Detecting Digital Image Forgeries Using Sensor Pattern Noise

Jan Lukáš, Jessica Fridrich, and Miroslav Goljan
Department of Electrical and Computer Engineering
SUNY Binghamton, Binghamton, NY 13902-6000

ABSTRACT

We present a new approach to detection of forgeries in digital images under the assumption that either the camera that took the image is available or other images taken by that camera are available. Our method is based on detecting the presence of the camera pattern noise, which is a unique stochastic characteristic of imaging sensors, in individual regions in the image. The forged region is determined as the one that lacks the pattern noise. The presence of the noise is established using correlation as in detection of spread spectrum watermarks. We proposed two approaches. In the first one, the user selects an area for integrity verification. The second method attempts to automatically determine the forged area without assuming any a priori knowledge. The methods are tested both on examples of real forgeries and on non-forged images. We also investigate how further image processing applied to the forged image, such as lossy compression or filtering, influences our ability to verify image integrity.

Keywords: Fixed pattern noise, digital forgery, digital forensic, pattern noise, pixel non-uniformity

1. INTRODUCTION

The practice of forging photographs is probably as old as the art of photography itself. Digital photography and powerful image editing software made it very easy today to create believable forgeries of digital pictures even for a non-specialist. As digital photography continues to replace its analog counterpart, the need for reliable detection of digitally doctored images is quickly increasing. Verifying the content of digital images or identifying forged regions would be obviously useful for instance in the court of law, when digital pictures are presented as evidence.

Recently, several different methods for detecting digital forgeries were proposed. Ng and Chang proposed a method for detection of photomontages [1]. Popescu et al. developed several methods for identifying digital forgeries by tracing artifacts introduced by resampling [2] and Color Filter Array (CFA) interpolation [3]. Recently, Johnson et al. [4] proposed another method based on inspecting inconsistencies in lighting conditions. In [5], Fridrich et al. established a method for detecting copy-move forgeries (a similar method was later proposed by Popescu et al. [6]). For each of these methods, there are circumstances when they will fail to detect a forgery. Method [1], for instance, has very restricting assumptions that are usually not fulfilled and even when they are satisfied, its misclassification rate is almost 28%. The copy-move detection method [5] is limited to one particular case of forgeries, when a certain part of the image was copied and pasted somewhere else in the same image (e.g., to cover an object). Methods based on detecting traces of resampling may produce less reliable results for processed images stored in the JPEG format. The method based on detection of inconsistencies in lighting assumes nearly Lambertian surface for both the forged and original areas and might not work when the object does not have a compatible surface, when pictures of both the original and forged objects were taken under approximately similar lighting conditions, or during a cloudy day when no directional light source was present.

Obviously, the problem of detection of digital forgeries is a complex one with no universally applicable solution. What is needed is a set of different tools that can be all applied to the image at hand. The decision about the content authenticity is then reached by interpreting the results obtained from different approaches. This accumulative evidence may provide a convincing enough argument that each individual method cannot.

{jan.lukas, fridrich, mgoljan}@binghamton.edu; phone 1 607 777-6177; fax 1 607 777-4464;
http://www.ws.binghamton.edu/fridrich

In this paper, we propose a new method for detection of digitally manipulated images based on the sensor pattern noise that each camera involuntarily inserts into each image it takes. The method is applicable whenever we are in a situation when the forged image is claimed to have been taken by a camera that we have in possession or at least, we have other images taken by the camera. Because the pattern noise appears to be a unique stochastic fingerprint of digital imaging sensors [7], [8], forged areas could be identified by verifying the consistency of their noise residual with the pattern noise of the particular sensor element.

In the next section, we review the properties of the pattern noise and methods for its detection in images. In Section 3, we describe a supervised algorithm for verifying the integrity of a given Region of Interest (ROI), e.g., when some a priori information about the forged area exists. In Section 4, we present a more general unsupervised algorithm for automatic identification of forged areas that can be subsequently confirmed by the supervised algorithm. The last section contains a summary, discussion of limitations, and outline of future directions.

2. PATTERN NOISE & DETECTION OF ITS PRESENCE

2.1 Signal processing in digital cameras

In this section, we briefly describe the processing stages inside a typical digital camera and discuss various imperfections that inevitably enter the image acquisition process. We focus on the sensor pattern noise.

The heart of every digital camera is the imaging sensor. The sensor is divided into very small minimal addressable picture elements (pixels) that collect photons and covert them into voltages that are subsequently sampled to a digital signal in an A/D converter. Before the light from the photographed scene reaches the sensor, however, it passes through the camera lenses, an antialiasing (blurring) filter, and then through a color filter array (CFA) [9]. The CFA is a mosaic of color filters that block out a certain portion of the spectrum, allowing each pixel to detect only one specific color. The Foveon™ X3 sensor is the only sensor that does not use CFA and is able to capture all three basic colors at every pixel.

If the sensor uses a CFA, the digitized sensor output is further interpolated (demosaicked) using color interpolation algorithms to obtain all three basic colors for each pixel. The resulting signal is then further processed using color correction and white balance adjustment. Additional processing includes gamma correction to adjust for the linear response of the imaging sensor and kernel filtering to visually enhance the final image. Finally, the digital image is written to the camera memory device in a user-selected image format. This may require additional processing, such as JPEG compression.

2.2 Pattern noise

There are several sources of imperfections and noise that influence the image acquisition process. When the imaging sensor takes a picture of an absolutely evenly lit scene, the resulting digital image will still exhibit small changes in intensity among individual pixels. This is partly because of random components, such as *readout noise* or *shot noise* (also known as *photonic noise* [10], [11]), and partly because of the *pattern noise*, which is a deterministic component that stays approximately the same if multiple pictures of the same scene are taken. Due to this property, the pattern noise is present in every image the sensor takes. In this paper, we use this noise for identification of forgeries. Note that while averaging multiple images reduces the random components, it enhances the pattern noise.

The two main components of the pattern noise are the *fixed pattern noise* (FPN) and the *photo-response non-uniformity noise* (PRNU)[1]. The fixed pattern noise (FPN) refers to pixel-to-pixel differences when the sensor array is *not* exposed to light (so called dark current). The FPN is an additive noise and some middle to high-end consumer cameras suppress this noise by subtracting a dark frame from every image they take. The FPN also depends on exposure and temperature.

The photo-response non-uniformity noise (PRNU) is the dominant part of the pattern noise in natural images. Before we discuss it further, we give a mathematical model of the image acquisition process. Let us denote the raw signal that would be registered by the sensor due to incoming light (if no other sources of noise existed) as $\boldsymbol{x} = (x_{ij})$,

[1] Different authors use different terminology when describing noise sources. We use the same terminology as [10]. Note that, for example, FPN means something else in [11].

$i = 1, \ldots, m, j = 1, \ldots, n$, where $m \times n$ is the sensor resolution. Denoting the random shot noise as $\boldsymbol{\eta} = (\eta_{ij})$, the additive random noise (represented by the read-out noise, etc.) as $\boldsymbol{\varepsilon} = (\varepsilon_{ij})$, and the dark current as $\boldsymbol{c} = (c_{ij})$, the digitized output of the sensor $\boldsymbol{y} = (y_{ij})$ can be expressed in the following form (before any other camera processing occurs)

$$y_{ij} = f_{ij}\,(x_{ij} + \eta_{ij}) + c_{ij} + \varepsilon_{ij}. \tag{1}$$

The factors f_{ij} are close to 1 and capture the PRNU noise, which is a multiplicative noise. The most important component of PRNU is the *pixel non-uniformity* (PNU), which is defined as different sensitivity of pixels to light. The PNU is caused by stochastic inhomogenities present in silicon wafers and other imperfections imposed during the sensor manufacturing process. As such, it is not dependent on ambient temperature and appears to be stable over time. The power spectrum of the PNU is continuous [12] with slightly attenuated high spatial frequencies. Light refraction on dust particles and optical surfaces and properties of the camera optics also contribute to the PRNU noise. These components are known as "doughnut" patterns and vignettage and are of low spatial frequency in nature [11]. Because these low frequency components are not a characteristic of the sensor, we do not use them for identification of forgeries. We only use the PNU component, which is an intrinsic characteristic (fingerprint) of the sensor.

The PRNU noise is not present in completely saturated areas of an image, where all sensor pixels were filled to a full capacity producing a constant signal. It is also clear from (1) that in very dark areas (when $x_{ij} \approx 0$) the PRNU noise is largely suppressed.

The signal $\boldsymbol{y}$ goes through a chain of complex processing before the final image file is stored on the camera's memory card. This processing includes operations on a local neighborhood of pixels, such as demosaicking, color correction, or kernel filtering. Some operations may be non-linear in nature, such as gamma correction, white balance, or adaptive color interpolation. The final pixel values p_{ij}, which we will assume to be in the range $0 \le p_{ij} \le 255$ for each color channel, are

$$p_{ij} = T(y_{ij}, \mathcal{N}(y_{ij}), i, j), \tag{2}$$

where T is a non-linear function of y_{ij}, the pixel location (i, j), and values y from a local neighborhood $\mathcal{N}(y_{ij})$ (e.g., the 5×5 neighborhood).

It is possible to suppress the pattern noise using a process called flat fielding [10], [11], in which the pixel values are first corrected for the additive FPN and then divided by a flat field frame obtained by averaging images of a uniformly lit scene. This process cannot be done correctly from the final pixel values p_{ij} and *must* be performed on the raw sensor output $\boldsymbol{y}$ before any further image processing. While it is commonly done for astronomical imaging, consumer digital cameras do not flat-field their images because it is difficult to achieve a uniform sensor illumination inside the camera.

Because essentially all imaging sensors (CCD, CMOS, JFET, or CMOS-Foveon™ X3) are built from semiconductors and their manufacturing techniques do not differ too much, the pattern noise in all these sensors has similar properties. As noted in [10] (page 92), CMOS sensors also experience both FPN and PRNU. Moreover, according to [13] the PRNU noise strength is comparable for both CMOS and CCD detectors. Because the JFET sensors are similar to CMOS sensors, they should behave similarly.

2.3 Detection of pattern noise (forgery identification)

Forged regions can be identified as those lacking the pattern noise. We detect the presence of the PNU noise in a region by calculating the correlation between the region noise residual and the pattern noise. Thus, forgery detection must start with determining the reference pattern for the camera. We describe this process first and then present the forgery identification algorithm.

For a given camera C, we obtain an approximation $\boldsymbol{P}_C$ to the camera reference pattern noise by averaging multiple images $\boldsymbol{p}^{(k)}$, $k = 1, \ldots, N_p$. This process can be sped up by suppressing the scene content from the image prior to averaging, which can be achieved using a denoising filter F and averaging the noise residuals $\boldsymbol{n}^{(k)}$ instead

$$\boldsymbol{n}^{(k)} = \boldsymbol{p}^{(k)} - F(\boldsymbol{p}^{(k)}). \tag{3}$$

Additional benefit of denoising is that the low-frequency components of PRNU (e.g., patterns due to light refraction on dust particles) are automatically removed. Obviously, the larger the number of images N_p, the more we suppress the impact of random noise components and the scene. In our experiments, we used $N_p = 300$ and we recommend using $N_p > 50$, if possible. The advantage of this method for obtaining the reference pattern is that it is applicable to all cameras and does not require access to the camera (images from the camera are sufficient).

We have experimented with several denoising filters F and eventually decided to use the filter described in [14] because it gave us the best experimental results. This is likely because the noise residual obtained using this particular filter contains the least amount of traces of the scene (areas around edges are usually misinterpreted by less sophisticated denoising filters, such as the Wiener filter or the median filter).

To decide whether a selected region $\mathcal{R}$ in image $\boldsymbol{p}$ is compatible with the pattern noise from camera C, we first calculate the correlation between the noise residual $\boldsymbol{n} = \boldsymbol{p} - F(\boldsymbol{p})$ with the camera reference pattern $\boldsymbol{P}_C$

$$\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R})) = \frac{(\boldsymbol{n}(\mathcal{R}) - \bar{\boldsymbol{n}}(\mathcal{R})) \cdot (\boldsymbol{P}_C(\mathcal{R}) - \bar{\boldsymbol{P}}_C(\mathcal{R}))}{\left\|\boldsymbol{n}(\mathcal{R}) - \bar{\boldsymbol{n}}(\mathcal{R})\right\| \left\|\boldsymbol{P}_C(\mathcal{R}) - \bar{\boldsymbol{P}}_C(\mathcal{R})\right\|}, \tag{4}$$

where $\boldsymbol{n}(\mathcal{R})$ and $\boldsymbol{P}_C(\mathcal{R})$ denote $\boldsymbol{n}$ and $\boldsymbol{P}_C$ constrained to $\mathcal{R}$ and written as vectors. The bar above symbols denotes the mean value, "$\cdot$" is the usual dot product, and $\|.\|$ is the L_2 norm.

Because the PRNU noise is multiplicative (see (1)), it is absent in completely saturated areas and largely suppressed in dark areas. Also, the denoising filter is less effective in removing the noise in highly textured areas with many fine edges. In such regions, the correlation $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$ will be naturally lower and care must be taken not to misinterpret such regions as tampered.

We may attempt to estimate the distribution of correlations $\rho(\boldsymbol{n}(\mathcal{Q}), \boldsymbol{P}_C(\mathcal{Q}))$ for regions $\mathcal{Q}$ in the same image that are disjoint with $\mathcal{R}$, have similar histogram (and thus saturation), and, if possible, similar texture. Then, we could reach a decision about the authenticity of $\mathcal{R}$ based on the statistical significance of the value $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$. The biggest problem with this approach is that there may not be sufficiently many such regions in the image to correctly estimate the distribution. Moreover, if there are other unsuspected regions in the image that were tampered, this approach may fail completely. The alternative is to calculate the statistical evidence that $\mathcal{R}$ *was* tampered. The advantage here is that we can always collect a large statistical sample necessary to evaluate the statistical significance of $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$. Here is how we proceed.

We calculate the correlations $\rho(\boldsymbol{n}(\mathcal{Q}_k), \boldsymbol{P}_C(\mathcal{R}))$ for regions $\mathcal{Q}_k$, $k = 1, \ldots, N_R$ of the same size and shape coming from other cameras or from the same camera but at a different location within the image. The regions $\mathcal{Q}_k$ serve as a sample of regions that do not contain the reference pattern $\boldsymbol{P}_C(\mathcal{R})$. We model the distribution of $\rho(\boldsymbol{n}(\mathcal{Q}_k), \boldsymbol{P}_C(\mathcal{R}))$, $k = 1, \ldots, N_R$, with the generalized Gaussian distribution[2] with cumulative distribution function $G(x)$. Using this model, the probability that a generalized Gaussian random variable with the estimated distribution will attain the value $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$ or larger is

$$p = 1 - G\left(\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))\right). \tag{5}$$

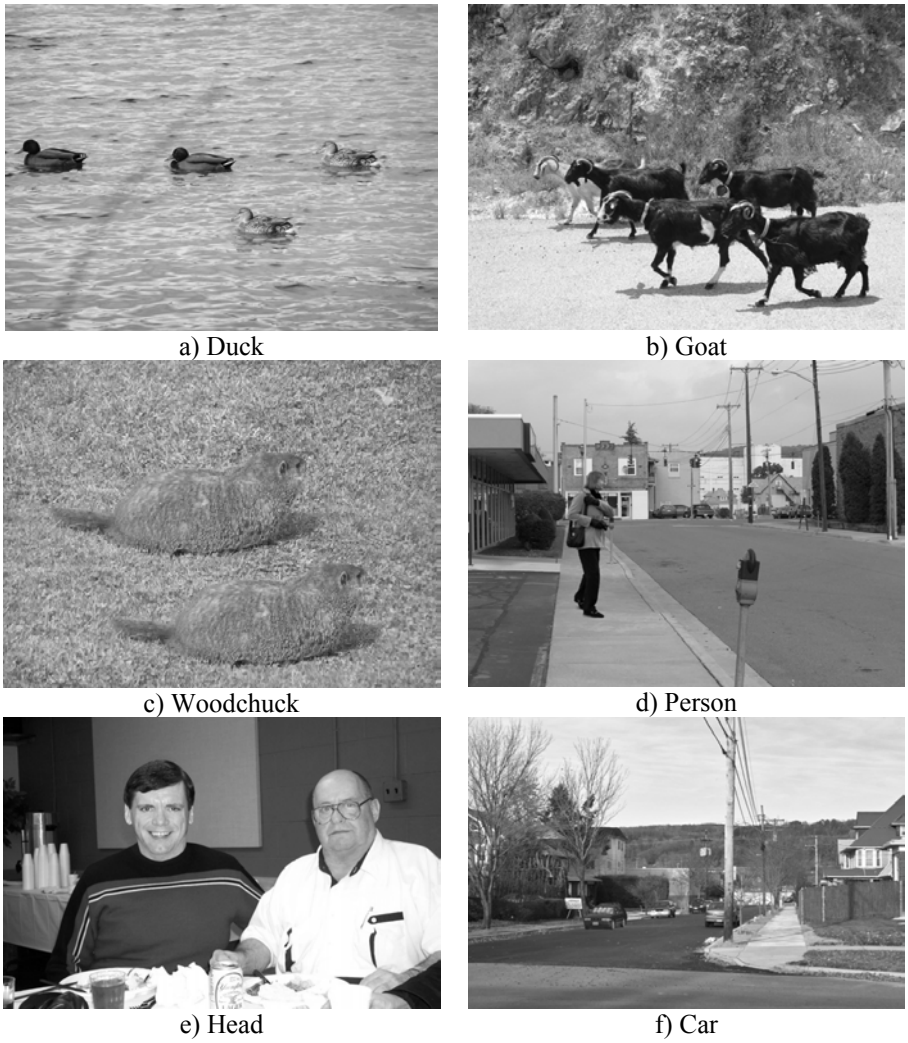
This p-value will be taken as the measure of statistical significance to evaluate the authenticity of the ROI $\mathcal{R}$. In particular, we decide that $\mathcal{R}$ was tampered if $p > \alpha = 10^{-3}$ and not tampered otherwise.

If the algorithm above decides that $\mathcal{R}$ was tampered, we need to validate the decision by applying the same algorithm to other regions $\mathcal{Q}$ in the same image that are disjoint with $\mathcal{R}$, have the same number of pixels, and, if possible, similar histogram. All such regions should be evaluated as non-tampered (equivalently, their p-value $p \leq \alpha$).

[2] For parameter estimation, we used the method of moments [16].

3. VERIFICATION OF A SELECTED ROI

In this section we study the problem of verifying the integrity of a selected ROI. For example, the ROI may have been determined manually by an operator as an area whose integrity is in question. Alternatively, the ROI could have been identified by one of the forgery-detection detection techniques capable of localizing the tampering ([2], [3], [5], [6]) and we wish to strengthen our evidence.



a) Duck b) Goat

c) Woodchuck d) Person

e) Head f) Car

Figure 1: Six digital forgeries.

For our experiments, we prepared 6 different forgeries, striving for a natural appearance of each forgery. The forged objects were pasted either from the same image, from another image from the same camera, or from an image from a completely different source. In some cases, the pasted objects had to be resampled to adjust their size. The original images were true color images either in the raw TIFF format or the JPEG format with varying quality factors. In one case, the original image was a low-resolution low-quality JPEG (the "goat" forgery below).

In Figure 1, we show all six forgeries. Figure 1 a) is a copy-move forgery created from a 1712×2288 Olympus C765 TIFF image. The female duck on the far right is a copy of the other female duck in the image. Figure 1 b) is a forgery created from a low-resolution low-quality (non-standard JPEG quantization matrix with the quality factor of approximately 73) Canon PowerShot G2 1200×1600 JPEG image. The goat in the lower right corner was pasted from

approximately the same quality image from the same camera taken a few seconds before. The "woodchuck" forgery shown in Figure 1 c) is another copy-move forgery created from a 1712×2288 Olympus C765 TIFF image. The woodchuck at the bottom is a copy of the other woodchuck in the image. The tampered area is relatively large. Figure 1 d) was created from a 1712×2288 Olympus C765 JPEG image of approximate JPEG quality factor 87. The person was pasted from a Canon G2 picture. Figure 1 e) is a forgery created from a 1200×1792 Kodak DC290 JPEG image of an unknown quality factor. The head of the person on the left was cut from another image taken with the same camera and pasted into the image since the person did not like his original appearance. Figure 1 f) shows a forgery created from a 1512×2268 Sigma SD9 TIFF image. The car parked on the left was cut from an Olympus C765 image. All forgeries were saved as TIFF images except for the "head" forgery, which was saved with a customized JPEG quantization matrix (considering only the five lowest frequency DCT coefficients, its quality factor was approximately 65).

We further note that some forged objects (e.g., the duck, goat, woodchuck, and head) were pasted from images taken under almost identical conditions (if not the same image) and from the same angle. Thus, they should resist attempts to expose them by tracing lighting inconsistencies [4]. The duck and the woodchuck were not resized, while the pasted objects from the other four forgeries were slightly resized. Thus, the duck and the woodchuck cannot be detected by inspecting traces of resampling [2] but could be probably detected using the copy-move detection method [5], [6]. Due to resizing and/or JPEG compression, it is also likely that the method based on detecting traces of CFA interpolation [3] might not apply to the "person," "head," and "goat" forgeries. On the other hand, because most pasted objects were resized and the forgeries saved as TIFFs, the method of [2] would likely be applicable to the "goat," "person," and "car" forgeries. We point out that since the "head" forgery was created from JPEG images and saved as relatively low-quality JPEG, it would not be exposed by any of the previously introduced methods. We now analyze all six forgeries using the proposed method.

3.1 ROI verification algorithm

We follow the procedure described in Section 2.3. After selecting the ROI $\mathcal{R}$, we first calculate the correlation $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$ and then the correlations $\rho(\boldsymbol{n}(Q_k), \boldsymbol{P}_C(\mathcal{R}))$, $k = 1, \ldots, N_R$, for regions Q_k of the same size as $\mathcal{R}$ coming from 90 non-tampered images obtained using 9 different digital cameras (10 from each digital camera). In each image, we randomly selected 100 regions, thus obtaining total of $90\times100=9000=N_R$ regions Q_k. The generalized Gaussian fit[3] to $\rho(\boldsymbol{n}(Q_k), \boldsymbol{P}_C(\mathcal{R}))$ constitutes a model of correlations for a forged region. Using this statistical model, we calculate the p-value (5), which is the probability that $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$ would be observed if it was a sample drawn from the estimated generalized Gaussian distribution (i.e., if it was forged). If $p > \alpha = 10^{-3}$, we do not reject our statistical hypothesis that the ROI $\mathcal{R}$ is forged.

If the ROI is deemed forged, we validate this decision by applying the same algorithm to 200 regions Q in the same image that are disjoint with $\mathcal{R}$ and have the same number of pixels and similar histogram. All these regions should be detected as non-forged.

As a final note, for color images we compute the correlation for each color channel separately and take as the final correlation value the maximum of these three numbers (because we expect the pattern noise in each channel).

3.2 Experiments

For each forgery, the ROI was selected manually (see two examples in Figure 2) and the algorithm described in Section 3.1 was applied. To find out how our ability to verify forged areas is influenced by lossy compression, we repeated the same process after recompressing the forged images using JPEG compression with quality factors ranging from 70 to 100. Table 1 shows the p-values (5) for the ROIs from all six forgeries. We conclude that, at the significance level $\alpha = 10^{-3}$, all six ROIs were correctly identified as tampered.

---

[3] For illustration, in Figure 3, we show the histogram of $\rho(\boldsymbol{n}(Q_k), \boldsymbol{P}_C(\mathcal{R}))$ and its generalized Gaussian fit for the (uncompressed) duck forgery.

Figure 2: Regions of interest for the duck and goat forgeries are displayed in white.

p-value	JPEG quality factor			
	100	90	80	70
Duck	0.60	0.80	0.91	0.84
Goat	0.82	0.89	0.59	0.67
Woodchuck	2.6×10^{-2}	0.11	0.08	0.18
Person	2.0×10^{-2}	8.8×10^{-3}	7.3×10^{-3}	4.8×10^{-3}
Head	0.47	0.63	0.39	0.75
Car	0.76	0.92	0.75	0.86

Table 1: *p*-values for forged regions as a function of JPEG compression quality factor.

We validated the decision for each image by running the same algorithm on 200 regions Q in the same forged image that are disjoint with $\mathcal{R}$ and have the same number of pixels and similar histogram. In all cases and for all six forgeries, the *p*-value for these regions was below α, validating our conclusion about the region $\mathcal{R}$. For illustration, in Figure 4 we show the scatter plot of the correlations $\rho(\boldsymbol{n}(Q), \boldsymbol{P}_C(Q))$ for the duck forgery after saving the forged image as uncompressed TIFF and as JPEG with various quality factors (the value $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$ is also shown). Note that the correlations gradually decrease with JPEG compression.

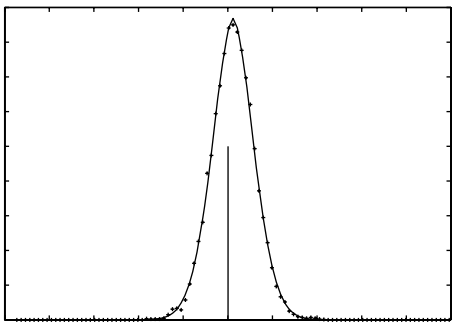


Figure 3: Histogram of $\rho(\boldsymbol{n}(Q_k), \boldsymbol{P}_C(\mathcal{R}))$ and its generalized Gaussian fit for the uncompressed duck forgery. The vertical line denotes the correlation in the region of interest, $\rho(\boldsymbol{n}(\mathcal{R}), \boldsymbol{P}_C(\mathcal{R}))$.

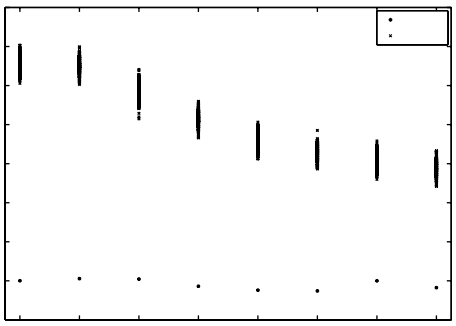


Figure 4: Correlation values in the ROI and other disjoint regions in the JPEG-compressed forged image for various JPEG quality factors.

4. AUTOMATIC ROI IDENTIFICATION

A more useful version of the forgery identification algorithm would identify the forged area automatically. In this section, we discuss and analyze some preliminary results of our effort to identify the ROI automatically without

assuming any prior information about its location, shape, or size. Once the region is identified, it is inspected using the algorithm of Section 3.1.

If there is a tampered area in the image, we assume that it is sufficiently large and compact but we have no information about its location, size, or shape. We know that the correlation in the tampered area is expected to be significantly lower compared to the rest of the image.

We experimented with sliding square blocks (e.g., 128×128) and computed the correlation at each location. However, this approach can only provide good performance for areas significantly larger than the sliding block. In particular, it poorly detects long but thin areas, areas with holes, and other areas with complicated shapes. Thus, we decided to use sliding blocks of different sizes and shapes. To speed up the algorithm, we do not use sliding blocks but partially overlapping blocks (overlapping approximately by 50–75%).

Automatic identification of forged regions

1. Prepare N blocks of different types (e.g., the blocks in Figure 5).
2. For each block type $i \in \{1, \ldots, N\}$, compute the correlation of the image noise residual with the camera reference pattern in overlapping blocks across the entire image. There will be total of n_i blocks and correlations $c_j, j = 1, \ldots, n_i$.
3. For each block type i, select m blocks $i_1, \ldots, i_m$ with the smallest correlation c_{i_k}, $k = 1, \ldots, m$. There will be total of $m \times N$ blocks $\mathcal{B}_k$.
4. Construct the mask $\mathcal{B} = \bigcup_{k=1}^{m \times N} \mathcal{B}_k$.
5. For each pixel $p \in \mathcal{B}$, let $t(p) = |\{\mathcal{B}_k | p \in \mathcal{B}_k\}|$ be the number of blocks selected in Step 3 covering p. Determine the threshold t as the median value of $t(p)$ for $p \in \mathcal{B}$.
6. The potentially tampered ROI is $\mathcal{R} = \{p | \, t(p) > t\}$.

The operator should continue the analysis and verify the identified ROI with the algorithm in Section 3.1.

The idea behind the algorithm is simple. For each block size and shape, some of the blocks with the lowest correlation most probably cover the forged area. The remaining blocks are somewhere else in the image, where a block of this size and shape with low correlation might be located. By removing pixels from the mask $\mathcal{B}$ that are covered by a smaller number of blocks than the rest, we eliminate the regions with low correlation spread somewhere else in the image.

This algorithm requires several parameters, such as the shapes and sizes of blocks. Generally the more shapes and sizes, the better results one can expect, but adding more sizes and shapes increases the computational time.

4.1 Experiment

In Figure 5, we list all $N = 12$ blocks that we used in our implementation of the algorithm. They include two squares 128×128 and 172×172, a rectangle 64×256 rotated 4 different ways, two circles with radii 73 and 109, and ellipses with half-axes 55 and 97 rotated 4 different ways. The parameter m was set to 8.

Figure 6 shows the results of the automatic ROI detection algorithm for four selected forgeries shown in Figure 1. In each case, the algorithm correctly identified majority of the tampered region. For the "person" forgery, there is a small rectangular region also potentially identified as tampered. This is because the region is very dark. The "car" forgery also contains a few regions in the tree branches potentially identified as tampered. After selecting each suspected region as ROI and applying the algorithm of Section 3.1, only the tampered regions were correctly deemed as forged because both the dark region and the tree branches contain some (suppressed) pattern noise that brings the p-values above 10^{-3}.

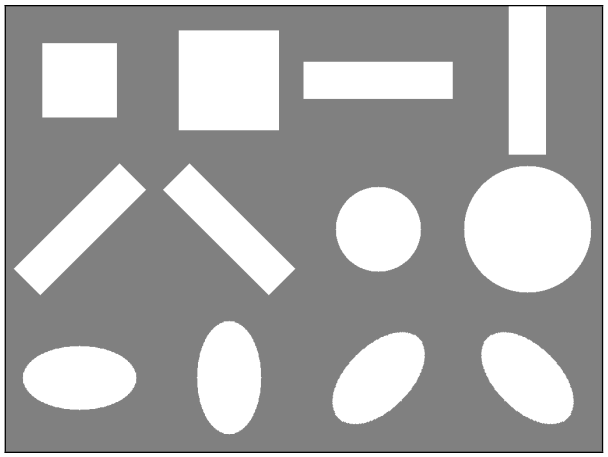
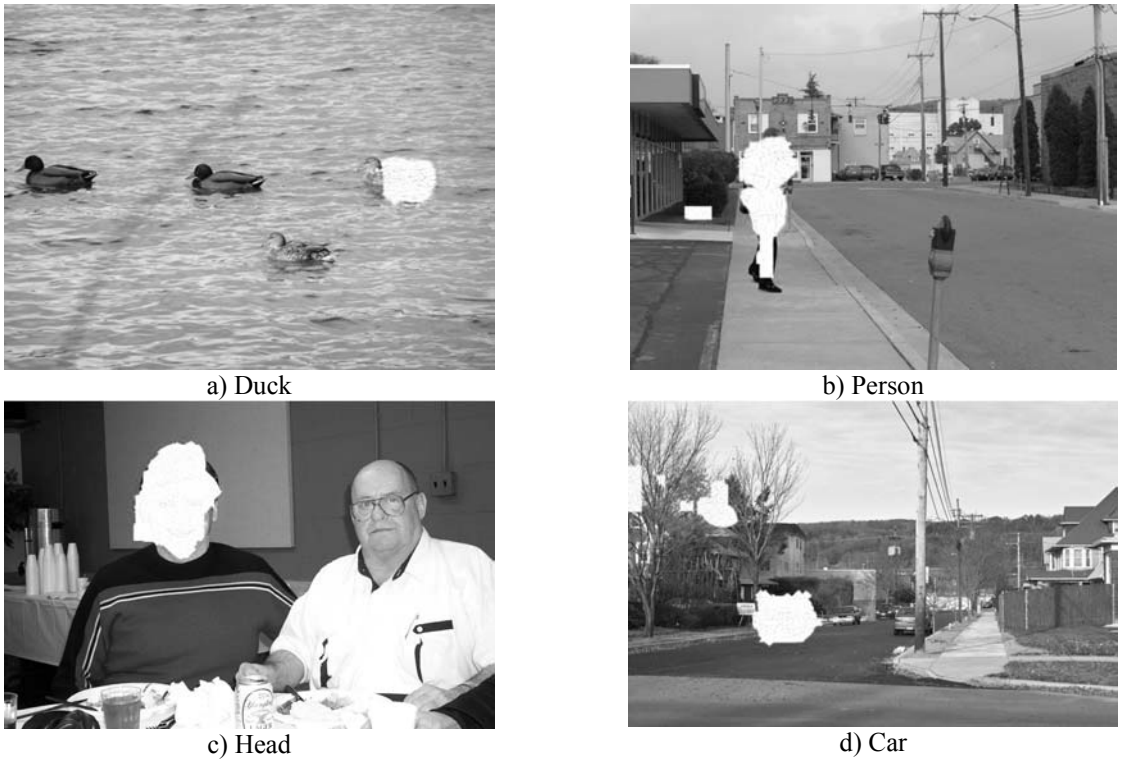


Figure 5: Sliding block shapes used for automatic ROI detection.



a) Duck b) Person

c) Head d) Car

Figure 6: Examples of automatically detected ROIs shown in white.

At this point, we note that when the image contains regions with a very complex texture, the automatic ROI detection algorithm combined with the analysis of Section 3.1 may fail to correctly identify such regions. To estimate how often this may happen, we ran the automatic forgery detection algorithm for 81 images that were not tampered and calculated the *p*-values for the ROIs found by the algorithm. For 17 images, the found ROIs were completely saturated at 0 or 255 and were thus eliminated from further analysis because our method (and, in fact, any other forgery detection method) cannot be applied. Out of the remaining 64 cases, only two regions had their *p*-values above 10^{-3} ($p = 5\times10^{-3}$). These constitute false alarms. Both regions had a very complex texture with partial saturation (see Figure 7).

Figure 7: Two examples of falsely identified regions (heavily textured and partially saturated) region.

5. CONCLUSIONS

We presented a new approach to detection of forgeries in digital images under the assumption that either the camera that took the image or sufficiently many images taken by that camera are available. This forgery detection method is based on detecting the presence of the camera sensor pattern noise, which is a unique stochastic characteristic of imaging sensors, in individual segments in the image. The presence of the pattern noise in each segment is established using correlation as in detection of spread spectrum watermarks. The forged region of interest is determined as the one that lacks the pattern noise and for which there is no natural explanation for the noise being suppressed (saturation, darkness, or complex texture preventing noise extraction). The statistical significance of the correlation in the ROI is evaluated using the *p*-value. Our research indicates that it is possible to perform relatively reliable identification of forged regions even from images that were JPEG compressed with quality factors as low as 70.

We also describe an algorithm for automatic identification of the forged ROI. It is determined as the region with the lowest pattern noise presence in the image. This is achieved by sliding over the image a set of basic shapes with different orientations and accumulating the lowest correlation values. We experimentally investigate the probability of falsely identifying a non-tampered region. More detailed investigation of this method as well as exploration of alternative approaches that might include region-growing algorithms will be the subject of our future work.

We point out that our method may not provide sufficiently conclusive statistical evidence for regions with naturally low presence of pattern noise (mainly saturated areas or very dark areas). Furthermore, geometrical processing, such as cropping or resizing, causes desynchronization with the sensor pattern noise and may require expensive searches that will likely increase the ratio of falsely identified forgeries.

We would like to stress that even though it might seem that the requirement to have available the camera that took the image is not feasible, there are many situations in forensics where the camera is available. For instance, in the court of law, an image of an unknown unspecified origin would be hardly even considered as possible evidence. Finally, we reiterate that reliable forgery detection should be approached from multiple directions, combining the evidence from other methods ([1], [2], [3], [4], [5], and [6]), if applicable.

ACKNOWLEDGEMENT

The work on this paper was supported by Air Force Research Laboratory, Air Force Material Command, USAF, under a research grant number F30602-02-2-0093. 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 Air Force Research Laboratory, or the U.S. Government. Special thanks belong to Taras Holotyak for providing us with Matlab code for the denoising filter. We would also like to thank Paul Blythe for many useful discussions.

REFERENCES

[1] Ng T.-T. and Chang S.-H.: "Blind Detection of Digital Photomontages using Higher Order Statistics", *ADVENT Technical Report* #201-2004-1, Columbia University, June 2004.

[2] 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.

[3] 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.

[4] 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.

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

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

[7] Lukáš J., Fridrich J., and Goljan M.: "Determining Digital Image Origin Using Sensor Imperfections", *Proc. SPIE Electronic Imaging*, *Image and Video Communication and Processing*, San Jose, California, pp. 249–260, January 16–20, 2005.

[8] Lukáš J., Fridrich J., and Goljan M.: "Digital 'Bullet Scratches' for Images", *Proc. ICIP'05*, Genova, Italy, September 2005.

[9] "Understanding How Image Sensors Work", http://www.shortcourses.com/how/sensors/sensors.htm .

[10] Holst, G. C.: *CCD Arrays, Cameras, and Displays*, 2nd edition, JCD Publishing & SPIE Pres, USA, 1998.

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

[12] Lukáš J., Fridrich J., and Goljan M.: "Digital Camera Identification from Sensor Pattern Noise", submitted to *IEEE Transactions on Information Forensics and Security*, 2005.

[13] Janesick, J. R.: "Dueling Detectors", *OE Magazine*, vol. 2(2), February 2002.

[14] 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, Arizona, vol. 6, pp. 3253–3256, March 1999.

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

[16] Meignen, S. and Meignen, H.: "On the Modeling of DCT and Subband Image Data for Compression," IEEE Trans. on Image Processing, vol. 4, pp. 186–193, 1995.