | 3343 | 3436 | 2225 | 793 | 7192 | 4509 | 4668 | 3193 | 1480 |
| | Q 60 | 1778 | 1245 | 1291 | 1037 | 461 | 2751 | 2029 | 2227 | 1706 | 919 |
| | TIFF | 72971 | 35086 | 30128 | 17494 | 2260 | 9282 | 8148 | 7764 | 8043 | 3058 |
| $r = 0.8$ | Q 90 | 19041 | 8045 | 7080 | 4171 | 955 | 15287 | 8988 | 8251 | 6060 | 2329 |
| $\tau=48.3$ | Q 75 | 2980 | 1610 | 1736 | 1115 | 417 | 5698 | 2979 | 3977 | 2550 | 1226 |
| | Q 60 | 763 | 645 | 614 | 348 | 222 | 2216 | 1224 | 1916 | 1337 | 754 |
| | TIFF | 41835 | 20058 | 17549 | 9674 | 910 | 6496 | 7986 | 7384 | 6030 | 2508 |
| $r = 0.7$ | Q 90 | 8255 | 3642 | 3731 | 1938 | 410 | 11434 | 7887 | 9210 | 4533 | 2037 |
| $\tau=50.4$ | Q 75 | 1316 | 724 | 979 | 507 | 170 | 4565 | 2593 | 3807 | 2142 | 995 |
| | Q 60 | 328 | 318 | 406 | 203 | 124 | 1774 | 1155 | 1860 | 1139 | 622 |
| | TIFF | 42192 | 18991 | 16902 | 8986 | 1001 | 4896 | 8249 | 6550 | 4353 | 1971 |
| $r = 0.6$ | Q 90 | 4619 | 2060 | 2195 | 1311 | 422 | 7679 | 6328 | 8574 | 2918 | 1445 |
| $\tau=52.1$ | Q 75 | 625 | 399 | 476 | 214 | 117 | 2883 | 1951 | 3443 | 1255 | 682 |
| | Q 60 | 190 | 166 | 196 | 85 | 53 | 1166 | 709 | 1568 | 669 | 459 |
| | TIFF | 24767 | 10719 | 9442 | 4781 | 487 | 4339 | 6592 | 5216 | 3006 | 1599 |
| $r = 0.5$ | Q 90 | 1721 | 997 | 927 | 533 | 196 | 6314 | 3989 | 6996 | 2206 | 1265 |
| $\tau=53.5$ | Q 75 | 211 | 168 | 221 | 134 | 111 | 2622 | 1089 | 2799 | 1003 | 695 |
| | Q 60 | 144 | 56 | 70 | (50) | 58 | 1216 | 398 | 1322 | 574 | 464 |
| | TIFF | 8083 | 3193 | 2897 | 1310 | 123 | 4008 | 4469 | 4163 | 1881 | 1086 |
| $r = 0.4$ | Q 90 | 457 | 227 | 280 | 151 | (52) | 3730 | 2274 | 5178 | 1230 | 846 |
| $\tau=54.8$ | Q 75 | 72 | (52) | 72 | (53) | *31* | 1407 | 723 | 2113 | 570 | 453 |
| | Q 60 | (35) | (43) | (43) | *26* | *34* | 599 | 255 | 1006 | 311 | 324 |
| | TIFF | 777 | 352 | 477 | 124 | *27* | 3518 | 1585 | 3030 | 916 | 837 |
| $r = 0.3$ | Q 90 | (46) | (41) | 60 | *38* | *30* | 2577 | 742 | 3414 | 657 | 609 |
| $\tau=55.9$ | Q 75 | *39* | *32* | *35* | *31* | *35* | 969 | 269 | 1378 | 300 | 259 |
| | Q 60 | *35* | *33* | *35* | *32* | *35* | 461 | 139 | 721 | 164 | 177 |

**Table 1:** PCE in the Scaling Only Test followed by JPEG compression. The PCE is in italic when the scaling ratio was not determined correctly. Values in parentheses are below the detection threshold $\tau$ (see the leftmost column) for $P_{FA} < 10^{-5}$.

### 2.2.2 Random cropping and random scaling simultaneously

This series of tests focused on the performance of the search method on image #2. The image underwent 50 *simultaneous* random cropping and scaling with both scaling and cropping ratios between 0.5 and 1 followed by JPEG compression with the same quality factors as in the previous tests. The maximum PCE values found in each search were sorted by the scaling ratio (since it has by far the biggest influence on the algorithm performance) and plotted the PCE in Fig. 11. The threshold $\tau = 56.315$ displayed in the figure corresponds to the worst scenario (largest search space) of 0.5 scaling and 0.5 cropping for false alarm rate below $10^{-5}$. In the test, no missed detection occurred for the JPEG quality factor 90, 1 missed detection for JPEG quality factor 75 and scaling ratio close to 0.5, and 5 missed detections for JPEG quality factor 60 when the scaling ratios were below 0.555. Even though these numbers will vary significantly with the image content, they provide insight into the robustness of the method on real images.

The last test was a large scale test intended to evaluate the real-life performance of the proposed methodology. The database of 720 images contained snapshots spanning the period of four years. All images were taken at the full CCD resolution and with a high JPEG quality setting. Each image was first subjected to a randomly-generated cropping up to 50% in each dimension. The cropping position was also chosen randomly with uniform distribution within the image. The cropped part was further resampled by a scaling ratio $r\in[0.5, 1]$. Finally, the image was compressed with 85% quality JPEG. The false alarm was set again to $10^{-5}$. Running our algorithm with $r_{\min} = 0.5$ on all images processed this way, we encountered 2 missed detections (Fig. 5a), which occurred for more difficult (textured) images. In all successful detections, the cropping and scaling parameters were detected with accuracy better than 2 pixels in either dimension.

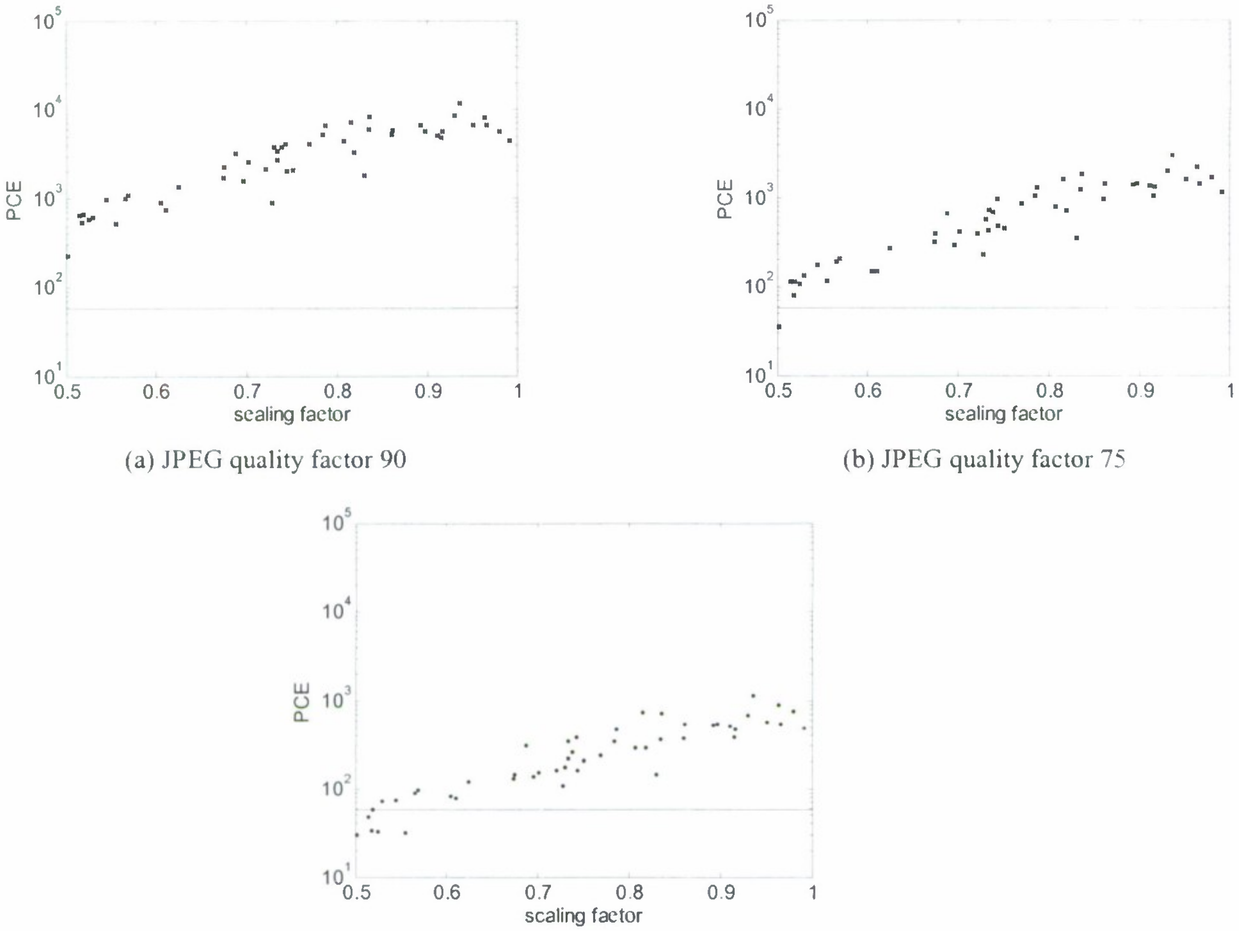
To complement this test, 915 images from more than 100 different cameras were downloaded from the Internet in native resolution, cropped to the 4 Mp size of Canon G2 and subjected to the same random cropping, scaling, and JPEG compression as the 720 images before. No single false detection was encountered. All maximum values of PCE were below the threshold with the overall maximum at 42.5.

### 2.2.3 Digital zoom

While optical zoom does not desynchronize PRNU with the image noise residual (it is equivalent to a change of scene), when a camera engages digital zoom, it introduces the following geometrical transformation: the middle part of the image is cropped and up-sampled to the resolution determined by the camera settings. This is a special case of our cropping and scaling scenario. Since the cropping may be a few pixels off the center, one needs to search for the scaling factor $r$ as well as the shift vector $\mathbf{s}$. The maximum digital zoom determines the upper bound on the search for the scaling factor (see Section 1.3.1). For simplicity, the same search is applied for cropping as before although it would be possible to use a restricted search range around the image center.
Some cameras allow almost continuous digital zoom (e.g., Fuji E550) while other offer only several fixed values. This is the case of Canon S2. The camera display indicates zoom values "24×", "30×", "37×", and "48×", which correspond to digital zoom scaling ratios 1/2, 1/2.5, 1/3.0833, and 1/4, considering the 12× camera optical zoom. The test using camera fingerprint revealed exact scaling ratios 1/2.025, 1/2.5313, 1/3.1154, and 1/4, corresponding to cropped sizes 1280×960, 1024×768, 832×624, and 648×486, respectively. Thus, in general for camera identification, one may wish to check these digital zoom scaling values first before proceeding with the rest of the search if no match is found. Note that none of the camera manuals for the two tested cameras (Fuji and Canon) contained any information about the digital zoom. The details about their digital zooms were found using the PRNU! Thus, this is an interesting example of using the PRNU as a template to recover processing history or reverse-engineer in-camera processing.

Table 2 shows the maximal PCE on 10 images taken with Canon S2 and Fuji E550, some of which were taken with digital zoom. Both cameras were identified with very high confidence in all 10 cases. Images from Fuji camera yielded smaller maximum PCEs, which suggests that (if the image content is dark or heavily textured) the identification of Fuji E550 camera could be more difficult than Canon S2. The detected cropped dimensions (see Table 2) are either precise or off only by a few pixels. This camera apparently has a much finer increment when adjusting the digital zoom than Canon S2. Since the Fuji E550 user is not informed about the fact that the digital zoom has been engaged, it may be quite tedious to find all possible scaling values in this case. The largest digital zoom the camera offers for full resolution output size is 1.4.  Fig. 12 shows images with detected cropped frame for the last two Fuji camera images of the same scene.

The fact that it is possible to obtain previous dimensions of the up-sampled images is an example of "reverse engineering" for revealing image processing history. Such information is potentially useful in forensics sciences even if the source camera is positively known beforehand.



(a) JPEG quality factor 90

(b) JPEG quality factor 75

(e) JPEG quality faetor 60

**Fig. 11.** PCE for image #2 after a series of random sealing and eropping followed by 90% quality JPEG eompression.

| | Canon S2 | | | Fuji E550 | | |
|---|---|---|---|---|---|---|
| Image # | scaling detected | max PCE | cropped dim | scaling detected | max PCE | cropped dim |
| 1 | 3.1154 | 1351 | 832×624 | 1.1530 | 358 | 2470×1853 |
| 2 | 2.5313 | 6020 | 1024×768 | 1.3434 | 238 | 2120×1590 |
| 3 | 2.0250 | 2792 | 1280×960 | 1.3940 | 102 | 2043×1532 |
| 4 | 2.0203 | 9250 | 1283×962 | 1.1837 | 310 | 2406×1805 |
| 5 | 2.5313 | 5929 | 1024×768 | 1.0234 | 576 | 2783×2087 |
| 6 | 3.1154 | 3509 | 832×624 | 1.0007 | 328 | 2846×2135 |
| 7 | 4.0062 | 2450 | 647×485 | 1.0000 | 314 | 2848×2136 |
| 8 | 4.0062 | 1265 | 647×485 | 1.0845 | 1022 | 2626×1970 |
| 9 | 4.0062 | 1620 | 649×487 | 1.1530 | 976 | 2470×1853 |
| 10 | 4.0062 | 1612 | 647×485 | 1.3428 | 378 | 2121×1591 |

**Table 2:** Deteetion of sealing and eropping for digitally zoomed images.

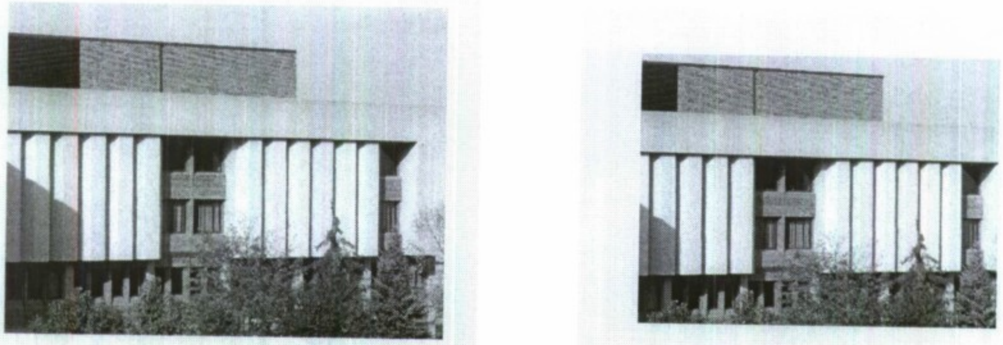


**Fig. 12:** Cropping detected for Fuji E550 images #9 and #10.

## 3. DEVICE LINKING

This section contains details and experimental verification of the device linking algorithm described in Sec. 1.3.2. The goal of device linking is to establish that two images came from the exact same physical camera even though the camera itself may not be available at all. From the analysis presented in Section 1.3.1, it is known that a pronounced, sharp peak in the normalized cross-correlation (NCC) (12) between the noise residuals $\mathbf{W}_1$ and $\mathbf{W}_2$ of both images is indicative of the fact that the images were taken with the same camera. Fig. 13a shows a typical example of such a peak. Besides the Peak Correlation to Energy ratio (PCE) (14) used to measure the peak in Section 1 and 2, there exist several alternative measures of peak sharpness [8]. In this section, the ratio between the primary peak to the secondary peak (PSR) will be used instead to demonstrate that the camera ID technology is robust with respect to the rather ad hoc measures of peak sharpness. It is defined as the largest value in the NCC excluding a central region around the primary peak. The size of this region is determined by observing when the NCC first drops to half of the primary peak.

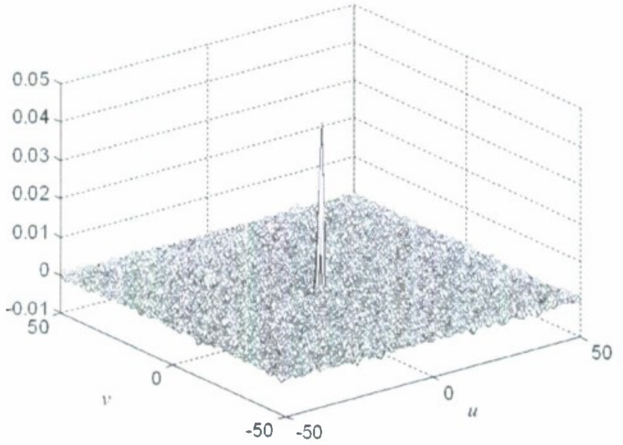


**Fig. 13:** NCC for the suboptimal test statistics (14) in the range $-50 \le u \le 50$, $-50 \le v \le 50$ for a pair of two aligned images produeed by the same eamera.

An image pair is declared to come from the same camera if PSR $\ge T$, where $T$ is a threshold selected to obtain a desired false positive rate (falsely identifying an image pair as coming from the same camera). From the Central Limit Theorem, the cross-correlation values for non-matching images are well approximated using Gaussian distribution. The cumulative density function (cdf) of the PSR for $n$ samples taken from a Gaussian distribution with pdf $f(x)$ and cdf $F(x)$ is

$$c(z) = 1 - nz \int_{-\infty}^{\infty} f(xz)F^{n-1}(x)dx,\ z \ge 1\,. \tag{27}$$

Thus, setting the threshold to $T$ will produce the false alarm rate of

$$P_{FA} = 1 - c(T). \tag{28}$$

For experiments, images were used coming from 8 cameras from different manufacturers with a variety of sensors and resolutions. They included 6 CCD cameras Canon G2, Canon S40, Kodak DC290, Olympus C3030, Olympus C765 (two cameras of the exact brand), and two CMOS cameras – Sigma SD9 with the Foveon sensor and Canon XT Rebel.

Total of 10 images of various indoor and outdoor scenes in the raw format were taken with each camera. Then, *for each camera*, the device linking algorithm for matching and non-matching image pairs was run. All 10×9/2 = 45 matching pairs were tested as well as 200 randomly chosen pairs where the first image was among the 10 images taken by the camera and the other image came from the remaining 7 cameras. For each test, the PSR value was registered. Some statistics (range and median) of the PSR values are displayed in Table 3. Fig. 14 shows a sample of 9 images from the tested cameras.

To see how the reliability of the device linking algorithm deteriorates with lossy compression, the same experiment was repeated after all images were compressed using JPEG with quality factor 90 and 75. The results are also shown in Table 3.

Regardless of the quality factor, the largest value of the PSR for an unmatched pair (among 3×8×200 pairs) was 1.3, while the smallest value for a matched pair (out of 3×8×45 pairs) was 1.0. Setting $T = 1.4$ would in this test produce zero false alarms (incorrectly classified non-matching pair) with the probability of false alarm $P_{FA} \cong 5\times10^{-5}$. Table 3 shows the percentage of correctly classified matching pairs with this theoretical false alarm rate. For example, 41 correctly classified cases out of 45 pairs of the raw Canon Rebel images result in 91.1% probability of correct detection of a matched pair (PDM).

The PDM is usually very high for raw images but deteriorates with a decreasing JPEG quality factor. Since the PRNU term **IK** is multiplicative, very dark images are more likely to be misclassified. The same is also true for highly textured images due to the limitation of the denoising filter, which fails to filter out the image content.

| | | Matched Pairs | | | Unmatched Pairs | | | PDM | | | Matched Pairs | | | Unmatched Pairs | | | PDM |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | min | med | max | min | med | max | | | | min | med | max | min | med | max | |
| Canon G2 | Raw | 7.4 | 11.6 | 24.3 | 1.00 | 1.03 | 1.19 | 100% | Olympus C765-1 | Raw | 3.6 | 5.5 | 9.1 | 1.00 | 1.04 | 1.28 | 100% |
| | Q90 | 4.1 | 6.6 | 16.5 | 1.00 | 1.03 | 1.20 | 100% | | Q90 | 1.8 | 3.6 | 4.8 | 1.00 | 1.03 | 1.25 | 100% |
| | Q75 | 1.2 | 2.6 | 6.3 | 1.00 | 1.03 | 1.28 | 97.8% | | Q75 | 1.2 | 1.8 | 2.9 | 1.00 | 1.03 | 1.26 | 88.9% |
| Canon S40 | Raw | 8.8 | 12.6 | 23.2 | 1.00 | 1.03 | 1.30 | 100% | Olympus C765-2 | Raw | 1.9 | 3.0 | 8.3 | 1.00 | 1.03 | 1.29 | 100% |
| | Q90 | 5.3 | 8.4 | 14.3 | 1.00 | 1.03 | 1.30 | 100% | | Q90 | 1.1 | 1.9 | 4.6 | 1.00 | 1.03 | 1.24 | 86.7% |
| | Q75 | 2.2 | 3.3 | 5.2 | 1.00 | 1.03 | 1.30 | 100% | | Q75 | 1.0 | 1.2 | 2.7 | 1.00 | 1.04 | 1.26 | 33.3% |
| Canon XT Rebel | Raw | 1.0 | 2.9 | 5.7 | 1.00 | 1.03 | 1.21 | 91.1% | Olympus C3030 | SHQ | 8.4 | 15.0 | 28.1 | 1.00 | 1.04 | 1.26 | 100% |
| | Q90 | 1.0 | 1.7 | 2.6 | 1.00 | 1.03 | 1.30 | 57.8% | | Q90 | 4.7 | 8.0 | 15.1 | 1.00 | 1.04 | 1.25 | 100% |
| | Q75 | 1.0 | 1.1 | 1.6 | 1.00 | 1.04 | 1.27 | 4.4% | | Q75 | 1.9 | 3.7 | 6.9 | 1.00 | 1.03 | 1.26 | 100% |
| Kodak DC290 | Raw | 2.2 | 7.2 | 13.8 | 1.00 | 1.03 | 1.19 | 100% | Sigma SD9 | Raw | 3.8 | 8.0 | 14.1 | 1.00 | 1.03 | 1.23 | 100% |
| | Q90 | 1.1 | 2.7 | 5.4 | 1.00 | 1.04 | 1.24 | 93.3% | | Q90 | 1.4 | 3.2 | 6.9 | 1.00 | 1.03 | 1.25 | 95.6% |
| | Q75 | 1.0 | 1.4 | 2.2 | 1.00 | 1.03 | 1.23 | 48.9% | | Q75 | 1.0 | 1.5 | 3.7 | 1.00 | 1.04 | 1.24 | 55.6% |

**Table 3:** Minimum, median, and maximum PSR and probability of detection (PDM) for tested image pairs from all cameras. The decision threshold was set so that the probability of false alarms was $P_{FA} \cong 5\times10^{-5}$.

**Fig. 14:** Some sample images used in our tests.

# 4. VIDEO IDENTIFICATION

This section contains extensive experimental verification of the fingerprint linking algorithm proposed for identification of video clips in Section 1.3.3. This algorithm is used to decide whether two video-clips A and B were produced by the exact same camcorder. Let $\mathbf{K}_A$ and $\mathbf{K}_B$ be the PRNUs estimated from both clips. Because the PRNU is a unique signature of the camera, the task of origin identification is equivalent to The test statistic (12) is the NCC between the estimates of both fingerprints, $\hat{\mathbf{K}}_A$ and $\hat{\mathbf{K}}_B$.

## 4.1 REMOVING ARTIFACTS FROM THE FINGERPRINT

Because PRNUs from two different sensors are uncorrelated, if both clips are indeed from the same camcorder, one expects to see a sharp peak in the NCC and a correspondingly large PCE. However, almost all camcorders use DPCM-Block DCT transform-type video coding, such as MPEG-x and H.26x. This creates (i) ringing artifacts at the frame boundaries caused by the padding required for frame dimensions not divisible by the block size and by operations such as motion estimation/compensation for out of frame movement; (ii) 16×16 blockiness artifacts inside the frame because most standard codecs are based on 16×16 macroblocks. These periodic pulse-like signals (Fig. 15a) propagate through the denoising filter into the estimated fingerprints and cause false correlations between otherwise uncorrelated fingerprints. Thus, they must be removed before calculating the NCC. Because of the heavy compression typically encountered in video coding, the fingerprints need to be estimated from thousands of video-frames and the periodic artifacts accumulate more than in the case of camera identification from images.

The boundary artifacts can be easily removed by cropping ~8 pixel wide boundaries in the spatial domain. The periodic pulse-like blockiness artifacts can be removed in the Fourier domain (Fig. 15b) by attenuating the Fourier coefficients at frequencies where most of the artifacts' energy is located. To illustrate how to locate the frequencies of these periodic pulse-like signals, consider the following one-dimensional periodic signal $x(n) = \delta(n-16m),\quad 0 \le n \le N-1$ whose DFT transform is $X(r)$

$$|X(r)| = \frac{\sin\left(\frac{2\pi}{N/16}\frac{k}{2}r\right)}{\sin\left(\frac{2\pi}{N/16}\frac{r}{2}\right)}, \tag{29}$$

where $k = \lfloor (N-1)/16 \rfloor$ and $r$ is the DFT index. Equation (29) shows that the energy of $|X(r)|$ concentrates around frequencies of integer multiples of $N/16$. Therefore, setting $X(r) = 0$ for those frequencies and their neighborhood (3–6 times frequency resolution) effectively reduces the strength of the periodic signal. In the experiments described in this section, a similar effect idea was realized using an FFT domain filter designed to mitigate the deteriorating effect of blockiness on the NCC. Fig. 15b and c show the Fourier magnitude of the fingerprint and the filtered fingerprint. Since in practice the NCC is calculated in the Fourier domain, one can conveniently perform blockiness removal at the same time. Furthermore, other artifacts that manifest themselves as peaks in the Fourier domain will be suppressed, such as artifacts due to color filter array interpolation and other hardware or software operations already mentioned in Section 1.2.

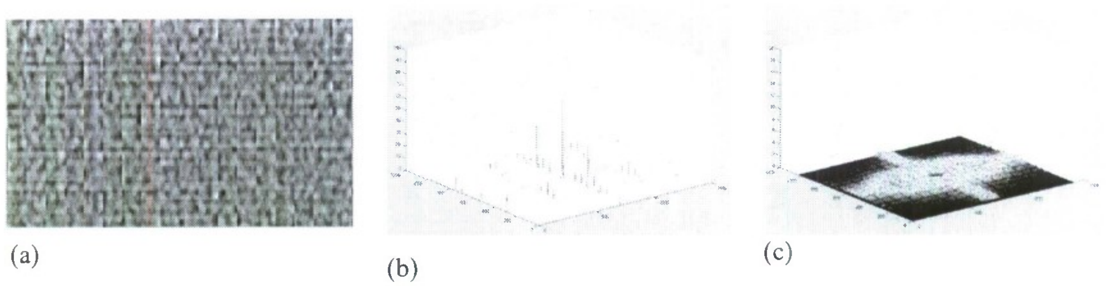


**Fig. 15:** (a) Blockiness artifacts in a small magnified portion of the estimated fingerprint; (b) Fourier magnitude of (a); (c) Fourier magnitude after removing the artifacts in the DFT domain.

## 4.2 EXPERIMENTAL RESULTS

This section contains selected experiments illustrating the effectiveness of the proposed forensic method for identifying the origin of video clips. Twenty-five consumer digital camcorders were used (20 SONY, 4 Hitachi, 1 Canon). The recording media was Mini-DV or DVD-RW and the sensor resolution varied from 0.68MP–4.1 MP. Three camcorders (one Canon DC40 and two camcorders of the same model SONY DCR-DVD105) were selected and tested against the remaining clips. The two SONY camcorders will be addressed as SONY DCR-1 and SONY DCR-2. With each camcorder, several high quality video clips were prepared (roughly 6 Mb/sec, DVD quality, resolution 536×720, frame rate 30 Hz, MPEG-2 VOB format) of various indoor and outdoor scenes. The clips contained brief periods of optical zooming in/out and panning. Some of the videos contained quickly moving objects (e.g., cars) while others had panned static scenes. All the camcorders had their Electronic Image Stabilization (EIS) and digital zooming turned off. All scenes were taped with the fully automatic settings.

The videos were also transcoded to low-bit rate formats, such as the MPEG-4 XviD format (~1Mbit/sec), the RealPlay format (~750 Kbit/sec), and the MPEG-4 DivX format (~450 Kbit/sec). These formats represent the most popular choices for distribution of video over the Internet today.

### 4.2.1 VOB, XviD, RealPlay, DivX vs. VOB

This purpose of this test is to investigate whether it is possible to correctly identify the source camera from videos that were transcoded to 4 different formats and bit-rates. First, the fingerprints were estimated from a 40-second randomly selected video segment from SONY DCR-1 clips in the VOB format. Then, three more fingerprints were estimated from three transcoded formats, Xvid, RealPlay, and DivX, obtaining thus four SONY DCR-1 fingerprints of varying quality. Then, the NCCs were computed with the fingerprints from a different 40-second SONY DCR-1 video clip in the VOB format and 24 fingerprints from 24 40-second video clips from all the other camcorders, also in the VOB format. For the SONY DCR-1, SONY DCR-2, and Canon DC40 camcorders, Fig. 16 shows the NCC surface and the PCE in a pictorial form. The results for the remaining 22 camcorders are summarized in the table below the figure.

In the same manner, two 40-second randomly selected SONY DCR-2 clips and Canon DC40 clips were randomly chosen and tested against all the fingerprints from the 25 camcorders (obtained from VOBs). The results are shown in the same format in Fig. 16b and Fig. 16c. The figures reveal the reliability of the proposed identification approach for all four bit rates. Also, one can see that with the same number of frames, the quality of the estimated fingerprints decreases as the video quality decreases (measured by the bit rate).

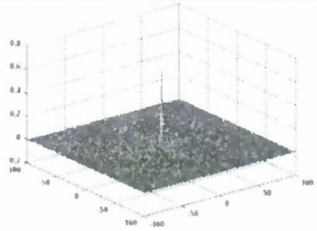
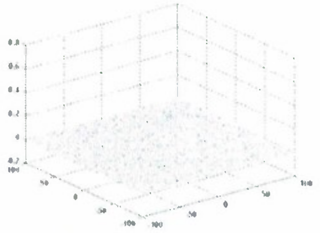
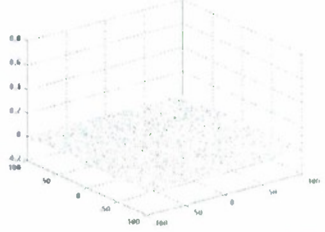
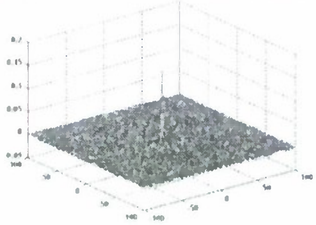
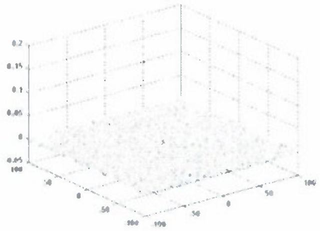
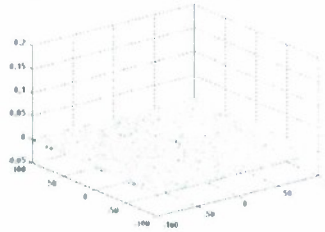
The degradation of the estimated fingerprints is the reason for deterioration of the NCC surface (and the decrease in PCE and correlation coefficient). Regardless of the video format, the PCE and the correlation coefficients obtained for the matched case are by several orders of magnitude larger than for the unmatched case.

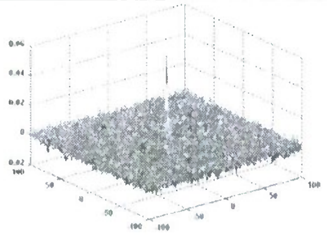
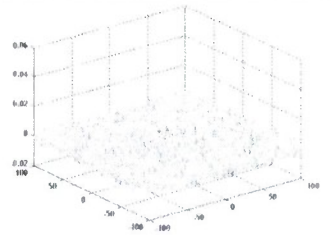
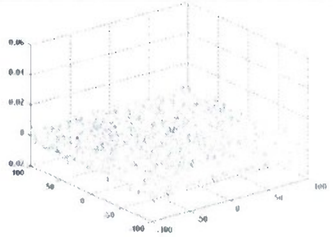
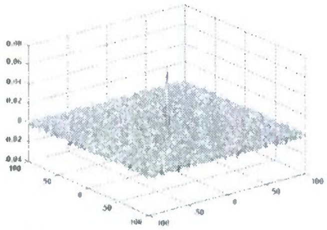
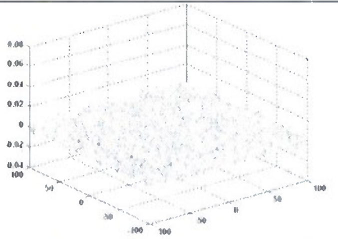
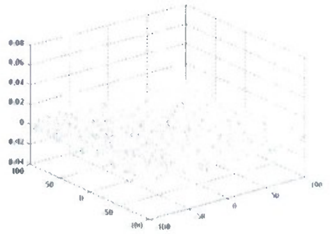
### 4.2.2 Xvid vs. Xvid for clips of different length

In the second experiment, two fingerprints were estimated from two 40-second SONY DCR-2 video clips of different scenes in the XviD-format and the NCC between them was calculated. Then, the same process was repeated with length of the clips increased to 80 seconds and 120 seconds. The resulting NCCs are shown in Fig. 17.

### 4.2.3 Low bit-rate experiment

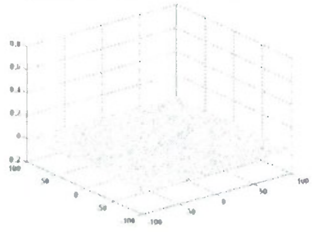
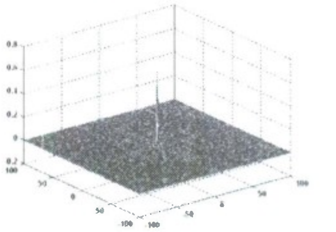
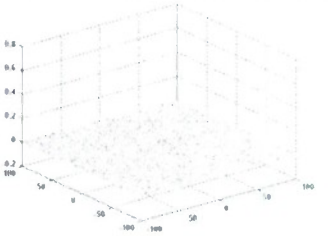
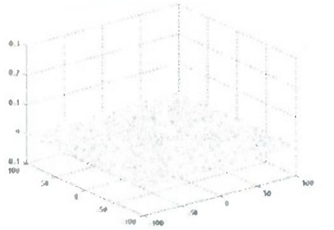
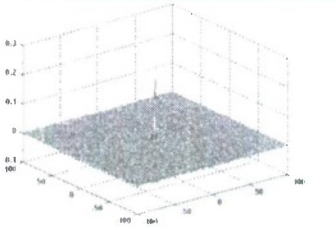
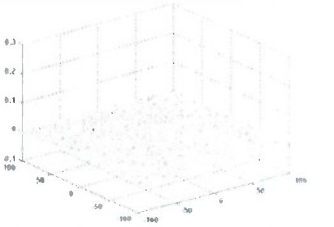
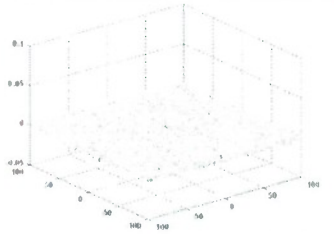
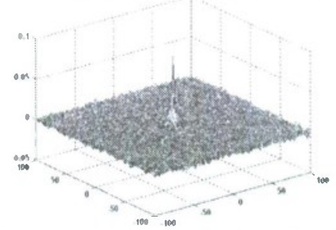
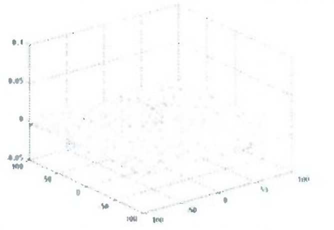
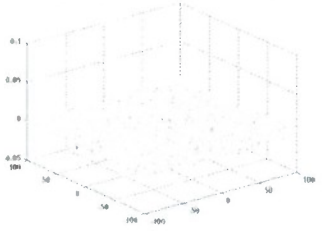
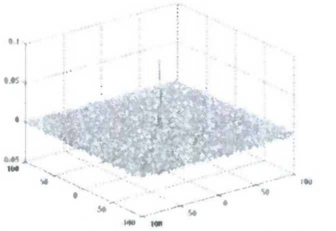
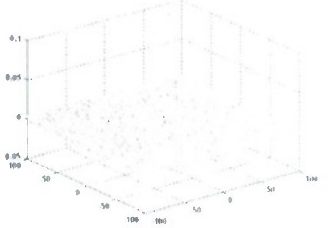
The third experiment focused on identification of "Internet-quality" clips with low resolution and very low bit-rate. Two clips were used – the one from SONY DCR-1 and one from Canon DC40 taken at LP resolution of 264×352 pixels. Both clips were then transcoded to 150kb/sec. in the RMVB format. Then both clips were tested for the presence of a fingerprint estimated from four 2.5min VOB clips from SONY DCR-1. The NCC surfaces and PCEs are shown in Figure 18. The identification is again possible and improves with the length of the clip.

| | SONY DCR-1 40 Secs, 6Mb/s, VOB | SONY DCR-2 40 Secs, 6Mb/s, VOB | Canon DV40 40 Secs, 6Mb/s, VOB |
|---|---|---|---|
| SONY DCR-1<br>40 Secs 6Mb/s VOB |  (a) PCE = 9.0e+04, CorrCoef = 0.6318 |  (b) PCE = 41.1, CorrCoef = 0.0016 |  (c) PCE = 34.1, CorrCoef = 0.0028 |
| SONY DCR-1<br>40 Secs<br>1Mb/s XviD |  (d) PCE = 6.2e+03, CorrCoef = 0.1512 |  (e) PCE = 37.5, CorrCoef = –0.0023 |  (f) PCE = 41.2, CorrCoef = 0.0024 |

| | | | |
|---|---|---|---|
| SONY DCR-1<br>40 Sees<br>750 Kb/s RealPlay | <br>(g) PCE = 936.8, CorrCoef = 0.0573 | <br>(h) PCE = 35.3, CorrCoef = –0.0022 | <br>(i) PCE = 27.6, CorrCoef = 0.0025 |
| SONY DCR-1<br>40 Sees<br>450 Kb/s DivX | <br>(j) PCE = 1.2e+03, CorrCoef = 0.0657 | <br>(k) PCE = 56.7.1, CorrCoef = 0.0024 | <br>(l) PCE = 35.8, CorrCoef = –0.0008 |

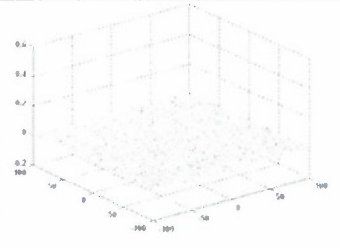
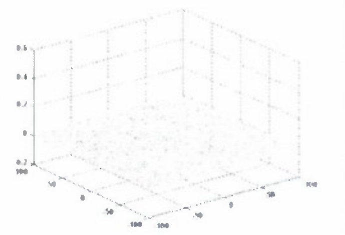
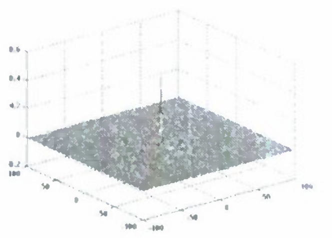
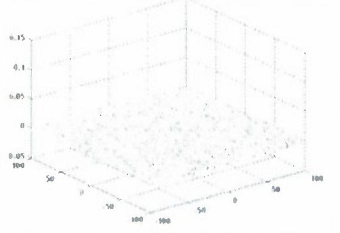
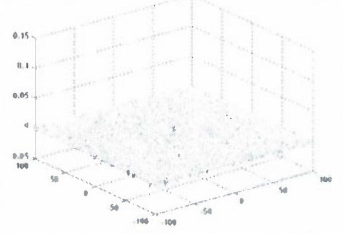
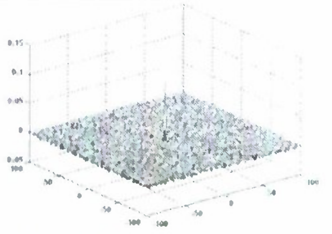
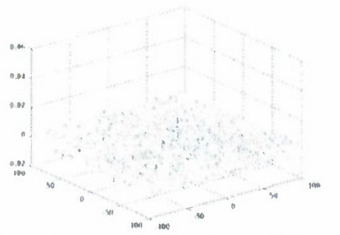
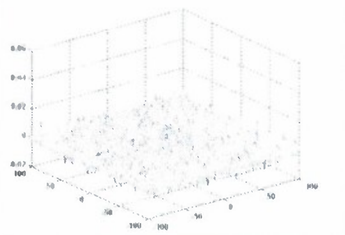
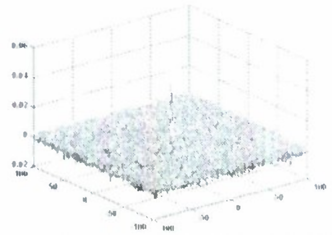
| Other 22 camcorders | | SONY DCR-1<br>40 Secs, 6Mb/s VOB | | SONY DCR-1<br>40 Secs, 1Mb/s XviD | | SONY DCR-1<br>40 Secs, 750 Kb/s RP | | SONY DCR-1<br>40 Secs, 450 Kb/s DivX | |
|---|---|---|---|---|---|---|---|---|---|
| | | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE |
| Statistics | Min | –0.0041 | 28.0 | –0.0044 | 28.2 | –0.0053 | 26.8 | –0.0035 | 25.5 |
| | Max | 0.0084 | 89.0 | 0.0045 | 90.3 | 0.0046 | 73.9 | 0.0050 | 156.9 |
| | Median | –0.0004 | 43.2 | –0.0005 | 37.3 | 0.0012 | 32.5 | 0.0007 | 38.7 |

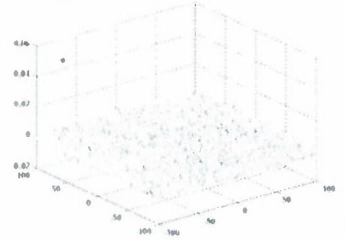
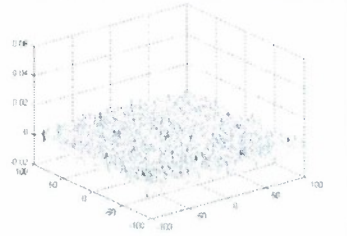
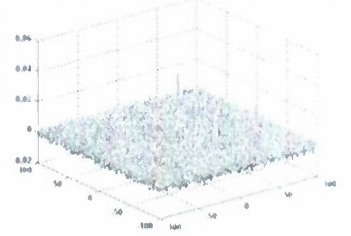
**Fig. 16a:** NCC of PRNUs of 4 differently transcoded versions of a SONY DCR-1 clip with PRNUs estimated from 25 camcorders in the VOB format.

| | SONY DCR-1 40 Secs, 6Mb/s, VOB | SONY DCR-2 40 Secs, 6Mb/s, VOB | Canon DV40 40 Secs, 6Mb/s, VOB |
|---|---|---|---|
| SONY DCR-2<br>40 Secs 6Mb/s VOB | <br>(a) PCE = 38.0, CorrCoef = 0.0028 | <br>(b) PCE = 8.2e+04, CorrCoef = 0.6464 | <br>(c) PCE = 38.8, CorrCoef = 0.0058 |
| SONY DCR-2<br>40 Secs<br>1Mb/s XviD | <br>(d) PCE = 32.8, CorrCoef = 0.0006 | <br>(e) PCE = 1.0e+04, CorrCoef = 0.2038 | <br>(f) PCE = 34.8, CorrCoef = 0.0006 |
| SONY DCR-2<br>40 Secs<br>750 Kb/s RealPlay | <br>(g) PCE = 28.1, CorrCoef = 0.0005 | <br>(h) PCE = 2.1e+03, CorrCoef = 0.0867 | <br>(i) PCE = 28.0, CorrCoef = 0.0013 |
| SONY DCR-2<br>40 Secs<br>450 Kb/s DivX | <br>(j) PCE = 37.2, CorrCoef = –0.0016 | <br>(k) PCE = 1.9e+03, CorrCoef = 0.0871 | <br>(l) PCE = 33.6, CorrCoef = –0.0018 |

| Other camcorders 22 | | SONY DCR-2<br>40 Secs, 6Mb/s VOB | | SONY DCR-2<br>40 Secs, 1Mb/s XviD | | SONY DCR-2<br>40 Secs, 750 Kb/s RP | | SONY DCR-2<br>40 Secs, 450 Kb/s, DivX | |
|---|---|---|---|---|---|---|---|---|---|
| | | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE |
| Statistics | Min | –0.0070 | 22.6 | –0.0058 | 29.5 | –0.0060 | 28.4 | –0.0044 | 27.6 |
| | Max | 0.0059 | 73.7 | 0.0051 | 100.8 | 0.0059 | 116.5 | 0.0065 | 61.6 |
| | Median | –0.0022 | 43.1 | –0.0015 | 38.5 | –0.0005 | 39.3 | 0.0005 | 35.8 |

**Fig. 16b**: NCC of PRNUs of 4 differently transcoded versions of a SONY DCR-2 clip with PRNUs estimated from 25 camcorders in the VOB format.

| | **SONY DCR-1 40 Sees, 6Mb/s, VOB** | **SONY DCR-2 40 Sees, 6Mb/s, VOB** | **Canon DV40 40 Secs, 6Mb/s, VOB** |
|---|---|---|---|
| **Canon DV40**<br>**40 Secs 6Mb/s VOB** |  **(a) PCE = 38.2, CorrCoef = 0.0032** |  **(b) PCE = 38.7, CorrCoef = 0.0072** |  **(c) PCE = 3.7e+04, CorrCoef = 0.4644** |
| **Canon DV40**<br>**40 Sees**<br>**1Mb/s XviD** |  **(d) PCE = 35.1, CorrCoef = –0.0002** |  **(e) PCE = 55.0, CorrCoef = 0.0002** |  **(f) PCE = 2.0e+03, CorrCoef = 0.0982** |
| **Canon DV40**<br>**40 Sees**<br>**750 Kb/s RealPla y** |  |  |  |

| | **(g) PCE = 25.8, CorrCoef = –0.0034** | **(h) PCE = 31.3, CorrCoef = –0.0016** | **(i) PCE = 370.9, CorrCoef = 0.0404** |
|---|---|---|---|
| **Canon DV40**<br>**40 Secs**<br>**450 Kb/s DivX** | <br>**(j) PCE = 32.6, CorrCoef = 0.0022** | <br>**(k) PCE = 40.5, CorrCoef = –0.0008** | <br>**(l) PCE = 390.0, CorrCoef = 0.0429** |

| **Other 22 camcorders** | | Canon DV40 40 Secs, 6Mb/s VOB | | Canon DV40 40 Secs, 1Mb/s XviD | | Canon DV40 40 Secs, 750 Kb/s RP | | Canon DV40 40 Secs, 450 Kb/s DivX | |
|---|---|---|---|---|---|---|---|---|---|
| | | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE | CorrCoef | PCE |
| Statistics | Min | –0.0058 | 26.8 | –0.0033 | 28.7 | –0.0045 | 32.5 | –0.0041 | 26.0 |
| | Max | 0.0111 | 121.1 | 0.0080 | 122.3 | 0.0050 | 94.5 | 0.0039 | 82.2 |
| | Median | 0.0021 | 56 | –0.0012 | 38.4 | –0.0013 | 32.4 | –0.0005 | 39.9 |

**Fig. 16c:** NCC of PRNUs of 4 differently transcoded versions of a Canon clip with PRNUs estimated from 25 camcorders in the VOB format.

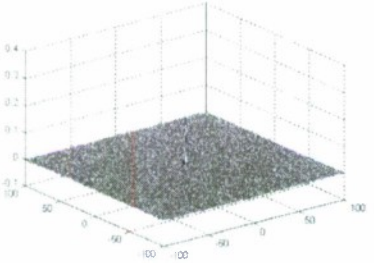
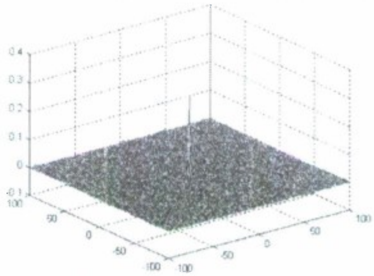
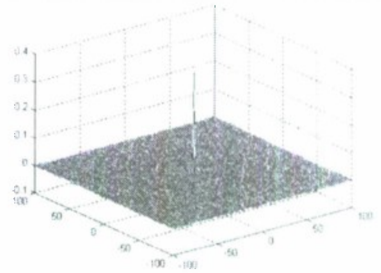
| SONY DCR-2, 40 Sees, 1Mb/s, XviD NCC<br>SONY DCR-2, 40 Sees, 1Mb/s, XviD | SONY DCR-2, 80 Secs, 1Mb/s, XviD NCC<br>SONY DCR-2, 80 Sees, 1Mb/s, XviD | SONY DCR-2, 120 Secs, 1Mb/s, XviD NCC<br>SONY DCR-2, 120 Secs, 1Mb/s, XviD |
|---|---|---|
|  <br>(a) PCE = 9.0e+03, CorrCoef = 0.1893 |  <br>(b) PCE = 1.8e+04, CorrCoef = 0.2830 |  <br>(e) PCE = 2.8e+04, CorrCoef = 0.3630 |

**Fig. 17:** NCCs of PRNUs from different SONY DCR-2 XviD-format video clips with the length 40, 80, and 120 seconds.

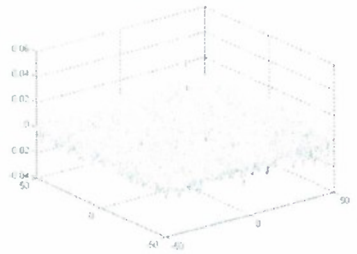
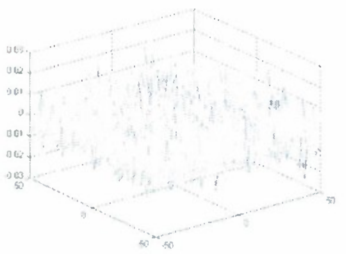
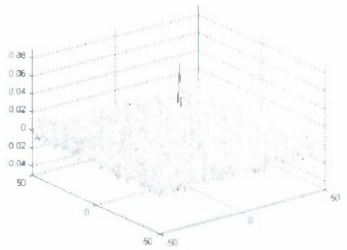
| | SONY DCR-1 10 Mins, 150 Kb/s RMVB | Canon DC40 10 Mins, 150 Kb/s RMVB | SONY DCR-1 40 Mins, 150 Kb/s RMVB |
|---|---|---|---|
| SONY DCT-1 10 Mins 1.5 Mb/s<br>VOB |  <br>(a) PCE = 111, CorrCoef = 0.0580 |  <br>(b) PCE = 22, CorrCoef = 0.0018 |  <br>(e) PCE = 196, CorrCoef = 0.0834 |

**Fig. 18:** NCC surface and PCE coefficient for two low-resolution, low bit-rate clips from SONY DCR-1 and Canon DC40 with PRNU estimated from a 10 minute VOB clip from SONY DCR-1. 4a) is for a 10 minute clip from SONY DCR-1 and 4b) for 40 minute clip.

## REFERENCES

[1] Janesick, J.R.: *Scientific Charge-Coupled Devices*, SPIE PRESS Monograph vol. PM83, SPIE–The International Society for Optical Engineering, January, 2001.

[2] Healey, G. and Kondepudy, R.: "Radiometric CCD Camera Calibration and Noise Estimation." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16(3), pp. 267–276, March, 1994.

[3] Kay, S.M.: *Fundamentals of Statistical Signal Processing*, Volume I, Estimation theory, Prentice Hall, 1998.

[4] Kay, S.M.: *Fundamentals of Statistical Signal Processing*, Volume II, Detection theory, Prentice Hall, 1998.

[5] El Gamal, A., Fowler, B. Min, H., and Liu, X.: "Modeling and Estimation of FPN Components in CMOS Image Sensors." *Proc. SPIE*, *Solid State Sensor Arrays: Development and Applications II*, vol. 3301-20, San Jose, CA, pp. 168–177, January 1998.

[6] Filler, T., Fridrich, J., and Goljan, M.: "Using Sensor Pattern Noise for Camera Model Identification." To appear in *Proc. IEEE ICIP 08*, San Diego, CA, September 2008.

[7] Holt, C.R.: "Two-Channel Detectors for Arbitrary Linear Channel Distortion," *IEEE Trans. on Acoustics, Speech, and Sig. Proc.*, vol. ASSP-35(3), pp. 267–273, March 1987.

[8] Vijaya Kuma, B.V.K. and Hassebrook, L.: "Performance Measures for Correlation Filters," *Appl. Opt.* 29, 2997–3006, 1990.

[9] Goljan, M. and Fridrich, J.: "Camera Identification from Cropped and Scaled Images." *Proc. SPIE Electronic Imaging, Forensics, Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, San Jose, California, January 28 – 30, pp. 0E-1–0E-13 2008.

[10] Goljan, M. and Fridrich, J.: "Camera Identification from Printed Images." *Proc. SPIE Electronic Imaging, Forensics, Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, San Jose, California, January 28 – 30, pp. OI-1–OI-12, 2008.

[11] Goljan, M., Mo Chen, and Fridrich, J.: "Identifying Common Source Digital Camera from Image Pairs." *Proc. IEEE ICIP 07*, San Antonio, TX, 2007.

[12] Chen, M., Fridrich, J., and Goljan, M.: "Source Digital Camcorder Identification Using CCD Photo Response Non-uniformity." *Proc. SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, San Jose, California, January 28 – February 1, pp. 1G–1H, 2007.

[13] Chen, M., Fridrich, J., Goljan, M., and Lukáš, J.: "Determining Image Origin and Integrity Using Sensor Noise." *IEEE Transactions on Information Security and Forensics*, vol. 3(1), pp. 74–90, March 2008.

[14] Gou, H., Swaminathan, A., and Wu, M.: "Robust Scanner Identification Based on Noise Features." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 0S–0T, 2007.

[15] Khanna, N., Mikkilineni, A.K., Chiu, G.T.C., Allebach, J.P., and Delp, E.J. III: "Forensic Classification of Imaging Sensor Types." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 0U–0V, 2007.

[16] Sankur, B., Celiktutan, O., and Avcibas, I.: "Blind Identification of Cell Phone Cameras." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 1H–11, 2007.

[17] Gloe, T., Franz, E., and Winkler, A.: "Forensics for Flatbed Scanners." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 11–1J, 2007.

[18] Khanna, N., Mikkilineni, A.K., Chiu, G.T.C., Allebach, J.P., and Delp, E.J. III: "Scanner Identification Using Sensor Pattern Noise." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 1K–1L, 2007.

[19] Sutcu, Y., Bayram, S., Sencar, H.T., and Memon, N.: "Improvements on Sensor Noise Based Source Camera Identification." Proc. IEEE International Conference on Multimedia and Expo, pp. 24–27, July, 2007.

[20] Bloy, G.J.: "Blind Camera Fingerprinting and Image Clustering." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30(3), pp. 532–534, March 2008.

[21] Khanna, N., Chiu, G.T.-C., Allebach, J.P., and Delp, E.J.: "Forensic Techniques for Classifying Scanner, Computer Generated and Digital Camera Images." *Proc. IEEE ICASSP*, pp. 1653 – 1656, March 31 – April 4, 2008.

[22] Mihcak, M.K., Kozintsev, I., and Ramchandran, K.: "Spatially Adaptive Statistical Modeling of Wavelet Image Coefficients and its Application to Denoising." *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Phoenix, AZ, vol. 6, pp. 3253–3256, March 1999.

[23] Kharrazi, M., Sencar, H.T., and Memon, N.: "Blind Source Camera Identification", Proc. ICIP' 04, Singapore, October 24–27, 2004.

[24] Bayram, S., Sencar, H.T., and Memon, N.: "Source camera identification based on CFA interpolation," ICIP 05, Genoa, Italy, September 2005.

[25] Swaminathan, A., Wu, M., and Liu, K.J.R.: "Non-intrusive Forensic Analysis of Visual Sensors Using Output Images," IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'06), May 2006.

[26] Geradts, Z., Bijhold, J., Kieft, M., Kurosawa, K., Kuroki, K., and Saitoh, N.: "Methods for Identification of Images Acquired with Digital Cameras", Proc. of SPIE, Enabling Technologies for Law Enforcement and Security, vol. 4232, pp. 505–512, February 2001.

[27] Lukáš, J. Fridrich, J., and Goljan, M.: "Digital Camera Identification from Sensor Pattern Noise," IEEE Transactions on Information Security and Forensics, vol. 1(2), pp. 205–214, June 2006.

[28] Chen, M., Fridrich, J., and Goljan, M.: "Digital Imaging Sensor Identification (Further Study)," Proc. SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, San Jose, California, January 28 – February 1, pp. 0P–0Q, 2007.

[29] Cox, I., Miller, M.L., and Bloom, J.A.: Digital Watermarking, Morgan Kaufmann, San Francisco, 2001.

[30] Swaminathan, A., Wu, M., and Liu, K.J.R.: "Image Authentication via Intrinsic Fingerprints." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 1J–1K, 2007.

[31] Kurosawa, K., Kuroki, K., and Saitoh, N.: "CCD Fingerprint Method – Identification of a Video Camera from Videotaped Images." *Proc. ICIP'99*, Kobe, Japan, pp. 537–540, October 1999.

[32] Lukáš, J., Fridrich, J., and Goljan, M.: "Digital Camera Identification from Sensor Pattern Noise." *IEEE Transactions on Information Security and Forensics*, vol. 1(2), pp. 205–214, June 2006.

[33] Ng, T.-T., Chang, S.-F., and Sun, Q.: "Blind Detection of Photomontage Using Higher Order Statistics." *Proc. IEEE International Symposium on Circuits and Systems,* vol. **5**, Vancouver, Canada, pp. v-688–v-691, 2004.

[34] Avcibas, I., Bayram, S., Memon, N., Ramkumar, M., and Sankur, B.: "A Classifier Design for Detecting Image Manipulations." *Proc. ICIP'04*, vol. 4, pp. 2645–2648, 2004.

[35] Lin, Z., Wang, R., Tang, X., and Shum, H.-Y.: "Detecting Doctored Images Using Camera Response Normality and Consistency." *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. **1**, pp. 1087–1092, 2005.

[36] Popescu, A.C. and Farid, H.: "Exposing Digital Forgeries by Detecting Traces of Resampling." *IEEE Transactions on Signal Processing*, vol. **53**(2), pp. 758–767, 2005.

[37] Popescu, A.C. and Farid, H.: "Exposing Digital Forgeries in Color Filter Array Interpolated Images." *IEEE Transactions on Signal Processing*, vol. **53**(10), pp. 3948–3959, 2005.

[38] Johnson, M.K. and Farid, H.: "Exposing Digital Forgeries by Detecting Inconsistencies in Lighting." *Proc. ACM Multimedia and Security Workshop*. New York, pp. 1–9, 2005.

[39] Farid, H.: "Exposing Digital Forgeries in Scientific Images." *Proc. ACM Multimedia & Security Workshop*. Geneva, Switzerland, pp. 29–36, 2006.

[40] Johnson, M.K. and Farid, H.: "Exposing Digital Forgeries Through Chromatic Aberration." *Proc. ACM Multimedia and Security Workshop*. Geneva, Switzerland, pp. 48–55, 2006.

[41] Chen, W. and Shi, Y.: "Image Splicing Detection Using 2D Phase Congruency and Statistical Moments of Characteristic Function." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, January 29–February 1, San Jose, CA, pp. 0R–0S, 2007.

[42] Fridrich, J., Soukal, D., and Lukáš, J.: "Detection of Copy-Move Forgery in Digital Images." *Proc. Digital Forensic Research Workshop*. Cleveland, August 2003.

[43] Popescu, A.C. and Farid, H.: "Exposing Digital Forgeries by Detecting Duplicated Image Regions." *Technical Report,* TR2004-515. Dartmouth College, Computer Science, 2004.

[44] Lukáš, J., Fridrich, J., and Goljan, M.: "Detecting Digital Image Forgeries Using Sensor Pattern Noise." *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII*, vol. 6072. San Jose, California, pp. 0Y1–0Y11, 2006.

[45] Bayram, S., Avcibas, I., Sankur, B. and Memon, N.: "Image Manipulation Detection," *Journal of Electronic Imaging*, vol. 15(4), 041102, 2006.

[46] Dirik, A., Sencar, E., Husrev T., Memon, N., and Khrazzi, M.: "Source Camera Identification Based on Sensor Dust Characteristics." *Proc. IEEE Workshop on Signal Processing Applications for Public Security and Forensics, SAFE '07.* April 11–13, 2007, pp. 1–6.