DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

Effect of Compression on Sensor-Fingerprint Based Camera Identification

Miroslav Goljan,[*] Mo Chen,[+] Pedro Comesaña,[$] and Jessica Fridrich[*], [*]Department of ECE, SUNY Binghamton, NY, USA, {mgoljan,fridrich}@binghamton.edu, [+]Jadak Tech., Syracuse, NY, mochen8@gmail.com, [$]Department of Signal Theory and Communications, University of Vigo, Spain, pcomesan@gts.uvigo.es

Abstract

Camera identification using sensor fingerprints is nowadays a well-established technology for reliably linking an image or a video clip to a specific camera. The sensor fingerprint is typically estimated from images (video frames) provably taken by the imaging device. An image or a video clip is then associated with the fingerprint when some form of a matched filter exceeds a certain threshold, which is set to achieve a prescribed false alarm. However, when the images from which the sensor fingerprint is estimated are lossy compressed, the statistical properties of the detection statistic change, which requires an adjustment of the decision threshold to guarantee the same false-alarm rate. In this paper, we study this effect both theoretically and experimentally. A very good match between the theoretical and experimental results validates our approach. This study is especially important for video forensic because of the higher compression rates.

Motivation

Today, camera identification based on sensor fingerprint is a mature area of research [1, 7, 10, 17]. The method works by first estimating the so-called sensor fingerprint from a set of images positively known to have been taken by the camera. To prove that a given image (or a video clip) under investigation was acquired by the *exact same* camera (not just the same model), one establishes the presence of the fingerprint in the image using signal detection techniques. A positive match between an image and a camera fingerprint ties the image with a very high certainty to the camera. Applications of this technology include all cases when a crime is committed by taking a picture, such as in child pornography and movie piracy cases. It can also be used for detection of malicious image manipulation [4, 5, 12, 6] and for general intelligence gathering. In July 2011, the methodology passed the Daubert challenge (`http://en.wikipedia.org/wiki/Daubert_standard`) in the State of Alabama. It has also been used in high-profile cases, such as the "Operation Algebra" (`http://p10.hostingprod.com/@spyblog.org.uk/blog/2009/05/`).

The methodology is applicable to all digital imaging devices that contain CCD or CMOS sensors. The vast majority of published work focuses on still images with only a handful of papers directed to digital videocameras [3, 8, 18, 19, 13]. Lossy compression that is typically applied to images and video complicates setting appropriate detection thresholds to guarantee a preset false alarm (mistakenly identifying an image as coming from a specific camera). The authors are not aware of any systematic and rigorous study of the effect of lossy compression on sensor-based camera identification. For still images, a large scale test [10] was used to experimentally determine the decision thresholds for the Peak Correlation to Energy ratio (PCE) [3]. No such study is currently available for video.

Although this forensic technique is applicable to both still images and video, there are some important differences: video usually has a much lower spatial resolution, harsher compression, and widely varying ISO. On the other hand, many more images (frames) for both estimation and detection of sensor fingerprint are available. The lower spatial resolution is somewhat compensated for by the large number of frames typically available for analysis, i.e., the total amount of information, which can be in the first approximation expressed using the file size, is typically much larger for video than for digital images. On the other hand, if there is a systematic artifact or modulation present in the compressed signal, it gets undesirably enhanced during fingerprint estimation. It has been recognized already in [3] that sensor fingerprints estimated from video frames contain "artifacts of JPEG compression" that need to be filtered out to lower the false-alarm rate. In [3], a filter in the Fourier domain was proposed for this purpose. However, the effectiveness of this filter in controlling the false alarm (and its effect on positive detection) was never properly studied.

The main goal of this paper is to analyze the effect of lossy compression on the detection statistic in order to properly adjust the decision threshold for a given false alarm. In the next section, we provide a brief overview of the technology behind camera identification using sensor fingerprints. Then, in the third section, we carry out theoretical analysis of the effect of lossy compression on the detection statistic PCE. The following section contains experiments carried out on still images as well as video. A close match between the theoretically derived results and experiment is observed. The focus of the study is on the statistical properties of the detection statistic under the null hypothesis when the fingerprints do not match as this is what determines the threshold for a fixed false-alarm rate. The paper is concluded in the last section.

Boldface symbols will be used for vectors and matrices, while their elements will be denoted with the correspond-

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

ing non-bold font. For example, pixel values in an uncompressed 8-bit grayscale image $\mathbf{X}$ will be denoted with $X_{ij} \in \mathcal{I} \triangleq \{0,\dots,255\}$, $1 \le i \le n_1, 1 \le j \le n_2$, where $n_1 \times n_2$ are image dimensions. Whenever the range of indices i,j is not specified for brevity, it will always be assumed to be $1 \le i \le n_1, 1 \le j \le n_2$. We will use $\mathcal{N}(\mu,\sigma^2)$ for a Gaussian random variable with mean μ and variance σ^2. The fact that a random variable X follows distribution f will be denoted $X \sim f$. The probability of an event E is $\Pr\{E\}$. The operation of rounding to integers is 'round'.

Preliminaries

The main part of the sensor fingerprint is the so-called Photo-Response Non-Uniformity (PRNU) [14], which quantifies the fact that each pixel on the sensor outputs a photon count (charge) that slightly but consistently differs from its nominal value. Consequently, each image the sensor takes is overlaid with a unique noise-like pattern modulated by the scene light intensity.

Let us assume that we have N grayscale images taken by a single camera (alternatively, N luminance frames from a digital video). The pixel intensity of the i,j-th pixel in the kth image $\mathbf{X}^{(k)}$ will be denoted $X_{ij}^{(k)}$, $1 \le i \le n_1$, $1 \le j \le n_2$. Furthermore, we will denote the noise residual of the kth image as

$$\mathbf{W}^{(k)} = \mathbf{X}^{(k)} - F(\mathbf{X}^{(k)}),\ k = 1,\dots,N, \tag{1}$$

where F is a denoising filter estimating the pixels' expectation. In this paper, we use the wavelet based denoising filter as described in our prior work [1, 7, 10, 17].

Assuming the following linearized sensor model for the residual [2, 16, 9],[1]

$$\mathbf{W}^{(k)} = \mathbf{K} \cdot \mathbf{X}^{(k)} + \mathbf{\Xi}^{(k)}, \tag{2}$$

where $\mathbf{\Xi}^{(k)}$ is an $n_1 \times n_2$ matrix of independent and identically distributed (i.i.d.) Gaussian random variables with zero mean, the maximum likelihood estimator of the PRNU multiplicative factor $\mathbf{K}$ (the fingerprint) is:

$$\hat{\mathbf{K}} = \frac{\sum_{k=1}^{N} \mathbf{X}^{(k)} \cdot \mathbf{W}^{(k)}}{\sum_{k=1}^{N} (\mathbf{X}^{(k)})^2}. \tag{3}$$

We note that in Eqs. (2)–(3) all matrices are of dimension $n_1 \times n_2$ and all matrix operations are executed elementwise.

Given two fingerprint estimates, $\hat{\mathbf{K}}_1$ and $\hat{\mathbf{K}}_2$, the detector is facing the following two-channel hypothesis testing problem

$$\begin{aligned} \mathrm{H}_0 &: \quad \mathbf{K}_1 \neq \mathbf{K}_2, \\ \mathrm{H}_1 &: \quad \mathbf{K}_1 = \mathbf{K}_2, \end{aligned} \tag{4}$$

with observables

$$\hat{\mathbf{K}}_1 = \mathbf{K}_1 + \mathbf{\Xi}_1, \quad \hat{\mathbf{K}}_2 = \mathbf{K}_2 + \mathbf{\Xi}_2, \tag{5}$$

[1]Note that the gamma correction factor has been absorbed into the PRNU matrix $\mathbf{K}$.

and a Gaussian corrupting noise $\mathbf{\Xi}_{1,2} \sim \mathcal{N}(\mu, \sigma_\Xi^2)$. A computationally efficient approximation to the generalized likelihood ratio test for the above hypothesis testing problem [11] is the normalized correlation $\rho(\hat{\mathbf{K}}_1, \hat{\mathbf{K}}_2; 0, 0)$ between $\hat{\mathbf{K}}_1$ and $\hat{\mathbf{K}}_2$, which we conveniently introduce here using the cross-correlation:

$$\rho(\mathbf{U},\mathbf{V};\tau_1,\tau_2) = \frac{\sum_{i,j}(U_{ij}-\overline{\mathbf{U}})(V_{i+\tau_1,j+\tau_2}-\overline{\mathbf{V}})}{\sqrt{\sum_{i,j}(U_{ij}-\overline{\mathbf{U}})^2}\sqrt{\sum_{i,j}(V_{i+\tau_1,j+\tau_2}-\overline{\mathbf{V}})^2}}, \tag{6}$$

with the bar standing for the sample mean and the range of indices i,j,τ_1,τ_2 is $1 \le i,\tau_1 \le n_1, 1 \le j,\tau_2 \le n_2$.

Note that under H_0, we are correlating two i.i.d. Gaussian signals since the fingerprint itself is well modeled by an i.i.d. Gaussian random variable. It can be easily established from the central limit theorem that in this case $\rho(\hat{\mathbf{K}}_1, \hat{\mathbf{K}}_2; 0, 0) \sim \mathcal{N}(0, 1/N)$. Thus, in order to set a fixed threshold for the correlation that guarantees a prescribed false alarm, it needs to be scaled by $\sqrt{N}$:

$$\rho(\mathbf{K}, \hat{\mathbf{K}}; 0, 0) \to \sqrt{N}\rho(\mathbf{K}, \hat{\mathbf{K}}; 0, 0). \tag{7}$$

A frequently used detection statistic in practice is the Peak Correlation to Energy ratio (PCE) or signed PCE also referred to as the Circular Correlation Norm (CCN) [15] defined as:

$$\mathrm{PCE}(\mathbf{K}, \hat{\mathbf{K}}) = \frac{\rho^2(\mathbf{K}, \hat{\mathbf{K}}; 0, 0) \times \mathrm{sign}\left(\rho(\mathbf{K}, \hat{\mathbf{K}}; 0, 0)\right)}{\frac{1}{n_1 \cdot n_2 - |\mathcal{N}_{\max}|} \sum_{(\tau_1,\tau_2) \notin \mathcal{N}_{\max}} \rho^2(\mathbf{K}, \hat{\mathbf{K}}; \tau_1, \tau_2)}, \tag{8}$$

where $\mathcal{N}_{\max}$ is a small neighborhood around the origin and $|\mathcal{N}_{\max}|$ is the number of elements in the neighborhood. Note that the PCE can be viewed as another way to normalize the correlation – the denominator is an estimate of the variance of the correlation under the assumption that it has a zero mean. One of the main results of this paper is the fact that in the presence of JPEG compression, this variance estimate is biased, which will necessitate an adjustment of the decision threshold for the PCE when the fingerprints are estimated from JPEG compressed images.

Denoting the pixel values in the bth 8×8 block by $X_{ij}^{(b)}$, $i,j = 0,\dots,7$, their discrete cosine transform (DCT) coefficients $\mathbf{C}^{(b)} = \mathrm{DCT}(\mathbf{X}^{(b)})$ are $C_{kl}^{(b)} = \sum_{i,j=0}^{7} f_{ij}^{kl} X_{ij}^{(b)}$, where

$$f_{ij}^{kl} = \frac{1}{4} w_k w_l \cos\frac{\pi}{16} k(2i+1) \cos\frac{\pi}{16} l(2j+1) \tag{9}$$

$$w_0 = 1/\sqrt{2}, \quad w_k = 1 \text{ for } 1 \le k \le 7. \tag{10}$$

During JPEG compression, DCT coefficients $C_{kl}^{(b)}$ are divided by quantization steps Q_{kl}, rounded to integers, and clipped to the required dynamic range if necessary, $D_{kl}^{(b)} = \mathrm{round}\left(C_{kl}^{(b)}/Q_{kl}\right)$. Finally, the pixel values in the decompressed image are $\tilde{\mathbf{X}}^{(b)} = \mathrm{round}\left(\mathrm{DCT}^{-1}(\mathbf{Q} \cdot \mathbf{D}^{(b)})\right)$, where the matrix operations are again elementwise.

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

Theoretical analysis

In this section, we derive the statistical properties of the normalized correlation (6) and the PCE (8) under the null hypothesis in (4) (non-matching fingerprints) when the images from which the fingerprints are estimated have been JPEG compressed. The main goal is to derive the decision threshold for the correlation (and for PCE) that would guarantee a prescribed false-alarm rate and to determine its dependence on the JPEG quality factor.

To make the analysis more feasible, we will assume that the fingerprints are estimated from one flat-content image $\mathbf{X}$, such as a picture of blue sky. In this special case, there is no need to use the denoising filter F (1) to remove the content because the signal mean is a good estimate of the pixel expectations and it can be subtracted from the image, obtaining thus a zero-mean noise residual $\mathbf{R} = \mathbf{X} - \overline{\mathbf{X}}$. We will model the residual as an array of i.i.d. realizations of zero-mean Gaussians $R_{ij} \sim \mathcal{N}(0, \sigma_R^2)$. Assuming we have two such "estimates" of the PRNU from two different images $\mathbf{X}$ and $\mathbf{Y}$ taken by two different cameras (the null hypothesis in camera identification (4)), we are interested in computing the statistical properties of the normalized correlation $\rho(\mathbf{R}, \mathbf{S}, 0, 0)$ (6) as well as $\rho(\tilde{\mathbf{R}}, \tilde{\mathbf{S}}; q)$, where $\tilde{\mathbf{R}}$ and $\tilde{\mathbf{S}}$ are the residuals JPEG compressed with quality factor q and decompressed to the spatial domain without rounding.

Normalized correlation

In order to analyze the normalized correlation between two images, we first study the normalized correlation between two 8×8 blocks. Then, we consider the correlation between the two full images by employing two simplifications: 1) the denominator of the normalized correlation between two 8×8 blocks stays almost the same across the blocks and thus can be factored out; 2) the numerator of the normalized correlation for each 8×8 block is independent of its counterpart for any other block, which allows us to apply the Central Limit Theorem (CLT) to their sum over all blocks and model it with a Gaussian. Formally, assuming both $\mathbf{R}$ and $\mathbf{S}$ are zero mean and denoting their values from the bth 8×8 block with $R_{kl}^{(b)}$ and $S_{kl}^{(b)}$, $0 \le k, l \le 7$,

$$\rho(\mathbf{R}, \mathbf{S}; \tau_1, \tau_2) = \frac{\sum_b \sum_{k,l=0}^{7,7} R_{kl}^{(b)} S_{k+\tau_1, l+\tau_2}^{(b)}}{\sqrt{\sum_b \sum_{k,l=0}^{7,7} (R_{kl}^{(b)})^2} \sqrt{\sum_b \sum_{k,l=0}^{7,7} (S_{k+\tau_1,l+\tau_2}^{(b)})^2}} \approx \frac{1}{64 N_b \sigma_R \sigma_S} \sum_b \sum_{k,l=0}^{7,7} R_{kl}^{(b)} S_{k+\tau_1,l+\tau_2}^{(b)}, \quad (11)$$

where N_b is the number of 8×8 blocks in the image.

Formally, we consider the normalized correlation between a single 8×8 block of $\tilde{\boldsymbol{R}}$ and an arbitrarily shifted version of an 8×8 block of $\tilde{\boldsymbol{S}}$. Because the statistical properties of the normalized correlation are periodic due to the 8×8 size of the block DCT, we only need to consider 64 possible shifts of $\tilde{\boldsymbol{S}}$: (τ_1, τ_2), $1 \le \tau_1, \tau_2 \le 8$. Furthermore, the shifted residual $\tilde{\boldsymbol{S}}$ will never straddle more than four adjacent DCT blocks.

Since the 8×8 DCT is an orthonormal transform, we can compute the normalized correlation equivalently in the DCT domain. To this end, we define a matrix $\mathbf{A}(\tau_1, \tau_2) \in \mathbb{R}^{64\times 256}$, which defines the linear relationship between each coefficient of the four 8×8 blocks of DCT coefficients of $\boldsymbol{D}$ and the block of 8×8 DCT coefficients of $\tilde{\boldsymbol{S}}$ obtained after shifting by (τ_1, τ_2), i.e, $\lambda_1(\tilde{\boldsymbol{S}}) = \mathbf{A}(\tau_1, \tau_2)\lambda_2(\boldsymbol{D})$, where $\lambda_1 : \mathbb{R}^{8\times 8} \to \mathbb{R}^{64}$ and $\lambda_2 : \mathbb{R}^{16\times 16} \to \mathbb{R}^{256}$ vectorize an 8×8 block and four 8×8 blocks into column vectors, respectively. In particular, λ_1 performs the vectorization by columns as in Matlab, while λ_2 applies the same columnwise scan within each block as well as across the four adjacent 8×8 blocks:

$$\begin{pmatrix} 0 & \cdots & 56 & 128 & \cdots & 184 \\ \vdots & & \vdots & \vdots & & \vdots \\ 7 & \cdots & 63 & 135 & \cdots & 191 \\ 64 & \cdots & 120 & 192 & \cdots & 248 \\ \vdots & & \vdots & \vdots & & \vdots \\ 71 & \cdots & 127 & 199 & \cdots & 255 \end{pmatrix}. \quad (12)$$

Taking this ordering into account, the matrix $\mathbf{A}(\tau_1, \tau_2)$ can be expressed as

$$A_{ij}(\tau_1, \tau_2) \triangleq \sum_{k_1=l_1}^{u_1} \sum_{k_2=l_2}^{u_2} f_{k_1 k_2}^{\xi_1 \xi_2} f_{(k_1+\tau_1) \bmod 8, (k_2+\tau_2) \bmod 8}^{\nu_1 \nu_2}, \quad (13)$$

where $0 \le i \le 63$, $0 \le j \le 255$, $\mu = \lfloor \frac{j}{64} \rfloor$, $\mu_1 = \mu \bmod 2$, $\mu_2 = \lfloor \frac{\mu}{2} \rfloor$ denote the block index, $\nu = j \bmod 64$, $\nu_1 = \nu \bmod 8$, $\nu_2 = \lfloor \frac{\nu}{8} \rfloor$ denote the position within the block of the original coefficients (before shifting), and $\xi_1 = i \bmod 8$, $\xi_2 = \lfloor \frac{i}{8} \rfloor$ stand for the coefficient position after shifting. The symbols l_1, l_2, u_1, u_2 stand for the bounds in the pixel domain of the 8×8 block whose block DCT defines $\tilde{\boldsymbol{S}}$, i.e.,

$$l_1 \triangleq \begin{cases} 0, & \text{if } \mu_1 = 0 \\ 8 - \tau_1, & \text{otherwise} \end{cases},$$

$$u_1 \triangleq \begin{cases} 7 - \tau_1, & \text{if } \mu_1 = 0 \\ 7, & \text{otherwise} \end{cases},$$

$$l_2 \triangleq \begin{cases} 0, & \text{if } \mu_2 = 0 \\ 8 - \tau_2, & \text{otherwise} \end{cases},$$

$$u_2 \triangleq \begin{cases} 7 - \tau_2, & \text{if } \mu_2 = 0 \\ 7, & \text{otherwise} \end{cases}.$$

For example, for the calculation of the 65th column of matrix $\mathbf{A}(3,4)$ we consider the 16×16 matrix

$$\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{0} \end{pmatrix}, \quad (14)$$

where $\mathbf{0}$ is an 8×8 matrix of zeros and $\mathbf{B}$ is also 8×8 with $B_{ij} = 0$ everywhere except for $B_{11} = 1$. Note that the

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

(τ_1,τ_2)	0	1	2	3	4	5	6	7
0	2.7226	2.3541	2.3352	2.3422	2.3217	2.3422	2.3352	2.3541
1	2.4337	2.1599	2.1355	2.1532	2.1258	2.1532	2.1355	2.1599
2	2.3672	2.0926	2.0609	2.0865	2.0519	2.0865	2.0609	2.0926
3	2.3602	2.0982	2.0732	2.0876	2.0623	2.0876	2.0732	2.0982
4	2.4063	2.1274	2.1022	2.1231	2.0938	2.1231	2.1022	2.1274
5	2.3602	2.0982	2.0732	2.0876	2.0623	2.0876	2.0732	2.0982
6	2.3672	2.0926	2.0609	2.0865	2.0519	2.0865	2.0609	2.0926
7	2.4337	2.1599	2.1355	2.1532	2.1258	2.1532	2.1355	2.1599

(τ_1,τ_2)	0	1	2	3	4	5	6	7
0	2.6087	2.3539	2.3699	2.3832	2.3744	2.3454	2.4087	2.2651
1	2.4740	2.1679	2.1822	2.1218	2.1641	2.1575	2.1373	2.1459
2	2.3576	2.0731	2.0793	2.1191	2.0501	2.1134	2.1234	2.1445
3	2.4687	2.0840	2.0806	2.1154	2.0859	2.1332	2.1710	2.1975
4	2.4389	2.1026	2.0281	2.1515	2.1289	2.0908	2.1264	2.2148
5	2.4258	2.1573	2.1180	2.0511	2.0269	2.0761	2.1280	2.0904
6	2.3807	2.1572	2.0365	2.1383	2.0985	2.1328	2.1464	2.0770
7	2.4837	2.2574	2.2561	2.2042	2.1324	2.1838	2.1583	2.2291

Table 1. Theoretical (top) and empirical (bottom) values of $\sigma_\rho^2(\tau_1,\tau_2)$ (19), scaled by a factor of 10^6 as a function of the spatial shift (τ_1,τ_2) for JPEG quality factor 80, $\sigma_R^2=\sigma_S^2=10$, and images with 1000×1000 pixels.

8×8-blockwise IDCT of this 16×16 matrix is

$$\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \frac{1}{8}\mathbf{1} & \mathbf{0} \end{pmatrix}. \tag{15}$$

For the $(3,4)$ shift, the resulting top-left 8×8 corner of $\mathbf{A}(3,4)$ is

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{16}$$

Note that since we are considering $j=64$, then $\mu=1$, $\mu_1=1$, and $\mu_2=0$, and, consequently, $l_1=5$, $u_1=7$, $l_2=0$, and $u_2=3$, yielding the intervals of the summation indices in (13) that can also be inferred from the last matrix. The resulting matrix of DCT coefficients is

$$\begin{pmatrix} 0.188 & 0.170 & 0 & -0.060 & 0 & 0.040 & 0 & -0.034 \\ -0.209 & -0.190 & 0 & 0.067 & 0 & -0.045 & 0 & 0.038 \\ 0.082 & 0.074 & 0 & -0.026 & 0 & 0.017 & 0 & -0.015 \\ 0.030 & 0.028 & 0 & -0.010 & 0 & 0.007 & 0 & -0.006 \\ -0.063 & -0.057 & 0 & 0.020 & 0 & -0.013 & 0 & 0.011 \\ 0.020 & 0.018 & 0 & -0.007 & 0 & 0.004 & 0 & -0.004 \\ 0.034 & 0.031 & 0 & -0.011 & 0 & 0.007 & 0 & -0.006 \\ -0.042 & -0.038 & 0 & 0.013 & 0 & -0.009 & 0 & 0.008 \end{pmatrix}, \tag{17}$$

which, by scanning according to λ_1, yields the 65th column of $\mathbf{A}(3,4)$. Note that the numbers in the matrix have been rounded to three decimal places to fit into the single column while still being readable.

Concerning the statistical characterization of $\tilde{\boldsymbol{R}}$ and $\tilde{\boldsymbol{S}}$, we need to take into into account that $\Pr\{\tilde{R}_{kl}=t\}=\alpha_R(m,k,l)\cdot\delta(t-m\cdot Q_{kl})$, where $m\in\mathbb{Z}$, and $\alpha_R(m,k,l)=\int_{(m-1/2)Q_{kl}}^{(m+1/2)Q_{kl}} f_R(\tau)d\tau$, and similarly for $\tilde{\boldsymbol{S}}$, where f_R stands for the Gaussian probability density function of R_{kl} (please note that the R_{kl}'s are i.i.d., and consequently their densities $f_{R_{kl}}$ do not depend on k or l). Finally, it is worth

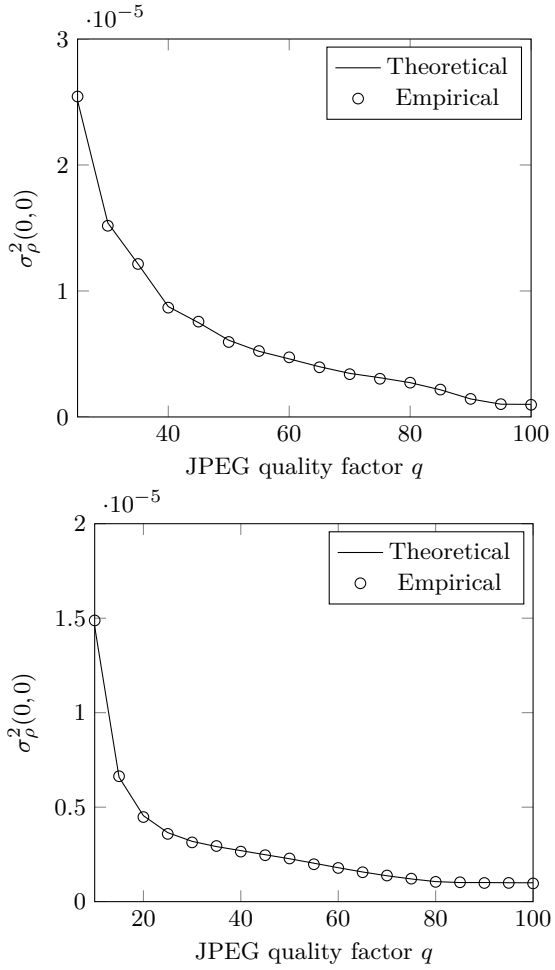

Figure 1. *Comparison of empirical and theoretical values $\sigma_\rho^2(0,0)$ as a function of the JPEG quality factor for and images of 1000×1000 pixels. The empirical results were obtained using 10^4 Monte Carlo realizations. Top: $\sigma_R^2=\sigma_S^2=10$, Bottom: $\sigma_R^2=\sigma_S^2=100$.*

noting that for the arguments of the two square roots in the denominator of (6) their means and variances linearly increase with the block size, which means that the standard deviation of the denominator of the normalized correlation can be neglected in comparison with its mean. Consequently, this term can be accurately modeled as deterministic, supporting thus our first simplification introduced at the beginning of this section and allowing us to factor it out when computing the normalized correlation for the entire image. The normalized correlation for one 8×8 block can thus be modeled by a zero-mean random variable with variance

$$\overline{\sigma}_\rho^2(\tau_1,\tau_2)=\frac{\sum_{i=0}^{63}\sum_{j=0}^{255}A_{ij}^2(\tau_1,\tau_2)\sigma_{\tilde{R}_{\xi_1\xi_2}}^2\sigma_{\tilde{S}_{\nu_1\nu_2}}^2}{\left[\sum_{i=0}^{63}\sigma_{\tilde{R}_{\xi_1\xi_2}}^2\right]\cdot\left[\sum_{i=0}^{63}\sum_{j=0}^{255}A_{ij}^2(\tau_1,\tau_2)\sigma_{\tilde{S}_{\nu_1\nu_2}}^2\right]}, \tag{18}$$

where $\sigma_{\tilde{R}_{kl}}^2=\sum_{m=-\infty}^{\infty}\alpha_R(m,k,l)\cdot m^2\cdot Q_{kl}^2$, and similarly

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

$\sigma^2_{\tilde{S}_{kl}} = \sum_{m=-\infty}^{\infty} \alpha_S(m,k,l) \cdot m^2 \cdot Q^2_{kl}$. Note that in the case when $\mathbf{R}$ is obtained from an image compressed with a different quality factor than $\mathbf{S}$, the previous two expressions for $\sigma^2_{\tilde{R}_{kl}}$ and $\sigma^2_{\tilde{S}_{kl}}$ would contain their corresponding quantization matrices $Q^{(R)}_{kl}$ and $Q^{(S)}_{kl}$.

Although the numerators of normalized correlations corresponding to different blocks are not independent, for the sake of simplicity, and based on our experimental results shown below, we neglect the dependence between the different blocks. This means that, when applying the CLT, the distribution of the normalized correlation for the full image can be accurately approximated by a zero-mean Gaussian with variance

$$\sigma^2_\rho(\tau_1,\tau_2) = 1/N_b \times \overline{\sigma}^2_\rho(\tau_1,\tau_2) \tag{19}$$

where N_b stands for the number of 8×8 DCT blocks in the image.

We next verify the theoretical analysis, and thus validate the simplifying assumptions made, experimentally on synthetic signals. Figure 1 confirms a close match between $\sigma^2_\rho(0,0)$ (19) and its value obtained from 10,000 Monte Carlo simulations as a function of the JPEG quality factor when the variance of the two compared PRNUs is $\sigma^2_R = \sigma^2_S = 10$ and $\sigma^2_R = \sigma^2_S = 100$, for image size $n_1 \times n_2 = 1000 \times 1000$. Furthermore, Table 1 compares the variance of the normalized correlation $\sigma^2_\rho(\tau_1,\tau_2)$ as a function of the considered shift for the theoretical expression in (19) with the variance estimated using 5,000 Monte Carlo realizations. The discrepancy is at most 5%.

PCE

Concerning the distribution of the PCE, inspecting (8) and taking into account the discussion in the previous section, it should be clear that the numerator in the PCE can be modelled by a chi-square distribution with one degree of freedom with mean $\sigma^2_\rho(0,0)$ and variance $2\sigma^4_\rho(0,0)$. On the other hand, in order to derive the distribution of the PCE denominator, we will exploit the fact that $\rho^2(\tilde{\boldsymbol{R}},\tilde{\boldsymbol{S}};k,l)$ are uncorrelated as long as $|k| \geq 7$ or $|l| \geq 7$, meaning that there is a finite number of non-zero coefficients per row and column of the corresponding covariance matrix. Note, indeed, that some of those correlated coefficients will be excluded from the sum in the denominator of the PCE definition by the fact that we consider the neighborhood $\mathcal{N}_{\max}$ around the origin. Consequently, the mean of the denominator will asymptotically not depend on the image size, while its variance can be upper-bounded (neglecting the aforementioned neighborhood) by

$$\frac{225}{(n_1 n_2)^2} \sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} E[\rho^4(\tilde{\boldsymbol{R}},\tilde{\boldsymbol{S}};k,l)], \tag{20}$$

which decreases with the number of pixels $n_1 n_2$. This means that for large images the standard deviation of the PCE denominator can be neglected in comparison with its mean, and, consequently, we can accurately approximate the denominator by a constant. The distribution of the PCE is thus mainly determined by the numerator. In summary, we approximate the distribution of the PCE by a chi-square distribution with one degree of freedom.

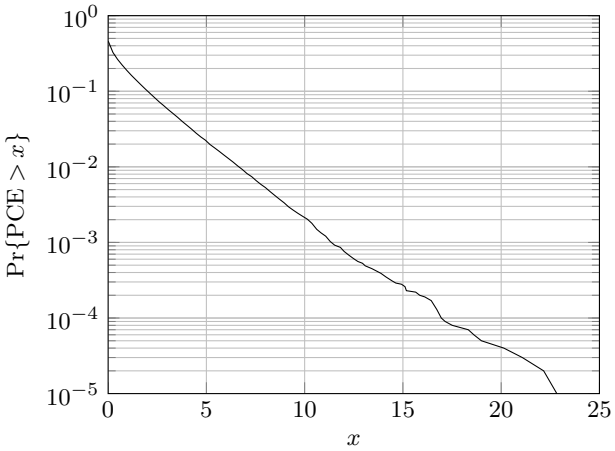

Figure 2. *Empirical distribution of the tail of the PCE,* $\Pr\{\text{PCE} > x\}$*, for JPEG quality factor* $q = 80$*,* $\sigma^2_R = \sigma^2_S = 10$*, and images with* 1000×1000 *pixels, obtained using* 10^5 *Monte Carlo realizations.*

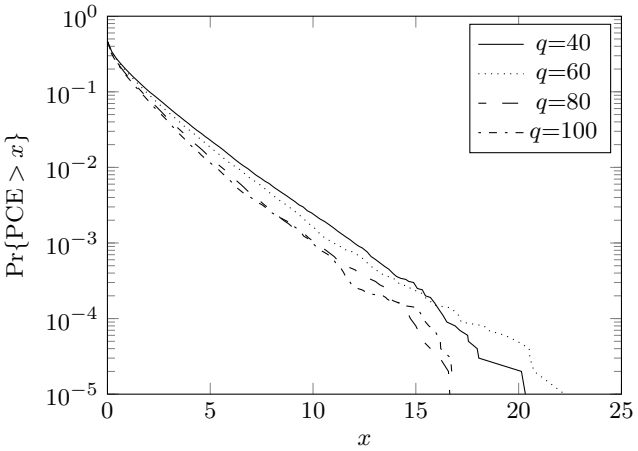

Figure 3. *Empirical distribution of the tail of the PCE,* $\Pr\{\text{PCE} > x\}$*, for JPEG quality factors* $q = 40, 60, 80, 100$*,* $\sigma^2_R = \sigma^2_S = 100$*, and images with* 416×600 *pixels, obtained using* 10^5 *Monte Carlo realizations.*

The good agreement of the proposed theoretical analysis and the Monte Carlo results concerning the distribution of the PCE can be checked by inspecting Figure 2 showing the empirical distribution of the tail of the PCE. In this plot, where a logarithmic scale is used for the y-axis, the log-probability of the right tail, $\log \Pr\{\text{PCE} > x\}$ decreases approximately linearly with x, which is in agreement with the tail of a chi-square distributed random variable with one degree of freedom. Figure 3 depicts the empirical log-probability at four different compression levels and smaller image dimensions (1/4 Mpixel) that can be compared with Figure 8 in the experimental section. Finally, Figure 4 shows the empirical variance of PCE as a function of the JPEG quality factor. As expected, the variance decreases

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

with the quality factor, while the mean (not shown) stays close to zero independently of the quality factor.

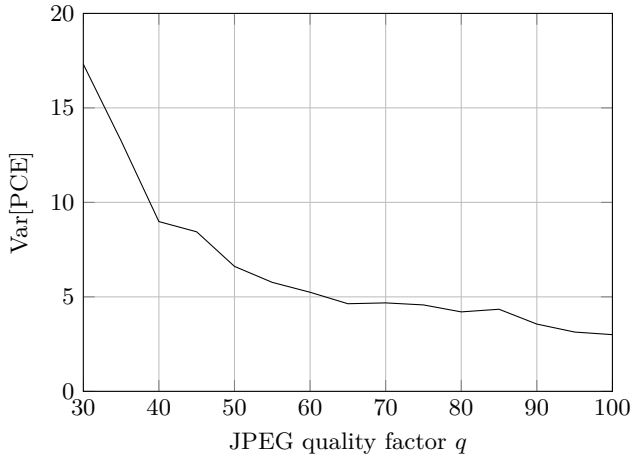

Figure 4. *Empirical variance of the PCE as a function of the JPEG quality factor q, for $\sigma_R^2 = \sigma_S^2 = 10$, images of 1000×1000 pixels, and obtained using $5 \cdot 10^3$ Monte Carlo realizations.*

Experiments

The purpose of this section is to study the effect of compression on the detection statistic (both normalized correlation and the PCE) experimentally and with realistic data and contrast the results with the theory from the previous section. The results of three experiments are included in this section: still images, video in Motion JPEG (M-JPEG), and video in the MPEG-4 format. The M-JPEG format was selected intentionally as a step between still images and MPEG-4 video because it applies frame-by-frame JPEG compression. By choosing to work with realistic data, we introduce many deviations from the assumptions our theoretical analysis is based on. The images and video frames are not i.i.d. Gaussian signals, we are compressimg real content rather than flatfields, and the fingerprints are estimated from multiple noise residuals obtained from JPEG compressed images rather than compressing the fingerprint estimates themselves as assumed in the previous section. Furthermore, we work with color images and estimate the fingerprint from each color channel and then combine the estimates into one estimate. Unfortunately, theoretically analyzing a process of this complexity is not feasible. Thus, there will naturally be discrepancies between the theory and experiments, which is why we focus mainly on qualitative evaluation and comparison of changes in the detection statistic due to JPEG compression.

Still images

Rather than choosing an artificial case with the camera fingerint estimated from one flat-content grayscale image we demonstrate the effect of compression on a more realistic and practical case with 12 digital cameras whose fingerprints were estimated from 20 natural 24-bit color images. Only for one camera out of 12 (Canon G2), the fingerprint was estimated from blue sky images.

Camera model	File format	Image resolution
Nikon D200 #1	PNG	2592× 3872
Nikon D200 #2	PNG	2592× 3872
Nikon D70 #1	PNG	2000 × 3008
Nikon D70 #2	PNG	2000 × 3008
Nikon D70s #1	PNG	2000 × 3008
Nikon D70s #2	PNG	2000 × 3008
Canon G2	TIFF	1704 × 2272
Kodak DC290	TIFF	1200 × 1792
Panasonic DMC ZS7	JPEG	3000 × 4000
Panasonic DMC FZ50	JPEG	2736 × 3648
Canon PowerShot S40	TIFF	1704 × 2272
Leica 100	TIFF	3468 × 5212

Table 2. List of cameras included in the experiment with still images.

To be able to control the JPEG compression, all images were acquired in either the RAW format or a high quality JPEG. The list of all 12 cameras is in Table 2. Images from Nikon cameras are from the Dresden Image Database and were converted from the RAW format to PNG in LightRoom 5.6 (on Export to 24-bit TIFF, no post-processing was selected except for "sharpen for screen"). We went with the highest quality JPEG compression for the Panasonic cameras since, like most consumer cameras, they do not support RAW format. Images from Canon G2 were flat-field type (blue sky). The same number of 20 images from each camera was used for the fingerprint estimation.

The estimation of camera fingerprints starts with wavelet denoising of noise with variance equal to 2 and subtracting the denoised image content. The estimate 3 of the PRNU is post-processed in three steps: zero-mean filter in each (RGB) color channel (i.e., subtract column means and row means from all elements), following by RGB-to-GRAY-like transform, and Wiener filter in Fourier domain. All 12 camera fingerprints make 66 pairs for the null hypothesis tests. In order to obtain matching dimensions and to create more pairs, we slice each fingerprint into non-overlapping 1 Mpixel chunks of size 840×1200 for Experiment #1 and to 1/4 Mpixel chunks of size 420×600 in the second set of tests for Experiment #2. The test statistic $\rho(\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j)$ and the PCE 8 were evaluated for each pair of fingerprint chunks $\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j$ from unmatched cameras. This process was repeated after JPEG compression of the original images with the quality factor in the set 30, 35, ..., 95, 100.

Experiment #1 (1 Mpixel fingerprint chunks)

After slicing 12 camera (grayscale) fingerprints into segments of 840×1200 pixels, we generated $N = \sum_{i \neq j} n_i n_j = 2338$ test pairs, where $n = [9, 9, 4, 4, 4, 4, 2, 1, 9, 9, 2, 16]$ is the number of segments from the 12 fingerprints. Because the number of pixels in each segment is approximately 1 million, the results of this experiment can be compared with the simulations on 1 million pixels (Figure 1). In particular, we will relate σ_{K}^2, the sample variance of $\rho(\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j)$ over

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

all 2,338 pairs i, j, as a function of the quality factor q with the variance $\sigma_\rho^2(0,0)$ obtained from theory. To this end, we need to estimate the variances σ_R^2 and σ_S^2. However, these parameters strongly vary not only between cameras and images but also locally within the images themselves. We estimated the variance σ_R^2 by first computing the pixel variances in each 8×8 (non-overlapping) block and averaged them over all blocks in the image. The variance σ_R^2 estimated this way varied between 60 and 220 across images and reached values as low as 1.8 for flat-field images. This means that the correlated signals are strongly non-stationary, which introduces a discrepancy with the theoretical assumptions made in the previous section. Despite these differences, Figure 5 reveals a rather close match between $\sigma_{\rm K}^2$ and the theoretical variance, which we obtained with $\sigma_R^2{=}\sigma_S^2 = 100$ as the median value over images. We attribute the slightly larger values of $\sigma_{\rm K}^2$ to hidden dependencies in the original images – notice that with no compression (denoted by '∞') $\sigma_{\rm K}^2(q)$ is still larger than $\sigma_\rho^2(0,0)$ at quality factor 100.

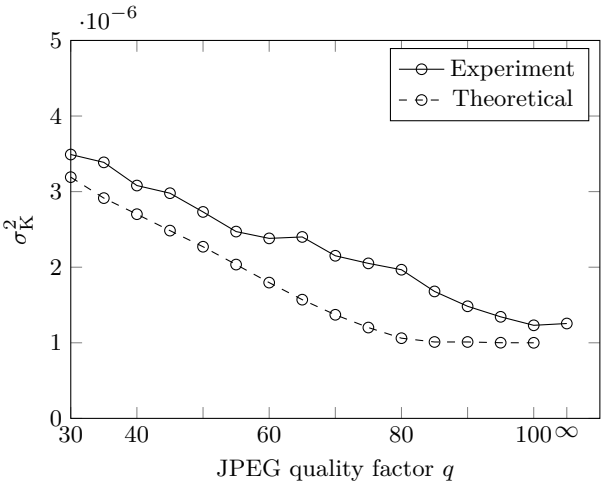

Figure 5. *Experimental $\sigma_{\rm K}^2$ and theoretical $\sigma_\rho^2(0,0)$ as a function of JPEG quality factor q for images with 840×1200 pixels. The result with uncompressed images is marked with '∞'.*

Continuing this experiment, we collected the values of the PCE for each pair $\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j$ and evaluated the sample variance $\mathrm{Var}[\mathrm{PCE}_{\rm K}]$ of $\mathrm{PCE}(\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j)$ across the pairs i, j as a function of the quality factor q. Figure 6 shows that this variance is also larger than what is obtained from the theory.

One consequence of the increased variance of the detection statistic under the null hypothesis is that the false-alarm probability increases with JPEG compression if the threshold is fixed. Knowing how the PCE depends on the quality factor allows us to adjust the detection threshold $T_{\rm PCE}$ to guarantee a given false alarm probability. For this purpose, we define a scaling factor γ,

$$\gamma(q) = \frac{\mathrm{Std}[\mathrm{PCE}_{\rm K}(q)]}{\mathrm{Std}[\mathrm{PCE}_{\rm K}(\infty)]}, \tag{21}$$

where $\mathrm{Std}[\mathrm{PCE}_{\rm K}(\infty)]$ is the standard deviation of the PCE obtained from pairs of uncompressed images. If all images in the test were compressed with the same quality factor q, then the threshold $\gamma(q) T_{\rm PCE}$ leads to the same false-alarm probability as $T_{\rm PCE}$ for uncompressed images. The scaling factor γ computed in this experiment is shown as one of the curves in Figure 7, in which we reversed the x-axis for better readability.

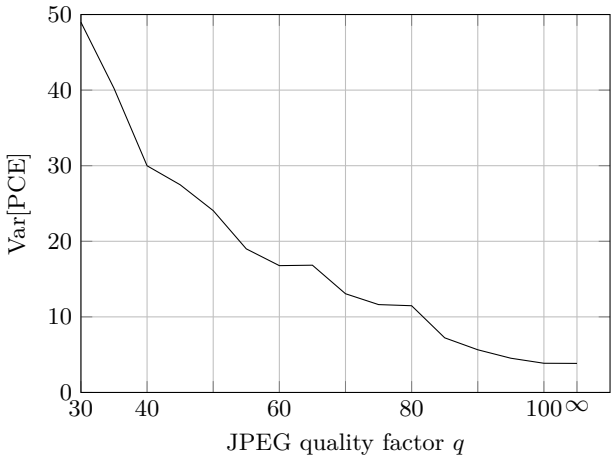

Figure 6. *Experimental variance of the PCE as a function of JPEG quality factor q for images with 840×1200 pixels.*

Experiment #2 (1/4 Mpixel fingerprint chunks)

The purpose of the next experiment is to verify the right tail model for the distribution of the PCE (the chi-square distribution) as this tail determines the false-alarm probability. To this end, we had to increase the statistical sample. Starting with the same 12 cameras and their fingerprints, we sliced them into smaller sizes of $420 \times 600{=}252,000$ pixels and obtained a total of 46,016 fingerprint pairs, which is more than 20 times more than in Experiment #1. Figure 8 shows the log-tail plot for the PCE, which appears compatible with the chi-square model. Note that the variance $\sigma_{\rm K}^2$ in Figure 9 is larger than the one obtained from 1 Mpixel images.

Table 3 shows the variance of the normalized cross-correlation $\rho(\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j, \tau_1, \tau_2)$ as a function of the spatial shift (τ_1, τ_2). Note that in this case, the match with the theoretical values (Table 1) is quite poor. In particular, the cross-correlation values for $\tau_1 = 4$ and $\tau_2 = 4$ are much larger than the rest of the values for similar shifts. This indicates the presence of periodic signals in the estimated fingerprints, perhaps due to some non-unique artifacts (NUAs), which are patterns that are not unique to a particular camera but may repeat across the same sensor types or the same camera models, such as those caused by color demosaicing.

To suppress the effect of these artifacts on the normalized cross-correlation, we repeated Experiment #2 with the images downsized by factor 1.7. The resizing makes the periodic NUAs out of synch with the 8×8 grid and thus the NUAs do not affect the detection statistic as much (Table 4). The variance at $\tau_2 = 0$ (no vertical shift) is larger

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

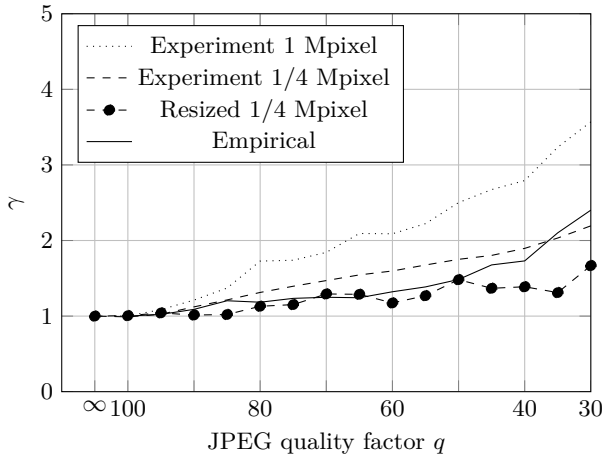

Figure 7. *Scaling factor γ as a function of JPEG quality factor q for 1Mp images, 1/4 Mp images and their scaled versions, and from Monte Carlo simulation on simplified model in the section on theory (empirical for 1000×1000). This factor is used for adjusting the threshold on PCE to meet a fixed false-alarm probability.*

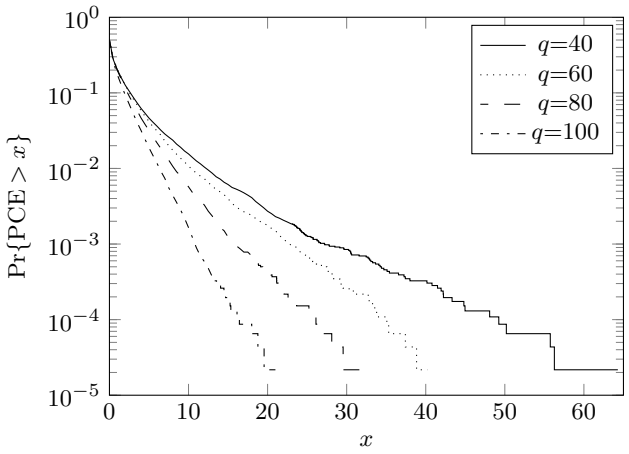

Figure 8. *Experimental distribution of the tail of the PCE, $\Pr\{\mathrm{PCE} > x\}$, for JPEG quality factor $q = 40, 60, 80, 100$, obtained from $46,016$ pairs of 420×600 PRNU estimates.*

than at $\tau_2 > 0$ in agreement with theoretical finding. The position of spikes, such as those at $(\tau_1, \tau_2) = (5, 4)$, depends on the resizing factor.

As the last result of Experiment #2, we report the scaling factor (21) for 1/4 Mp images and their resized versions, together with an empirically obtained scaling factor based on the variance from Figure 4.

(τ_1,τ_2)	0	1	2	3	4	5	6	7
0	6.3601	4.1065	5.7369	4.1299	6.5367	4.2075	6.0225	4.1783
1	4.7043	2.8641	4.1822	2.8852	4.8600	2.9451	4.4120	2.9217
2	6.2708	4.0243	5.7149	4.0443	6.4352	4.1232	5.9767	4.0971
3	4.7025	2.8659	4.1810	2.8858	4.8579	2.9473	4.4105	2.9225
4	6.3585	4.1067	5.7373	4.1299	6.5360	4.2071	6.0228	4.1799
5	4.7042	2.8637	4.1822	2.8850	4.8594	2.9449	4.4117	2.9214
6	6.2708	4.0247	5.7139	4.0444	6.4344	4.1225	5.9776	4.0973
7	4.7021	2.8654	4.1805	2.8858	4.8580	2.9471	4.4097	2.9227

Table 3. Experimental values of $\sigma_\rho^2(\tau_1, \tau_2)$ scaled by a factor of 10^6 as a function of (τ_1, τ_2) for JPEG quality factor 80 and images with 420×600 pixels.

(τ_1,τ_2)	0	1	2	3	4	5	6	7
0	4.3590	4.1194	3.9991	4.1988	4.2314	4.1933	3.9460	4.2097
1	4.7758	4.4610	4.4198	4.5787	4.7600	4.5902	4.3962	4.5575
2	4.3639	4.2198	4.0123	4.2916	4.2427	4.2735	3.9368	4.2910
3	4.2610	4.0463	3.9124	4.1424	4.1959	4.1357	3.8593	4.1367
4	4.3569	4.1201	3.9961	4.1923	4.2267	4.1945	3.9459	4.2126
5	4.7862	4.4670	4.4301	4.5824	4.7699	4.5982	4.4013	4.5637
6	4.3739	4.2242	4.0213	4.2978	4.2510	4.2820	3.9457	4.2965
7	4.2624	4.0474	3.9130	4.1437	4.1958	4.1415	3.8628	4.1385

Table 4. Experimental values of $\sigma_\rho^2(\tau_1, \tau_2)$ scaled by a factor of 10^6 as a function of (τ_1, τ_2) for JPEG quality factor 80 and images with 420×600 pixels downsampled by factor 1.7.

PRNU estimated from video

From our video library we selected 18 cameras whose videos had the highest bit rate (typically around 11,000 kbps) at the original SD resolution (640×480). For each camera, we chose 4 or 5 videos, 80 video files total, and worked with total of 3,019 unmatched video pairs of the

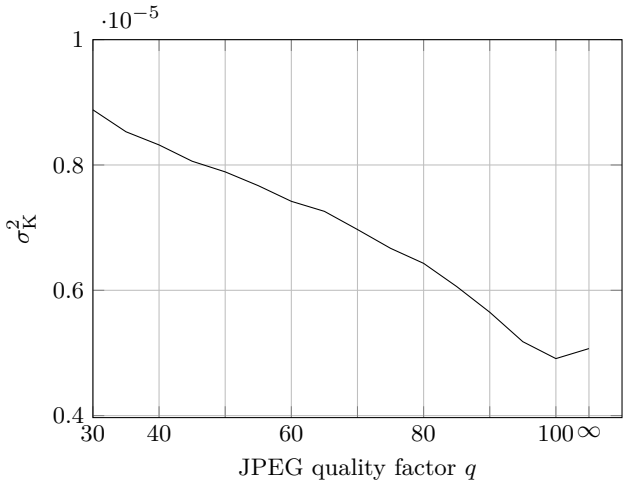

Figure 9. *Experimental σ_{K}^2 as a function of JPEG quality factor q for images of 420×600 pixels. The results were obtained from 46016 pairs of PRNU estimates.*

original resolution. We extracted the exact middle 1000 frames from each video and used VideoWriter in Matlab to transcode these 1000 video frames into MPEG-4 and the Motion JPEG format with quality settings 30, 45, 60, 75, and 90. The camera fingerprints were computed from videos in the same way as was done from images (each video frame making one image) with the exception of setting the wavelet denoising parameter sigma to 1.5. The post-processing was a little different. The first step included applying the zero-mean filter in each (RGB) color channel, followed by an RGB-to-GRAY-like transform, Wiener filtering in the Fourier domain, and applying the notch filter in the Fourier domain as found in reference [3] to remove periodic signals. Finally, we cropped out a 32-pixel boundary to removing occasional boundary artifacts, which gave us final fingerprints of size 416×576.

As seen in Figure 10, compression has the expected effect on the variance σ_{K}^2. It is important to realize that

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

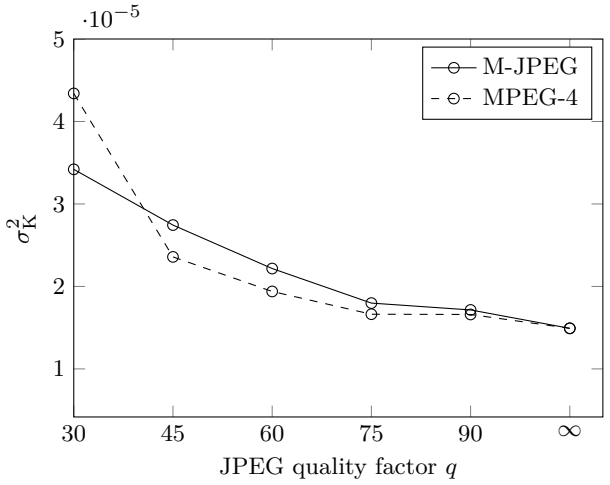

Figure 10. *Experimental σ^2_{K} as a function of JPEG quality factor q for SD video at resolution 416×576 pixels. Compression was achieved using VideoWriter in Matlab. The results were obtained from $3,019$ pairs of PRNU estimates.*

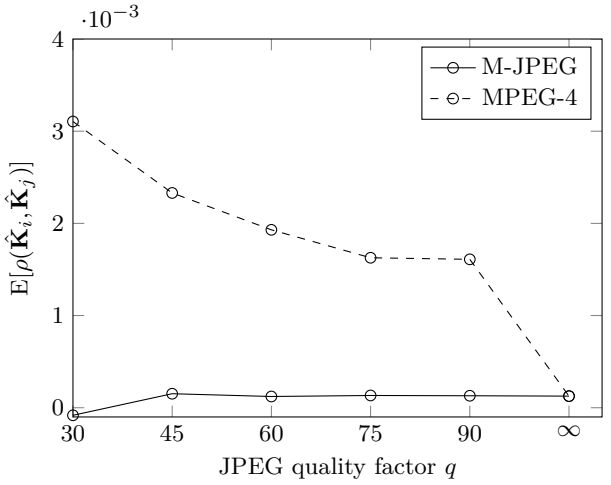

Figure 11. *Experimental mean $E[\rho(\hat{\mathbf{K}}_i, \hat{\mathbf{K}}_j)]$ as a function of JPEG quality factor q for SD video at resolution 416×576 pixels. Compression was achieved using VideoWriter in Matlab. The results were obtained from $3,019$ pairs of PRNU estimates.*

the MPEG-4 with quality factor q provides much more severe data compression that JPEG compression at the same quality factor. Nevertheless, comparing experimental σ^2_{K} in Figure 10 with the same quantity in Figure 9 for slightly larger images we notice that the normalized correlation σ^2_{K} computed from video files is a few times larger, including the original video (denoted by the ∞ symbol). We attribute this difference mostly to the agressive filtering with the notch filter that decreases the effective signal dimension in the frequency domain by setting a large number of cefficients to zero. (The difference between the size of image segments and dimensions of videos only contributes with about 5%.) There is an unexpected positive mean of the correlation statistic after MPEG-4 compression (Figure 11). The setup of this test proves that the positive bias in the detection statistic ρ after MPEG-4 compression is not caused by the video content or the PRNU estimation process. The bias was introduced during the compression, either in the video codec or in the VideoWriter application.

Conclusions

The main goal of this paper is to investigate the effect of JPEG compression on the detection statistic used in camera identification using sensor fingerprints. According to the best knowledge of the authors, this topic is quite important for practical deployment of sensor-fingerprint based camera identification, yet it has not been addressed in the literature.

First, we study this effect theoretically under simplifying assumptions, and then we execute several tests with real images and video clips in the MPEG-4 and M-JPEG formats. The theoretical analysis, verified by Monte Carlo simulations, is in agreement with the experiments – JPEG compression increases the variance of the normalized correlation as well as the variance of the PCE. Consequently, the decision threshold needs to be adjusted to guarantee a prescribed false-alarm probability. We computed this adjusting factor from experiments with 1 Mpixel still images compressed with a range of JPEG quality factors. For example, with JPEG quality 60, the threshold for PCE needs to be approximately doubled.

Apart from the experiment with JPEG compression of images, the experiment with video compression revealed that not only variance of the normalized correlation depends on the compression quality but in case of MPEG-4 a positive bias increases the normalized correlation as well. We leave a deeper study of this phenomenon for future research. We also plan to put the findings of this paper under scrutiny in a large scale test on video files.

The Matlab code used to generate the data in plots in this paper is available from `http://dde.binghamton.edu/download`.

Acknowledgments

The work on this paper was supported by Rome Research Corporation under subcontract number RRC SC-N11097-12 and Air Force Office of Scientific Research under the research grant number FA9950-12-1-0124. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of AFOSR or the U.S. Government.

References

[1] C. Chen and Y. Q. Shi. JPEG image steganalysis utilizing both intrablock and interblock correlations. In *Circuits and Systems, ISCAS 2008. IEEE International Symposium on*, pages 3029–3032, Seattle, WA, May, 18–21, 2008.

[2] M. Chen, J. Fridrich, and M. Goljan. Digital imaging sen-

DOI: 10.2352/ISSN.2470-1173.2016.8.MWSF-086

sor identification (further study). In E.J. Delp and P.W. Wong, editors, *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 0P 1–12, San Jose, CA, January 2007.

[3] M. Chen, J. Fridrich, and M. Goljan. Source digital camcorder identification using ccd photo response nonuniformity. In E.J. Delp and P.W. Wong, editors, *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 1G 1–12, San Jose, CA, January 2007.

[4] G. Chierchia, D. Cozzolino, G. Poggi, C. Sansone, and L. Verdoliva. Guided filtering for prnu-based localization of small-size image forgeries. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6231–6235, Florence, Italy, May 2014.

[5] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva. A bayesian-mrf approach for prnu-based image forgery detection. *Information Forensics and Security, IEEE Transactions on*, 9(4):554–567, April 2014.

[6] D. Cozzolino, D. Gragnaniello, and L. Verdoliva. Image forgery localization through the fusion of camera-based, feature-based and pixel-based techniques. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5302–5306, Paris, France, Oct 2014.

[7] J. Fridrich. Digital image forensic using sensor noise. *IEEE Signal Processing Magazine*, 26(2):26–37, 2009.

[8] Z. Geradts. Linking cameras to images and videostreams with pixel response non-uniformity. *Science & Justice*, 50(1):44, 2010. Special Issue: 5th Triennial Conference of the European Academy of Forensic Science. Knowledge, Research, and Leadership in Forensic Science.

[9] Floris Gisolf, Anwar Malgoezar, Teun Baar, and Zeno Geradts. Improving source camera identification using a simplified total variation based noise removal algorithm. *Digital Investigation*, 10(3):207–214, 2013.

[10] M. Goljan, T. Filler, and J. Fridrich. Large scale test of sensor fingerprint camera identification. In N.D. Memon, E.J. Delp, P.W. Wong, and J. Dittmann, editors, *Proc. SPIE, Electronic Imaging, Media Forensics and Security XI*, volume 7254, pages 0I 01–12, San Jose, CA, January 19–21, 2009.

[11] C. R. Holt. Two-channel detectors for arbitrary linear channel distortion. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-35(3):267–273, March 1987.

[12] Jong-Uk Hou, Han-Ul Jang, and Heung-Kyu Lee. Hue modification estimation using sensor pattern noise. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5287–5291, Paris, France, Oct 2014.

[13] Dai-Kyung Hyun, Seung-Jin Ryu, Min-Jeong Lee, Jun-Hee Lee, Hae-Yeoun Lee, and Heung-Kyu Lee. Source camcorder identification from cropped and scaled videos. In A. Alattar, N. D. Memon, and E. J. Delp, editors, *Proc. SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics XIV*, volume 8303, pages 0E 1–8, January 23–26, 2012.

[14] J. R. Janesick. *Scientific Charge-Coupled Devices*, volume PM83. SPIE Press Monograph, 2001.

[15] Xiangui Kang, Yinxiang Li, Zhenhua Qu, and Jiwu Huang. Enhancing source camera identification performance with a camera reference phase sensor pattern noise. *IEEE Transactions on Information Forensics and Security*, 7(2):393–402, 2012.

[16] Chang-Tsun Li and Yue Li. Color-decoupled photo response non-uniformity for digital image forensics. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(2):260–271, Feb 2012.

[17] J. Lukáš, J. Fridrich, and M. Goljan. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, June 2006.

[18] Wiger van Houten and Zeno J. Geradts. Source video camera identification for multiply compressed videos originating from youtube. *Digital Investigation*, 6(1-2):48–60, 2009.

[19] Wiger van Houten and Zeno J. Geradts. Using sensor noise to identify low resolution compressed videos from youtube. In Zeno J. Geradts, Katrin Y. Franke, and Cor. J. Veenman, editors, *International Workshop on Computational Forensics*, pages 104–115, The Hague, Netherlands, 2009.

Author Biography

Miroslav Goljan received the Ph.D. degree in Electrical Engineering from Binghamton University in 2002 and the M.S. in Mathematical Informatics from Charles University in Prague, Czech Republic, in 1984. He is Research Scientist at the Dept. of Electrical and Computer Engineering at Binghamton University, Binghamton, NY. His research focuses on digital camera and digital image forensics, steganography, steganalysis, and reversible data hiding in digital media.

Mo Chen received his Ph.D. in electrical engineering from Binghamton University, NY in 2006. From 2006 to 2010, he was Research Scientist at Research Foundation of SUNY in Binghamton, NY. Since 2007, he has worked as Principle Machine Vision Engineer at JADAK LLC, NY (GSI Group). His research interests include machine vision and machine learning, digital image and video processing, and digital forensic.

Pedro Comesaña-Alfaro received both the Telecommunications Engineering (specialized in both Computer Science and Communications) and Ph. D degrees from the University of Vigo, Spain, in 2002 and 2006, respectively. Since 2012 he is Associate Professor in the School of Telecommunications Engineering, University of Vigo. His research interests lie in the areas of digital watermarking, information security, multimedia forensics and digital communications. He received the Best Paper Award of IEEE-WIFS 2014 and IWDW 2011.

Jessica Fridrich is Professor of Electrical and Computer Engineering at Binghamton University. She received her PhD in Systems Science from Binghamton University in 1995 and MS in Applied Mathematics from Czech Technical University in Prague in 1987. Her main interests are in steganography, steganalysis, and digital image forensic. Since 1995, she has received 20 research grants totaling over $9 mil that lead to more than 160 papers and 7 US patents.