Chapter 9

Media Forensics

Rainer Böhme, Matthias Kirchner

This is the author version of

Böhme, R., and M. Kirchner, "Media Forensics," in Katzenbeisser, S., and F. Petitcolas (eds.), *Information Hiding*, Artech House, pp. 231–259, 2016.

9.1 OBJECTIVES

Media forensics is the youngest subfield associated with information hiding and most closely related to the detection techniques discussed in previous chapters of this book. Like all forensic sciences, media forensics concerns the provision of evidence to support decisions, for example in the court of law. Over the past decade, scholars in media forensics have developed and evaluated a growing set of tools to extract information from media objects pertaining to the authenticity of digital media as valid representations of reality, such as the natural scene depicted in a digital image. Media forensics exploits the fact that potentially compromising editing operations (e. g., tampering) leave traces that render forgeries statistically distinguishable from authentic media objects. Forensically useful traces are often imperceptible, which connects to the theme of information *hiding*. However, unlike for the hiding techniques discussed in previous chapters, forensically useful traces are not actively embedded but emerge as side effects of other processing [1].

While ensuring authenticity is closest to the objectives of watermarking and fingerprinting, many known forensic methods are inspired by steganalysis: media

forensics at its core is a signal detection problem. The forensic analyst tests if an observed signal $\mathbf{X}$ is compatible with the distribution of authentic media objects:

$$\mathrm{H}_0: \quad \mathbf{X} \sim p_a, \tag{9.1}$$

$$\mathrm{H}_1: \quad \mathbf{X} \not\sim p_a. \tag{9.2}$$

As in the case of covers for steganalysis, the distribution of authentic media objects p_a is generally not known, arguably unknowable [2], and often conditional to the context and the prior knowledge of the analyst. Therefore, most analysis methods approximate the optimal statistical test using heuristics and often also human expertise. The role of the human analyst is to select and parametrize analysis methods as well as to cross-check and interpret the results.

Media forensics also shares similarities with other forensic sciences, such as computer forensics [3]. A common feature of both media forensics and computer forensics is the focus of analysis on *digital evidence*, which is data represented in discrete and perfectly observable symbols stored in computer systems. But media forensics and computer forensics assume different generation processes for the digital data. Computer forensics analyses data structures generated by (in principle) deterministic computer programs, such as file system tables in the case of data recovery. By contrast, the distinctive feature of media data is that it originates from the outside of a computer system. A *sensor* maps parts of reality into imperfect and not fully deterministic digital representations. We will see that sensors and their imperfections play a very important role in many techniques of media forensics.

The notion of media data as data acquired by sensors is very general. It comprises audio, image, and video signals as well as more exotic sensory inputs (e. g., location, acceleration). For a number of reasons, researchers have mainly focussed on forensic techniques for still images. Reflecting this state of the art, we focus our discussion in this chapter on digital *image* forensics. Many of the principles can be adapted to other kinds of media data, and we point to important characteristics for the forensics analysis of other media in Section 9.3.

9.1.1 Digital Image Forensics

A simple system model puts the forensic analysis at the end of a processing chain, which consists of at least one acquisition step with optional subsequent processing (see Figure 9.1). The abstract acquisition function takes as input a natural scene and outputs a digital signal. This involves the analog-to-digital conversion of a sensor. Every processing step assumes digital signals as inputs and outputs. The abstract processing step can be instantiated by the identity function (resulting in

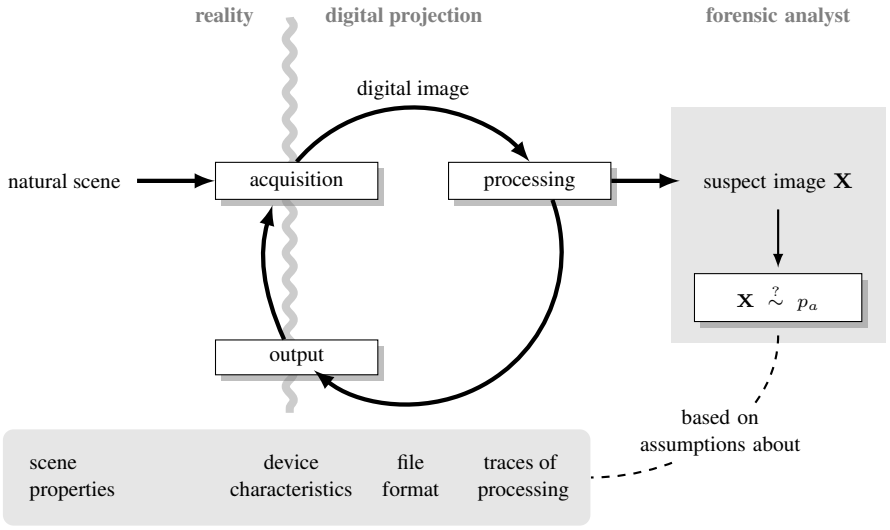

Figure 9.1 Image generation process and forensics.

authentic images) or any combination of operations used to produce a forgery. The intended result of a forensic analysis is a decision about unknown properties of the processing chain, for example whether the processing step was the identity function (H_0) or not (H_1). If ground truth is available, then forensic methods can be benchmarked by measuring decision errors with similar metrics, as introduced for steganalysis. The possibility of repeated transformations between analog and digital representations—for instance by redigitizing a 2 D-print of a digital photograph—is reflected by the loop in Figure 9.1. While the possibility of such complex processing chains must always be on the mindset of forensic practitioners, many methods presented in the literature assume simple processing chains without digital-to-analog conversions. Another difficulty in practice is that the distinction between acquisition and processing is not always as clear-cut as this model suggests. Many acquisition devices do substantial post-processing on the digitized data in order to compensate for mechanical or optical shortcomings, offer image processing operations to their users, or presume some sort of post-processing for instance with specialized software installed on a mobile device (e. g., smartphone) connected to the acquisition device (e. g., body-mount camera).

The notion of *passive* image forensics generally assumes no knowledge of the analyst about the specific instance of the processing chain. However, hypothesis tests for the signal detection problem are not possible without assumptions about typical processing chains. Formally, these assumptions are encoded in the probability distribution of authentic images p_a. As it is hard to deal with the high-dimensional joint distribution p_a in practice, assumptions about different steps of the processing chain are tested independently. Technically, this approach is a projection of the unknown distribution p_a on several low-dimensional subspaces, where for each subspace the distribution of a decision criterium between authentic and forged images can be obtained experimentally. One can organize the assumptions, their corresponding subspaces, and decision criteria along the stages of the system model. As illustrated at the bottom of Figure 9.1, known methods make use of assumptions on scene properties, device characteristics, data structures like the image file format, and traces of processing.

The properties of processing chains of interest to the forensic analyst can be broadly divided into properties indicating the source of an image and properties indicating possible manipulations. For authentic images we expect that

- The signal contains statistical traces of a plausible acquisition function (consistent throughout the entire signal);
- There are no traces of manipulation present in any part of the signal.

In practice, the question of authenticity is often tackled step by step.

9.1.2 Source Identification

Image source identification tries to infer information about the acquisition device from a given signal. Known methods differ in their levels of granularity depending on available information and accessibility of the presumed acquisition device.

The most basic question is to distinguish between natural and computer-generated images [4]. A general approach analyzes the noise characteristics of image pixels, which emerge from imperfections of sensor-based acquisition devices and are generally not modeled or mimicked in rendering software [5]. For specific image contents (e. g., landscapes, faces) approaches based on computer vision have been proposed. They try to identify imperfect modeling of physical or physiological properties in typical rendering software [6].

The next question to ask for natural images is to identify the class of acquisition devices, for example whether an image was digitized with a digital camera or

a flatbed scanner. Methods dealing with this question leverage knowledge of fundamental engineering differences between types of sensors. Images acquired with line sensors (e. g., in flatbed scanners) exhibit distinct noise characteristics compared to images acquired with sensor arrays (e. g., in digital still and video cameras). Moreover, many flatbed scanners do not interpolate color information from a filter array. The absence of the characteristic interpolation artifacts of color filter arrays (CFA, see Section 9.2.2.4) can identify scanned images [7, 8].

Each class of acquisition devices can be further divided into different models[1] (or makes). Images acquired with devices of the same model share characteristics introduced by the combination of hardware or software components in this model's processing chain. Emphasis is on the combination because there is no single characteristic known that systematically varies between models and is similar for all devices of a model. Typical methods measure a broad range of features spanning optical aberrations, noise metrics, digital signal processing artifacts, and parameter choices of the primary image compression. They feed the resulting feature vector to machine learning algorithms for classification [9]. To generate the labeled data necessary for supervised learning, example images from at least one (preferably more) devices of each model are needed. As the number of models grows constantly, maintaining a comprehensive training database becomes quite challenging [10].

If the actual acquisition device (possibly among others) or sufficiently many test images from that device are available to the forensic analyst, a digital image can even be linked to its acquisition device with high certainty by leveraging traces of inevitable manufacturing imperfections and wear and tear of the sensor (e. g., defective pixels) in the resulting image signal. (See Section 9.2.2.3 for details.)

9.1.3 Manipulation Detection

Manipulation detection tries to detect and possibly specify content-changing post-processing after the acquisition of a digital image. It broadly takes two approaches. First, if sufficient information about the acquisition device is known (from context or preparatory source identification forensics), then the device-specific characteristics can be checked for consistency. Global or local deviations from the reference values can be interpreted as indications of post-processing. For example, an image acquired with a specific digital camera may exhibit linear dependencies between pixels of different color channels resulting from the CFA interpolation in the acquisition device. If the dependence structure is missing or differs in parts of the image,

1 For consistency with the terminology in the literature, we overload the term *model*. It refers to a device type in this paragraph and to a set of simplifying assumptions in the rest of the chapter.

it is very likely that this region has been edited locally [11]. Other features of this kind include parameters of measurable aberrations relative to the optical center of the image, inconsistent sensor noise, or linear independence at block boundaries of the primary JPEG compression.

Second, many content-changing processing operations add statistical traces to the signal independent of how it was acquired. For example, geometric transformations, often used to adjust size and orientation of pasted objects to the environment, leave traces that are characteristic for the resampling method and the parameters of the transformation [12, 13]. Other traces of processing include duplicated image regions resulting from attempts to cover up manipulations with local operations like the copy stencil, or artifacts introduced after repeated quantization if intermediate states of a processing chain are stored in a lossy compressed image format.

9.2 METHODS

Traces useful for forensics, whether generated by the acquisition device or processing operations, appear on different layers of analysis. We broadly distinguish between scene level, signal level, and data structure level.

9.2.1 Layers of Analysis

The *data structure level* refers to the syntactical encoding of the data stream, which is defined—albeit loosely—by the file format or communication protocol specification. Forensic evidence emerges from different implementations of the specification as most complex standards include many optional elements and do not support a single canonical form. The multitude of metadata options available in image file formats (e. g., EXIF and custom application headers in JPEG files) has turned out to be a most valuable resource for image forensics on this layer [14]. Also the order of elements in a tagged data structure as well as the parameters of lossless encoding add variability that helps to identify at least the last encoder of the processing chain [15]. While critics argue that forensic analyses based on data structures alone are unreliable because metadata can be changed with relatively little effort, it still requires substantial knowledge, skill, and patience to do this plausibly and consistently. Another strong argument for analyses on more than one layer is that data structures may indicate some processing but do not reveal much about what operation has been applied. It may make a difference whether an image has been recompressed by a social media platform (unavoidable in many cases) or locally edited and then recompressed.

The borderline between *scene level* and signal level is less clear. As a rule of thumb, we speak of scene level if the forensic method tries to analyze macroscopic properties of the image to support a decision based on the extracted semantics. The last qualifier is important because, for instance, a global histogram is also a macroscopic property, but it conveys little information about the scene. Therefore, scene level analysis is more related to computer vision than to signal detection. Most known methods require more human intervention and are more sensitive to the human part than methods operating on the signal level. For example, [16] proposes a computer-aided method to estimate the direction of diffuse light at selected points in a suspect image. This may help to substantiate claims about inconsistent lighting of multiple objects in the same scene, which may indicate a composition from multiple source images. In a similar vein, [17] study light directions at specular reflections and [18] support the analysis of complex shading and shadows. All mentioned approaches fit geometric models of the physical world depicted in a scene to the digital representation.

The *signal level* is by far the best researched and so far the most promising approach to image forensics [19]. It combines many desirable properties such as the independence of the scene content (disregarding pathologic cases of singular scene content), high accuracy of automated decisions for simple processing chains, and sufficient information to carry out in-depth manual investigations of complex processing chains. The signal level carries characteristics of the acquisition device. In addition, characteristic traces of typical processing operations are measurable in the signal level as well. Taken together, all these characteristics offer a wealth of information to the forensic analyst. We will review the most important principles and methods to extract and interpret signal level information in the following sections.

9.2.2 Device Characteristics

Device characteristics refer to image characteristics that can be attributed to the acquisition device. They serve for source identification and as a kind of inherent watermark to track further processing, which may partly erase or transform the device characteristics, thereby unveiling the processing operation and its parameters. Considering the widespread use of digital cameras, our review will follow the literature and will emphasize digital camera characteristics. We refer to [20] for a comprehensive review of scanner characteristics.

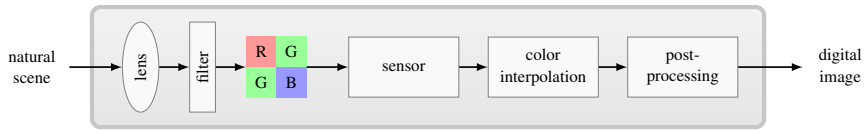

Figure 9.2 Digital image acquisition with a digital camera: stylized pipeline.

9.2.2.1 Digital Camera Pipeline

Figure 9.2 shows a stylized processing pipeline of a digital camera with its most relevant components. Incoming light of the scene is focused on the sensor by a complex system of lenses. An optical filter to reduce undesired light components (e. g., infrared light) sits between these components. Typical camera sensors capture image pixels by individual CCD or CMOS sensor elements, which output an electric charge proportional to the light received at the corresponding position on the two-dimensional sensor plane. These sensor elements are color-blind; they can only measure light intensity. Color information is obtained by arranging the sensor plane in the form of a color filter array (CFA) where each sensor element is sensitive to light of a certain wavelength only—red (R), green (G), and blue (B) in most cases. Missing color information can then be estimated from surrounding pixels of the raw intensity map. This process is also known as CFA interpolation or demosaicing. After CFA interpolation, the image is subject to a number of camera-internal post-processing steps, including for instance color correction, edge enhancement, and finally compression.

9.2.2.2 Lens Distortions

Modern digital cameras are equipped with a complex optical system that projects a scene to a sensor of much smaller dimension. This projection is in general not perfect. As a result, a plethora of lens distortions (also known as aberrations) are present in digital camera images. Forensic source identification assumes that shape and strength of lens distortions depend on the lens(es) in use. Tractable models of aberrations are typically parametrized by the radial distance to the optical center of the image. For the purpose of manipulation detection, these models can be fitted globally and then tested for local consistency throughout an image.

Prevalent types of distortion are lens radial distortion, vignetting, and chromatic aberrations. Lens radial distortion is a nonlinear geometric distortion that lets straight lines appear curved. This effect is generally more pronounced toward image

Figure 9.3 Lateral chromatic aberration in a digital camera image. Red color fringes along edges are marked by arrows in the magnified (and contrast-enhanced) details on the right. The occurrence and strength of color fringes generally varies with the position of edges in the image. This image was acquired with a Nikon 18–200 mm zoom lens at a focal length of 150 mm.

corners, typically modeled by a polynomial of small degree [21]. Some cameras try to correct for it during post-processing, which may leave characteristic traces by itself [22]. Vignetting refers to the radial decrease of light intensity toward the corners of an image. It is best visible and measurable in homogenous images captured with wide apertures [23], for which only a fraction of the light reaches the outer regions of the sensor plane.

Chromatic aberrations are perhaps most relevant for forensics. They describe the effect that polychromatic light is spread over different positions on the sensor plane because the lens' dispersion index varies with the wavelength. Lateral chromatic aberrations often produce visible color fringes along edges. They are particularly well measured by the spatial displacement (i. e., contraction or expansion) of different color channels relative to each other [24]. Let the green channel, G, be the reference. Then the coordinates of the displaced red or blue channel $D \in \{R, B\}$ with optical center (i'_D, j'_D) are modeled as

$$\begin{pmatrix} i_D \\ j_D \end{pmatrix} = \begin{pmatrix} \alpha_D \cdot (i_G - i'_D) + i'_D \\ \alpha_D \cdot (j_G - j'_D) + j'_D \end{pmatrix}. \tag{9.3}$$

The tuple of model parameters, (α_D, i'_D, j'_D), can be estimated efficiently from a single image [25]. Figure 9.3 illustrates how the orientation of these displacements varies across the image. The red channel expands relatively to the other channels in this example (i. e., $\alpha_R > 1$). This can be exploited for manipulation detection,

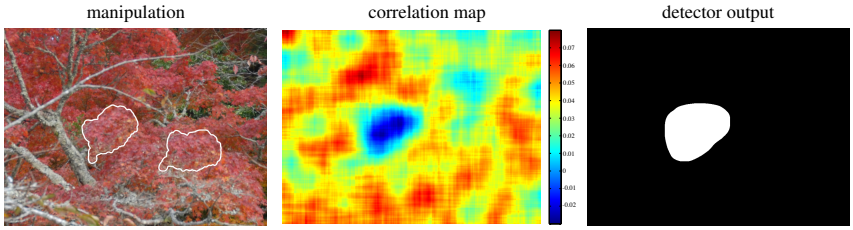

Figure 9.4 Manipulation detection based on sensor noise [26]. Image manipulation (left): part of the image was copied, rotated by 30° and then reinserted at a different position. The two regions are marked in the image. Correlation map (center): each intensity value in the map corresponds to the correlation score of a 128×128 pixel block from the image's noise residual with the corresponding fingerprint estimate. The manipulated region lacks the expected sensor noise pattern, yielding low correlation between local sensor noise estimates and the fingerprint of the image's camera (indicated by darker colors in the map). Detector output (right): post-processing and binarization of the correlation map gives a clear indication of the manipulated region.

for instance if one region of an image is copied to another region with a different expected aberration profile [24].

9.2.2.3 Sensor Imperfections

Sensor imperfections created by inevitable variations in the manufacturing process of sensor elements and sensor wear-out are valuable device characteristics. Sensor outputs are generally noisy: the intensity values fluctuate slightly even if the sensor plane is lit absolutely homogeneously. Sensor noise is composed of temporal and spatial noise. Temporal noise subsumes noise components that are stochastically independent between different images acquired with the same sensor. Shot noise and read-out noise are typical sources of temporal noise. By contrast, spatial noise is relatively stable over time and only varies between individual sensor elements. This makes spatial noise particularly interesting to forensic analysts. It can serve as a camera "fingerprint", and can also be tested for consistent appearance in different regions of an image [27, 28]. This quality is commonly attributed to photo-response nonuniformity (PRNU), a noise source that adds a camera-specific unique multiplicative pattern to the signal. It is caused by inevitable material imperfections and variations in the manufacturing process of individual sensor elements. A sensor fingerprint $\mathbf{K}$ can be estimated by some form of pixel-wise averaging of noise

residuals over a number of images $\mathbf{X}^{(n)}$ taken with the same camera,

$$\hat{\mathbf{K}} = \sum_{n=1}^{N} \mathbf{H}^{(n)} \mathbf{W}^{(n)} = \sum_{n=1}^{N} \mathbf{H}^{(n)} \left(\mathbf{X}^{(n)} - F(\mathbf{X}^{(n)}) \right) , \tag{9.4}$$

with $\mathbf{H}^{(n)} = 1/N$ for simple averaging [27], or $\mathbf{H}^{(n)} = \mathbf{X}^{(n)} / \sum_{n=1}^{N} \left(\mathbf{X}^{(n)} \right)^2$ for a maximum likelihood estimator of multiplicative noise [28]. The noise residuals $\mathbf{W}$ are obtained by processing images with a denoising filter $F(\cdot)$. Cameras can be identified by extracting the noise residual from the image under investigation and measuring its similarity to an estimated camera fingerprint. Suitable similarity metrics include Pearson correlation [27], normalized cross-correlation [28], and peak-to-correlation energy (PCE) [29].

For manipulation detection, the similarity between an image's noise residual and the camera fingerprint estimate is evaluated for small (possibly overlapping) blocks. If the processing operations of interest corrupt the local sensor noise pattern as a side effect, then low local similarity scores indicate a forgery. The two leftmost panels of Figure 9.4 illustrate this effect. Modern detectors apply more sophisticated criteria. For instance, the right panel of Figure 9.4 shows the outcome of a state-of-the-art detector that employs a Bayesian Markov random field model to account for local dependencies between blocks in close proximity [26].

Photo-response nonuniformity has also been applied to examine scanned images [30, 31]. Typical line sensors of flatbed scanners repeat spatial noise characteristics along rows. This directional characteristic of the noise pattern allows forensic investigators to distinguish between digital camera images and scanned images [32].

Sensor noise estimates may also contain traces of sensor defects (i. e., sensor elements that constantly output too high or too low intensity values). The occurrence of these defects is characteristic for individual cameras [33] and accumulates over time, enabling temporal forensics [34]. A similar effect is caused by dust particles on the sensor protective glass [35]. Yet many cameras try to correct sensor defects and sensor dust particles with post-processing. In general, their appearance also depends strongly on the image content and on lens settings. All these factors limit the usefulness of sensor defects compared to sensor noise.

9.2.2.4 Color Filter Array Characteristics

To acquire color images with sensors that are physically limited to measure light intensity only, the incoming light has to be split up in several components. Most digital cameras do this by combining a single sensor with an array of color filters

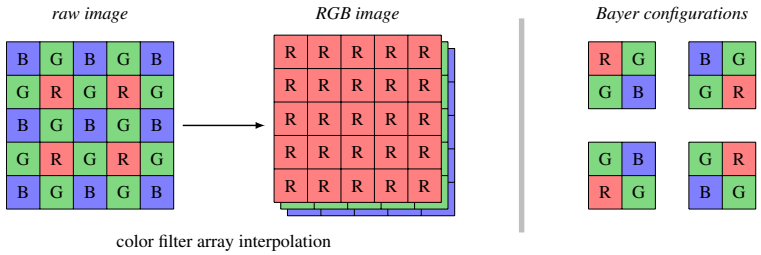

Figure 9.5 Typical digital cameras employ a color filter array (usually one of the four Bayer configurations shown on the right). Each sensor element is sensitive to light of a certain wavelength (here red, green, and blue) only. A color image is produced by interpolating the remaining color information from surrounding pixels of the raw image.

so that different sensor elements capture different color information. The missing information is then interpolated, a procedure that is also known as *demosaicing* (because color filters are arranged like mosaics; see Figure 9.5).

A CFA *configuration* describes how color filters are arranged. As different camera models use different CFA configurations, this parameter is a valuable device characteristic for forensics [36, 37, 38]. Although, in principle, a wide variety of CFA configurations is conceivable, the dominant CFA layout repeats a 2×2 Bayer pattern over the entire sensor pane. Bayer patterns exist in four configurations and are characterized by two green elements arranged diagonally with one red and one blue element filling up the remaining space (see Figure 9.5).

Demosaicing a Bayer pattern implies that at most one-third of all pixels in an RGB image contain genuine information from a sensor element. The remaining pixels are interpolated from the local neighborhood of the raw signal. As a side effect, pixels become locally dependent even stronger and more systematically than local correlations in the original signal. The repetition of a fixed pattern over the entire image causes periodic dependency structures in the image [11]. (See Section 9.2.3.3 below for a method to identify periodic dependencies.) The specific form depends not only on the CFA configuration, but also on the demosaicing algorithm. This observation has motivated CFA-based camera model identification approaches [36, 39]. Similar dependencies occur between the color channels of an image. Some forms of post-processing destroy these demosaicing traces. The resulting local inconsistencies have successfully been exploited to localize tampering [40]. Finally, the absence of any CFA traces is an indication that a given image was not acquired with a digital camera [7, 8].

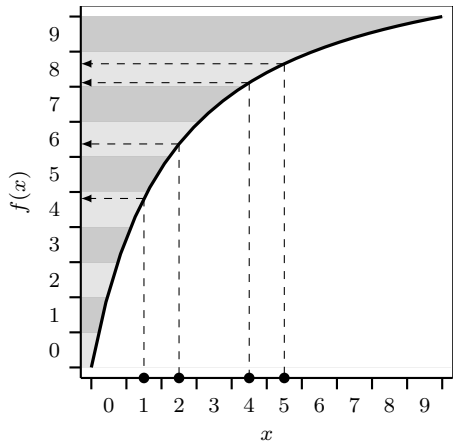

Figure 9.6 Requantization after applying continuous functions on discrete data. Depending on the curvature of the mapping (here: gamma correction) and the quantization step size (here: 1), some output values appear twice as frequently and others not at all. (This holds strictly for uniformly distributed input data and approximately for smooth marginal distributions as in typical media signals.)

9.2.3 Processing Traces

Recall from Section 9.1.1 that processing traces emerge as side effect of the image processing operations used to produce a perceptually convincing forgery (see also Figure 9.1). The presence of processing traces in a suspect image indicates manipulation and their exact realization may reveal information about the parameters of processing operations. Processing traces may permeate an entire image or parts of it. In the latter case, the distribution of traces within an image helps to localize tampering.

9.2.3.1 Requantization

Arguably the most important source of processing traces is requantization: already quantized discrete numbers are used as inputs of functions defined for continuous domains and codomains. The return values are quantized *again* in order to be mapped to the discrete alphabet of typical signal representations.

An introductory example for requantization is a detector of gamma correction. Gamma correction refers to the point operation defined by the continuous function

$$y_{ij} = (x_{ij})^{\gamma}, \tag{9.5}$$

where parameters $\gamma < 1$ decreases the contrast of a grayscale image $\mathbf{X}$ and $\gamma > 1$ increases the contrast. If $\mathbf{X}$ and $\mathbf{Y}$ are integer arrays interpreted as ℓ-bit fixed-point representations of numbers in the normalized intensity range $[0, 1]$, then the assignment in Equation (9.5) is implemented as discrete function $f : \{0, \ldots, 2^{\ell} - 1\} \to \{0, \ldots, 2^{\ell} - 1\}$. Specifically,

$$y_{ij} = f(x_{ij}) = \left[(2^{\ell} - 1) \left(\frac{x_{ij}}{2^{\ell} - 1} \right)^{\gamma} \right], \tag{9.6}$$

where square brackets denote rounding to the nearest integer.

Figure 9.6 shows this mapping for $x \in \{0, \ldots, 9\}$. Observe that for the chosen parameters, f can never take the value 5 because no discrete input maps to it. Likewise, two values in the domain of f, 4 and 5, map to the same value 8. If the input signal's histogram is broadly smooth, this coincidence will add up to a peak in the output histogram. This is exactly what we can observe in gamma-corrected grayscale images as illustrated in Figure 9.7. The contrast-reduced ($\gamma = 0.6$) lower half of the image exhibits gaps in the left tail of the histogram and peaks on the right side. For comparison, the histogram of the unprocessed upper half is locally smooth and does not contain such artifacts.

Gamma correction is a relevant operation in typical image processing chains, in particular for producing visually plausible compositions from parts of images taken under different lighting conditions or exposures. A simple way to automatically detect the resulting processing traces is to analyze the histogram of suspect image in the frequency domain. Gaps and peaks produce strong high-frequency components in the spectrum that do not appear in natural images. If the high-frequency components, after some necessary windowing close to the boundary values 0 and $2^{\ell} - 1$, exceed a certain threshold, a suspect image (or image region) is flagged as processed with gamma correction [41].

This simple method works best for images in spatial domain representations. Transformations to the frequency domain, like the DCT used in the popular lossy JPEG compression, tend to smooth the histograms of intensity values after back-transformation to the spatial domain. This attenuates peaks and gaps in the histogram and makes gamma correction more difficult to detect.

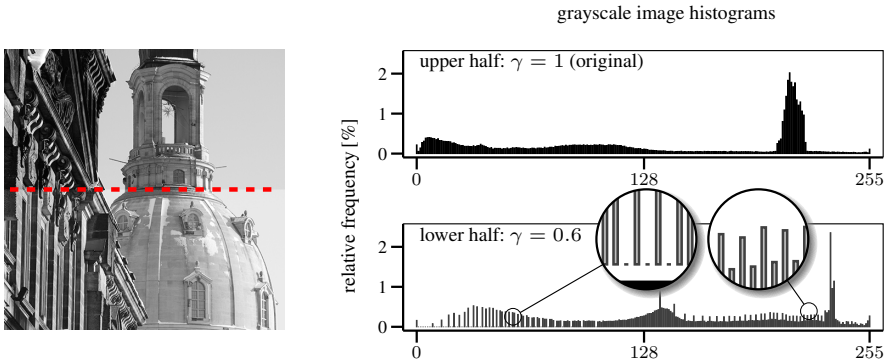

Figure 9.7 Processing traces realized as gaps and peaks in the histogram after gamma correction.

But lossy compression itself involves several requantization steps, which leave forensically useful processing traces. For example, requantization happens at many stages in a JPEG compression–decompression cycle:

- After color transformation from RGB to YCbCr;
- After chrominance channel subsampling;
- During the (fast) 2D-DCT transformation;
- During and after the (fast) inverse 2D-DCT transformation;
- After chrominance channel upsampling (for certain implementations);
- After color back-transformation to RGB;

and, most importantly,

- When DCT coefficients are explicitly quantized with frequency-dependent step sizes taken from the quality-dependent JPEG quantization matrix.

Every stage in the enumeration above can be written as a continuous function f with subsequent rounding (although some implementations reuse intermediate results for efficiency, complicating the analysis). A save-and-open sequence of image editing software goes through all these stages, thereby leaving traces of requantization in the form of perceptible or imperceptible compression artifacts.

9.2.3.2 Lossy Compression

The complexity of the popular JPEG compression and the interaction of many factors precludes a formal or even comprehensive treatment in the context of this chapter. What matters is that virtually all traces useful for the forensic exploration of the (potential) JPEG compression history [42] emerge from requantization in one form or another. In this sense, most forensic methods analyzing compression artifacts can be seen as special cases of detectors of requantization. The approaches proposed in the literature differ in how processing traces are extracted, in the supported image formats, and the assumed knowledge of (candidate) quantization matrices and of the specific implementation of the JPEG standard. For instance, methods exists to check spatial domain images for prior JPEG compression and its parameters [43] or to identify local inconsistencies in JPEG errors indicating compositions [44]. Images in JPEG format can be analyzed for double or multiple compressions [45]. The literature is so specialized that it contains already tailored methods to evaluate one form of requantization traces (e. g., JPEG history detection) in the presence of distortion by other forms of requantization (e. g., contrast enhancement) [46].

Technically, many methods rely on (recomputed) JPEG DCT coefficients and apply adapted versions of Benford's law on the distribution of numerical digits to test for singularities resulting from requantization [47]. This approach can draw on solid theory [48] and is surprisingly effective given that only first-order statistics are evaluated [49]. (Part of the reason is that good statistical models are known for histograms of DCT coefficients, unlike for spatial domain intensity histograms.) However, the approach loses precision if multiple compressions use exactly the same parameter and it reaches its limits if the quantization step sizes are very small (i. e., for images compressed with JPEG quality close to 100%).

Under these conditions, another approach is more reliable. It leverages the fact that JPEG compression cuts the image into nonoverlapping blocks of 8×8 pixels and then applies the compression–decompression chain on each block independently. Repeated requantization introduces complicated dependencies between pixels within a block. These effects are not fully modeled yet and do not seem to be measurable with first-order statistics, like histograms. However, a key observation is that blocks converge to a stable state after a small but seemingly random number of iterations. In this context, a block is called *stable* if all its pixel intensity values in the spatial domain representation take exactly the same values after a full compression–decompression cycle.

Empirical evidence from natural images as well as from synthetic data suggests that the distribution of the time until convergence is fairly independent of the

Figure 9.8 Convergence of JPEG blocks after repeated requantization. White marks indicate 8×8 blocks that remain stable between the first and the second JPEG compression (left) and the second and third JPEG compression (center and right). The right image has been locally manipulated after the first JPEG compression and the result was stored as JPEG. All experiments use `libjpeg` default settings for compression and decompression and quality factor 100 %. The test image size is 512×512 pixels.

image content (disregarding flat blocks, which are always stable after one iteration). By tabulating the steps until convergence for all blocks of an image, the estimated distribution can be matched against ground truth distributions obtained under controlled conditions. This enables a detector for prior JPEG compression up to the highest possible quality of 100% where all scale factors in the quantization matrix are set to one. This detector can also reveal the number of recompressions, which may indicate the depth of the image editing chain [50].

Block convergence can also help to localize tampered regions as illustrated in Figure 9.8. The left image has been compressed with JPEG quality factor 100% once. The white marks superimposed in the figure indicate blocks that remain stable after another compression–decompression cycle with the same parameters. Observe that the number of stable blocks increases substantially if the whole JPEG image is opened and resaved (i. e., recompressed) unaltered with image-editing software (center). Hence, the ratio of stable blocks indicates the compression history. The right image shows the distribution of stable blocks if local editing happened between opening and resaving the image: blocks in the altered region appear like never-compressed content and converge slower on average than the surrounding blocks.

In summary, block convergence analysis complements the analysis of DCT coefficient histograms in situations where the JPEG quality is high and the parameters of multiple compressions remain constant. However, the lack of solid theory and the reliance on more subtle higher-order dependencies observable only by counting steps until convergence makes block convergence more sensitive to the implementation of the JPEG standard, in particular the DCT and inverse DCT algorithms [51].

There also exist forensic methods evaluating artifacts of compression algorithms other than JPEG [52], but these methods are omitted here for their lower practical relevance in still image forensics.

9.2.3.3 Resampling

Realistic manipulations often involve resizing or rotating images or parts thereof. Technically, such geometric transformations can be described as *resampling* of the original image grid. A rather naive approach would be to transform the discrete source coordinates of every pixel with a continuous function and round the resulting coordinates to destination coordinates. This would result in severe visual distortion because the mapping between source and destination pixels is not always bijective. The effect is comparable to the source of requantization artifacts described in Section 9.2.3.1 with the only difference that it affects spatial coordinates rather than intensity values.

Researchers in image processing have recognized this problem for long and use interpolation to produce smooth and visually appealing transformations. Very similar to color filter array interpolation (cf. Section 9.2.2.4), resampling introduces linear dependencies between adjacent pixels. These dependencies vary periodically throughout the image and can be understood as traces of resampling [12, 53]. Figure 9.9 illustrates the formation of periodic linear dependencies for the particularly indicative case of upscaling by factor two using bilinear interpolation. Arbitrary geometric transformations produce similar artifacts (except strong downscaling and rotations by multiples of $90°$). The periodicity and amplitude generally depend on the transformation parameters as well as on the interpolation method [13].

The standard output of resampling detectors is a so-called p-map [12], where "p" stands for the probability that a pixel has been interpolated. P-maps have the same size as the image under investigation and can be computed from a linear predictor of pixel intensities [13],

$$r_{ij} = x_{ij} - \mathrm{Pred}(\mathcal{N}_{ij}) \,, \tag{9.7}$$

as known from steganalysis. Function $\mathrm{Pred}(\cdot)$ is a linear predictor that estimates the intensity of pixel x_{ij} from its local neighborhood $\mathcal{N}_{ij}$ [13]. The predictor residuals contain relevant information because interpolated pixels are more correlated with their neighbors pixels and have a better fit with the linear model. Therefore they produce comparably lower absolute prediction residuals than pixels with more genuine signal information. A simple variant of the p-map evaluates for each pixel the

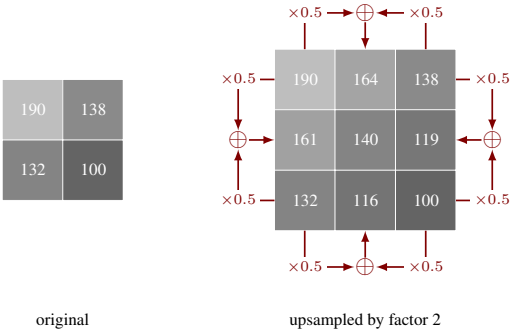

Figure 9.9 Bilinear resampling of a 2×2 pixel block (left) by a factor of two. Every other pixel in the resized block is a linear combination of its direct neighbors (right). Geometrically transformed images are composed of a large number of such blocks (i. e., periodic linear correlations occur).

likelihood p_{ij} that its predictor residue r_{ij} obeys a suitable global distribution assumption. An i. i .d. zero-mean Gaussian model has been found to work sufficiently well in practice,

$$p_{ij} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{r_{ij}^2}{2\sigma^2}\right) . \tag{9.8}$$

The empirical variance σ^2 can be estimated from the residual image. Periodic artifacts in a resampled (part of an) image are particularly well detected after transforming its p-map to the frequency domain, where distinct peaks become visible in the magnitude spectrum. The center panel of Figure 9.10 shows a typical example.

Linear predictor residuals and p-maps computed from them are not only sensitive to resampling artifacts, but naturally capture a much wider range of image characteristics. CFA interpolation, for example, is known to produce high-frequent periodic artifacts similar to upscaling by a factor of two [11]. As well, JPEG compression leaves traces in the p-map. The right panel of Figure 9.10 indicates that increased prediction errors along JPEG block boundaries result in periodic artifacts with a frequency of $1/8$. Forensic analysts need to carefully differentiate between these types of artifacts in practice.

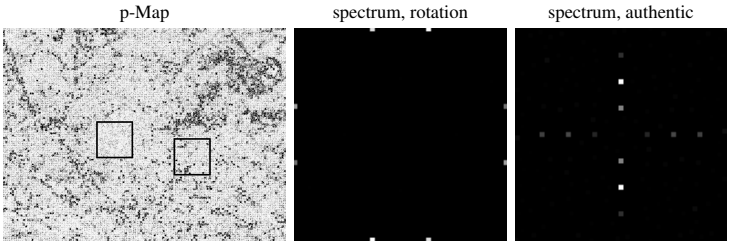

Figure 9.10 Resampling detection for the manipulation depicted in Figure 9.4. Plain p-map (left): brighter shades indicate that pixels are more correlated with their spatial neighbors. JPEG blocking artifacts are visible in authentic image regions. The rotated image region (marked by the left square) exhibits distinct characteristics. Fourier spectra from rotated and authentic regions of the p-map (right panels): rotation yields strong high-frequency peaks, visible at the borders of the left spectrum; JPEG blocking artifacts cause characteristic peaks (at multiples of $1/8$) in the spectrum of the authentic region.

9.2.3.4 Duplicate and Near-Duplicate Regions

Copy–move forgeries are another common class of image manipulations where a region of an image is copied, possibly filtered, and then reinserted at a different position in the *same* image. The copied region will typically undergo some form of post-processing for a more realistic alignment with its surroundings. Copy–move forgeries contain near-duplicate image regions, which can be localized with a suitable matching procedure. A straightforward approach considers (possibly overlapping) blocks of small size and compares local image contents block-wise. A manipulation is declared if a sufficiently large number of near-duplicate blocks share the same spatial relation [54]. To achieve robustness against operations beyond simple copying (e. g., geometric transformations of the reinserted region, image filtering, or lossy compression), image blocks are transformed to a suitable feature space prior to running the matching procedure. The rotation-invariant Zernike moments are among the most promising feature representations for this purpose [55, 56]. Post-processing the correspondence map from the matching procedure helps to remove isolated false positive matches.

A major challenge of block-based matching is computational complexity: an exhaustive search over all block pairs is prohibitive for megapixel-sized images. More efficient approaches restrict the search space by preordering blocks, by the use of structured data representations such as kD-trees, or by means of randomized nearest-neighbor search algorithms like PatchMatch [57]. Figure 9.11 presents a

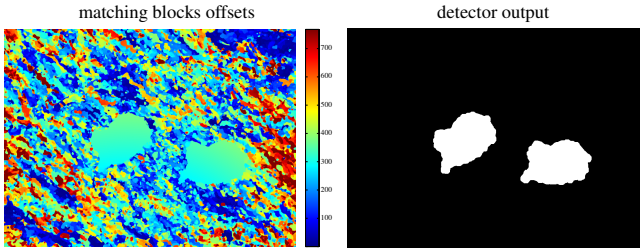

Figure 9.11 Copy–move forgery detection for the manipulation depicted in Figure 9.4. Matching blocks offsets (left): PatchMatch is used to find near-duplicate blocks (in terms of their Zernike moments) in the image [58]. The map visualizes the magnitudes of spatial offsets between matching block pairs. Detector output (right): post-processing and binarization of the offset map gives a clear indication of the duplicated regions.

typical result of a state-of-the-art copy–move detector that combines Zernike moments computed from overlapping blocks of size 16×16 pixels with a PatchMatch-based matching procedure [58]. In terms of computational efficiency, the algorithm is only outperformed by approaches that trade off the high localization accuracy of spatially dense block matching against a search over the much more sparsely populated set of key point descriptors (for instance based on the popular scale-invariant feature transform [59]). We refer to Christlein et al. [55] for a comprehensive benchmark of a variety of different feature representations and matching procedures.

A more general variant of copy–move forgery detectors relax the assumption that the copied region originates from the spurious image under investigation. The literature often refers to this type of forgery, where the image manipulation can be a composition of arbitrary image parts, as *splicing*. Splicing detection is generally a much more difficult problem. In practice, any forensic technique designed to uncover inconsistencies across different parts of image can be used as splicing detector, but we are not aware of a detector that reliably separates all inconsistencies caused by splicing from differences between spatial segments in authentic images.

9.3 LIMITATIONS AND OUTLOOK

Over the past decade, media forensics has developed as a serious research field combining security with signal processing. The resulting toolbox of specialized methods is still growing at an increasing rate. However, the available methods differ in their maturity. As in many fields, there remains a gap between laboratory results

and real-world performance. This gap is quite narrow for a few methods where the forensically useful characteristics have been shown to be robust, and where reliable benchmark datasets exist to validate the effectivity of known approaches [60, 61]. The gap is larger for methods that require specific conditions and many assumptions on the exact implementation of the processing chain.

For the case of digital still images, sensor imperfections, specifically PRNU (see Section 9.2.2.3), and metadata are the most reliable source of information for nowadays forensic analyses. Device identification using PRNU has been confirmed in many realistic settings and shown to be robust against various sources of distortion. Among the processing traces, requantization is pretty robust and most indicative in the special case of lossy JPEG compression. However, there are many reasons why images are recompressed. Hence, the compression history does not always answer all questions of the forensic analyst. Traces of resampling interfere with compression, which limits their applicability. Duplicate detection is computationally expensive in large images, sensitive to parameter settings, and still prone to false positives in many natural images. Scene level analyses still suffer from the subjectivity of the human operator and are by principle limited to specific scene contents. Nevertheless, while each method has its specific limitations (hopefully known to the decision maker), the combination of available methods puts forensic analysts in a much better position than imaginable 10 years ago.

A general limitation of forensic detectors is that most of them are designed with a signal processing mindset rather than a security mindset: few methods consider an intelligent adversary who tries to defeat or mislead forensic analyses by erasing traces or deliberately inserting false traces [62]. Technical methods that help the counterfeiter in defeating forensic analysts are typically referred to as *counter forensics* (or *anti forensics*). A simple yet effective approach is to reduce the available information for forensic analyses by downgrading the image quality after performing a manipulation in high resolution. Since most forensic methods are statistical in nature and thus rely on the law of large numbers and increase in precision with the number and precision of observations (samples), this method is effective in situations where low-quality media are plausible (e. g., on the web). More serious advances in counter forensics preserve the image quality and are actively researched in order to assess the limitations of known forensic methods [63, 64], to develop more robust methods [65, 66], and to erase indicative traces in legitimate cases [67], such as to protect the privacy of people sharing media data online. For an overview on digital image counter forensics we refer the reader to our book chapter [68].

As alluded above, forensic detectors can identify processing, but they cannot (and should not) conclude about the authenticity of a media signal in a nontechnical context. So-called *legitimate processing* is common practice: many digital images are resized or recompressed for transmission and almost all art directors adjust exposure and color before reproduction. Therefore, not every form of processing that technically qualifies as editing indicates an attempt of deception or other malicious intentions. There exist approaches to quantify the amount of processing in a metric that tries to respect editing conventions with customizable weights [69]. But the inherent subjectivity and unavoidable measurement errors from failing to recognize the context limits the applicability of this approach to very narrow domains. Our outlook here is reserved because even perfect technical detectors cannot answer the socially relevant questions of authenticity or legitimacy: the very same processing operation can be legitimate in one context and deceptive in another.

While the technical exposition in this chapter focused on digital still images, the most researched subfield of media forensics, many of the approaches presented generalize to other media as well (e. g., analyses on the data structure level, traces of requantization, lossy compression, etc., on the signal level [70]). Yet the detectors need to be adapted to the specific target signal and format. For example, the common practice of motion vector estimation in video compression displaces patches of fixed pattern noise and creates traces similar to copy–move forgeries if predictive-coded frames are analyzed like still images [71]. Temporal editing (e. g., removing frames) of compressed video leaves characteristic traces in the recompressed groups-of-pictures [72]. For audio recordings, the *electrical network frequency* (ENF) criterion deserves special attention [73]. It is a different type of trace that exists in the time domain of many audio recordings and has recently been explored for video as well [74, 75]. The ENF gets interspersed from the electrical network surrounding the recording device at the time of recording. As the frequency of this signal varies slightly around its norm (e. g., 50 Hz in Europe, Africa, and large parts of Asia, and 60 Hz in the United States and parts of Latin America), isolating and analyzing the ENF component in recordings allows the forensic analyst to verify the authenticity by checking the (broad) geolocation and time of recording (provided that a database of ENF time series is available, as in many forensics departments of law enforcement agencies). Moreover, inconsistencies in the ENF frequency or phase can reveal editing operations. A particular challenge is related to testing new forensic methods for more exotic media types, formats, and editing operations. Reliable benchmark datasets often do not exist and are expensive to create, in particular if they should include controlled forgeries that are convincing to human perception. Simply repurposing data generated for other reasons (e. g.,

for compression or pattern recognition) is prone to fallacies and often not adequate because these sources hardly contain any (known) forgeries.

In the future, undoubtedly the relevance and scope of media forensics is going to grow. What we consider media data today (audio, images, video) continues to become ever more prevalent. In addition, as sensors become smaller, cheaper, and more pervasive, the authenticity of sensor data in general will be of utmost importance for many decisions that affect people's lives. Key questions will remain on what confidence we can put in forensic methods, passive and active alike, to deliver reliable results, automated and without human intervention, in environments with intelligent adversaries. We should embrace that new methods for sensor and signal forensics, in a very general sense, will be developed, critically evaluated, and deployed at large scale. Many of them will draw on principles inspired by information hiding and signal detection, the foundations of which were reviewed in this book.

ACKNOWLEDGMENTS

The authors thank Luisa Verdoliva for providing the materials for Figures 9.4 and 9.11. The work of the first author on this chapter has been supported by Archimedes Privatstiftung, Innsbruck, Austria.

References

[1] Ng, T.-T., et al., "Passive-Blind Image Forensics," in *Multimedia Security Technologies for Digital Rights Management*, Academic Press, pp. 383–412, 1st ed., 2006.

[2] Böhme, R., "An Epistemological Approach to Steganography," in *Information Hiding*, Vol. LNCS 5806 of *LNCS*, Berlin, Heidelberg: Springer Verlag, 2009, pp. 15–30.

[3] Böhme, R., et al., "Multimedia Forensics is not Computer Forensics," in *Computational Forensics, Third International Workshop*, Vol. 5718 of *Lecture Notes in Computer Science*, Springer, 2009, pp. 90–103.

[4] Lyu, S., and H. Farid, "How Realistic is Photorealistic?" *IEEE Transactions on Signal Processing*, Vol. 53, No. 2, 2005, pp. 845–850.

[5] Dehnie, S., H. T. Sencar, and N. Memon, "Digital Image Forensics for Identifying Computer Generated and Digital Camera Images," in *IEEE International Conference on Image Processing (ICIP)*, 2006, pp. 2313–2316.

[6] Ng, T.-T., and S.-F. Chang, "Discrimination of Computer Synthesized or Recaptured Images from Real Images," in *Digital Image Forensics: There is More to a Picture Than Meets the Eye*, Springer-Verlag, pp. 275–309, 2013.

[7] Dirik, A. E., et al., "New Features to Identify Computer Generated Images," in *IEEE International Conference on Image Processing (ICIP)*, Vol. 4, 2007, pp. 433–436.

[8] McKay, C., et al., "Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008, pp. 1657–1660.

[9] Kirchner, M., and T. Gloe, "Forensic Camera Model Identification," in *Handbook of Digital Forensics of Multimedia Data and Devices*, John Wiley & Sons Ltd., 9, 2015.

[10] Gloe, T., "Feature-Based Forensic Camera Model Identification," in *LNCS Transactions on Data Hiding and Multimedia Security VIII (DHMMS)*, Vol. 7228 of *Lecture Notes in Computer Science*, 2012, pp. 42–62.

[11] Popescu, A. C., and H. Farid, "Exposing Digital Forgeries in Color Filter Array Interpolated Images," *IEEE Transactions on Signal Processing*, Vol. 53, No. 10, 2005, pp. 3948–3959.

[12] Popescu, A. C., and H. Farid, "Exposing Digital Forgeries by Detecting Traces of Re-sampling," *IEEE Transactions on Signal Processing*, Vol. 53, No. 2, 2005, pp. 758–767.

[13] Kirchner, M., "Fast and Reliable Resampling Detection by Spectral Analysis of Fixed Linear Predictor Residue," in *Proceedings of the Multimedia and Security Workshop*, ACM Press, 2008, pp. 11–20.

[14] Kee, E., M. K. Johnson, and H. Farid, "Digital Image Authentication from JPEG Headers," *IEEE Transactions on Information Forensics and Security*, Vol. 6, No. 3, 2011, pp. 1066–1075.

[15] Gloe, T., "Forensic Analysis of Ordered Data Structures on the Example of JPEG Files," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2012, pp. 139–144.

[16] Johnson, M. K., and H. Farid, "Exposing Digital Forgeries in Complex Lighting Environments," *IEEE Transactions on Information Forensics and Security*, Vol. 2, No. 3, 2007, pp. 450–461.

[17] O'brien, J. F., and H. Farid, "Exposing Photo Manipulation with Inconsistent Reflections," *ACM Transactions on Graphics*, Vol. 31, No. 1, 2012, pp. 4:1–4:11.

[18] Kee, E., J. F. O'brien, and H. Farid, "Exposing Photo Manipulation from Shading and Shadows," *ACM Transactions on Graphics*, Vol. 33, No. 5, 2014.

[19] Popescu, A. C., and H. Farid, "Statistical Tools for Digital Forensics," in *Information Hiding. 6th International Workshop, IH 2004, Toronto, Canada, May 2004, Revised Selected Papers*, Vol. LNCS 3200 of *LNCS*, Berlin, Heidelberg: Springer Verlag, 2004, pp. 128–147.

[20] Chiang, P.-J., et al., "Printer and Scanner Forensics: Models and Methods," in *Intelligent Multimedia Analysis for Security Applications*, Springer Verlag, No. 282 in Studies in Computational Intelligence, pp. 145–187, 2010.

[21] Choi, K. S., E. Y. Lam, and K. K. Y. Wong, "Automatic Source Camera Identification Using Intrinsic Lens Radial Distortion," *Optics Express*, Vol. 14, No. 24, 2006, pp. 11551–11565.

[22] Gloe, T., S. Pfennig, and M. Kirchner, "Unexpected Artefacts in PRNU-Based Camera Identification: A 'Dresden Image Database' Case-Study," in *Proceedings of the Multimedia and Security Workshop*, ACM Press, 2012, pp. 109–114.

[23] Lyu, S., "Estimating Vignetting Function from a Single Image for Image Authentication," in *Proceedings of the Multimedia and Security Workshop*, New York: ACM Press, 2010, pp. 3–12.

[24] Johnson, M. K., and H. Farid, "Exposing Digital Forgeries through Chromatic Aberration," in *MM&Sec'06, Proceedings of the Multimedia and Security Workshop 2006*, New York: ACM, 2006, pp. 48–55.

[25] Gloe, T., K. Borowka, and A. Winkler, "Efficient Estimation and Large-Scale Evaluation of Lateral Chromatic Aberration for Digital Image Forensics," in *Proceedings of SPIE: Media Forensics and Security II*, Vol. 7541, 2010, p. 754107.

[26] Chierchia, G., et al., "A Bayesian-MRF Approach for PRNU-Based Image Forgery Detection," *IEEE Transactions on Information Forensics and Security*, Vol. 9, No. 4, 2014, pp. 554–567.

[27] Lukáš, J., J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, Vol. 1, No. 2, 2006, pp. 205–214.

[28] Chen, M., et al., "Determining Image Origin and Integrity Using Sensor Noise," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 1, March 2008, pp. 74–90.

[29] Goljan, M., J. Fridrich, and T. Filler, "Large Scale Test of Sensor Fingerprint Camera Identification," in *Proceedings of SPIE: Media Forensics and Security XI*, Vol. 7254, 2009, p. 72540I.

[30] Gloe, T., E. Franz, and A. Winkler, "Forensics for Flatbed Scanners," in *Proceedings of SPIE: Security, Steganography, and Watermarking of Multimedia Contents IX*, Vol. 6505, 2007, p. 65051I.

[31] Khanna, N., et al., "Scanner Identification with Extension to Forgery Detection," in *Proceedings of SPIE: Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, Vol. 6819, 2008, pp. 6819–16.

[32] Caldelli, R., I. Amerini, and F. Picchioni, "A DFT-Based Analysis to Discern between Camera and Scanned Images," *International Journal of Digital Crime and Forensics*, Vol. 2, No. 1, 2010, pp. 21–29.

[33] Geradts, Z. J., et al., "Methods for Identification of Images Acquired with Digital Cameras," in *Proceedings of SPIE: Enabling Technologies for Law Enforcement and Security*, Vol. 4232, 2001, pp. 505–512.

[34] Fridrich, J., and M. Goljan, "Determining Approximate Age of Digital Images Using Sensor Defects," in *Proceedings of SPIE: Media Forensics and Security III*, Vol. 7880, 2011, p. 788006.

[35] Dirik, A. E., H. T. Sencar, and N. D. Memon, "Digital Single Lens Reflex Camera Identification from Traces of Sensor Dust," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 3, September 2008, pp. 539–552.

[36] Swaminathan, A., M. Wu, and K. J. R. Liu, "Nonintrusive Component Forensics of Visual Sensors Using Output Images," *IEEE Transactions on Information Forensics and Security*, Vol. 2, No. 1, 2007, pp. 91–106.

[37] Dirik, A. E., and N. Memon, "Image Tamper Detection Based on Demosaicing Artifacts," in *IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 1497–1500.

[38] Kirchner, M., "Efficient Estimation of CFA Pattern Configuration in Digital Camera Images," in *Proceedings of SPIE: Media Forensics and Security II*, Vol. 7541, 2010, p. 754111.

[39] Cao, H., and A. C. Kot, "Accurate Detection of Demosaicing Regularity for Digital Image Forensics," *IEEE Transactions on Information Forensics and Security*, Vol. 4, No. 4, 2009, pp. 899–910.

[40] Swaminathan, A., M. Wu, and K. J. R. Liu, "Digital Image Forensics via Intrinsic Fingerprints," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 1, 2008, pp. 101–117.

[41] Stamm, M. C., and K. J. R. Liu, "Forensic Detection of Image Manipulation Using Statistical Intrinsic Fingerprints," *IEEE Transactions on Information Forensics and Security*, Vol. 5, No. 3, 2010, pp. 492–506.

[42] Neelamani, R., et al., "JPEG Compression History Estimation for Color Images," *IEEE Transactions on Image Processing*, Vol. 15, No. 6, 2006, pp. 1365–1378.

[43] Luo, W., J. Huang, and G. Qiu, "JPEG Error Analysis and Its Applications to Digital Image Forensics," *IEEE Transactions on Information Forensics and Security*, Vol. 5, No. 3, 2010, pp. 480–491.

[44] Bianchi, T., and A. Piva, "Image Forgery Localization via Block-Grained Analysis of JPEG Artifacts," *IEEE Transactions on Information Forensics and Security*, Vol. 7, No. 3, 2012, pp. 1003–1017.

[45] Pevný, T., and J. Fridrich, "Detection of Double-Compression in JPEG Images for Applications in Steganography," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 2, 2008, pp. 247–258.

[46] Ferrara, P., et al., "Reverse Engineering of Double Compressed Images in the Presence of Contrast Enhancement," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2013, pp. 141–146.

[47] Fu, D., Y. Q. Shi, and W. Su, "A Generalized Benford's Law for JPEG Coefficients and its Applications in Image Forensics," in *Security and Watermarking of Multimedia Content IX*, Vol. 6505 of *Proceedings of SPIE*, SPIE, 2007, p. 65051L.

[48] Pérez-González, F., G. L. Heileman, and C. T. Abdallah, "Benford's Law in Image Processing," in *IEEE International Conference on Image Processing (ICIP)*, Vol. 1, IEEE, 2007, pp. 405–408.

[49] Milani, S., M. Tagliasacchi, and S. Tubaro, "Discriminating Multiple JPEG Compressions Using First Digit Features," *APSIPA Transactions on Signal and Information Processing*, Vol. 3, 2014, pp. e19.

[50] Lai, S., and R. Böhme, "Block Convergence in Repeated Transform Coding: JPEG-100 Forensics, Darbon Dating, and Tamper Detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 3028–3032.

[51] Carnein, M., P. Schöttle, and R. Böhme, "Forensics of High-Quality JPEG Images with Color Subsampling," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2015.

[52] Lin, W. S., et al., "Digital Image Source Coder Forensics Via Intrinsic Fingerprints," *IEEE Transactions on Information Forensics and Security*, Vol. 4, No. 3, 2009, pp. 460–475.

[53] Gallagher, A. C., "Detection of Linear and Cubic Interpolation in JPEG Compressed Images," in *Canadian Conference on Computer and Robot Vision (CRV)*, 2005, pp. 65–72.

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

[55] Christlein, V., et al., "An Evaluation of Popular Copy-Move Forgery Detection Approaches," *IEEE Transactions on Information Forensics and Security*, Vol. 7, No. 6, 2012, pp. 1841–1854.

[56] Ryu, S.-J., et al., "Rotation Invariant Localization of Duplicated Image Regions Based on Zernike Moments," *IEEE Transactions on Information Forensics and Security*, Vol. 8, No. 8, 2013, pp. 1355–1370.

[57] Barnes, C., et al., "PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing," *ACM Transactions on Graphics*, Vol. 28, No. 3, 2009.

[58] Cozzolino, D., G. Poggi, and L. Verdoliva, "Efficient Dense-Field Copy-Move Forgery Detection," *IEEE Transactions on Information Forensics and Security*, Vol. 10, No. 11, 2015, pp. 2284–2297.

[59] Amerini, I., et al., "A SIFT–based Forensic Method for Copy-Move Attack Detection and Transformation Recovery," *IEEE Transactions on Information Forensics and Security*, Vol. 6, No. 3, 2011, pp. 1099–1110.

[60] Gloe, T., and R. Böhme, "The Dresden Image Database for Benchmarking Digital Image Forensics," *Journal of Digital Forensic Practice*, Vol. 3, 2010, pp. 150–159.

[61] Dang-Nguyen, D.-T., et al., "RAISE — A Raw Images Dataset for Digital Image Forensics," in *Proceedings of the 6th ACM Multimedia Systems Conference*, ACM, 2015, pp. 219–224.

[62] Barni, M., and F. Pérez-González, "Coping With the Enemy: Advances in Adversary-Aware Signal Processing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 8682–8686.

[63] Gloe, T., et al., "Can We Trust Digital Image Forensics?" in *15th International Conference on Multimedia*, ACM Press, 2007, pp. 78–86.

[64] Kirchner, M., and R. Böhme, "Hiding Traces of Resampling in Digital Images," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 4, 2008, pp. 582–592.

[65] Goljan, M., J. Fridrich, and M. Chen, "Defending Against Fingerprint-Copy Attack in Sensor-Based Camera Identification," *IEEE Transactions on Information Forensics and Security*, Vol. 6, No. 1, 2011, pp. 227–236.

[66] Lai, S., and R. Böhme, "Countering Counter-Forensics: The Case of JPEG Compression," in *Information Hiding, 13th International Conference*, Vol. 6958 of *Lecture Notes in Computer Science*, Springer Verlag, 2011, pp. 285–298.

[67] Dirik, A. E., H. T. Sencar, and N. Memon, "Analysis of Seam-Carving-Based Anonymization of Images Against PRNU Noise Pattern-Based Source Attribution," *IEEE Transactions on Information Forensics and Security*, Vol. 9, No. 12, 2014, pp. 2277–2290.

[68] Böhme, R., and M. Kirchner, "Counter-Forensics: Attacking Image Forensics," in *Digital Image Forensics: There is More to a Picture Than Meets the Eye*, Springer-Verlag, pp. 327–366, 2013.

[69] Kee, E., and H. Farid, "A Perceptual Metric for Photo Retouching," *Proceedings of the National Academy of Sciences*, Vol. 108, No. 50, 2011, pp. 19907–19912.

[70] Böhme, R., and A. Westfeld, "Feature-based Encoder Classification of Compressed Audio Streams," *Multimedia Systems*, Vol. 11, No. 2, 2005, pp. 108–120.

[71] Milani, S., et al., "An Overview on Video Forensics," *APSIPA Transactions on Signal and Information Processing*, Vol. 1, 2012.

[72] Stamm, M. C., W. S. Lin, and K. J. R. Liu, "Temporal Forensics and Anti-Forensics for Motion Compensated Video," *IEEE Transactions on Information Forensics and Security*, Vol. 7, No. 4, 2012, pp. 1315–1329.

[73] Grigoras, C., "Digital Audio Recording Analysis: the Electric Network Frequency (ENF) Criterion," *Speech, Language and the Law*, Vol. 12, No. 1, 2005, pp. 63–76.

[74] Garg, R., A. L. Varna, and M. Wu, "Seeing ENF: Natural Time Stamp for Digital Video via Optical Sensing and Signal Processing," in *ACM International Conference on Multimedia*, ACM Press, 2011, pp. 23–32.

[75] Su, H., et al., "Exploiting Rolling Shutter for ENF Signal Extraction from Video," in *IEEE International Conference on Image Processing (ICIP)*, IEEE, 2014, pp. 5367–5371.