Engineering 4 (2018) 29–39

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng

Research
Cybersecurity—Review

Recent Advances in Passive Digital Image Security Forensics: A Brief Review

Xiang Lin [a], Jian-Hua Li [a], Shi-Lin Wang [a,*], Alan-Wee-Chung Liew [b], Feng Cheng [a], Xiao-Sa Huang [a]

[a] School of Cyber Security, Shanghai Jiao Tong University, Shanghai 200240, China
[b] School of Information and Communication Technology, Gold Coast Campus, Griffith University, Southport, QLD 4222, Australia

ARTICLE INFO

Article history:
Received 8 December 2017
Revised 20 December 2017
Accepted 15 February 2018
Available online 17 February 2018

Keywords:
Digital image forensics
Image-tampering detection
Multimedia security

ABSTRACT

With the development of sophisticated image editing and manipulation tools, the originality and authenticity of a digital image is usually hard to determine visually. In order to detect digital image forgeries, various kinds of digital image forensics techniques have been proposed in the last decade. Compared with active forensics approaches that require embedding additional information, passive forensics approaches are more popular due to their wider application scenario, and have attracted increasing academic and industrial research interests. Generally speaking, passive digital image forensics detects image forgeries based on the fact that there are certain intrinsic patterns in the original image left during image acquisition or storage, or specific patterns in image forgeries left during the image storage or editing. By analyzing the above patterns, the originality of an image can be authenticated. In this paper, a brief review on passive digital image forensic methods is presented in order to provide a comprehensive introduction on recent advances in this rapidly developing research area. These forensics approaches are divided into three categories based on the various kinds of traces they can be used to track—that is, traces left in image acquisition, traces left in image storage, and traces left in image editing. For each category, the forensics scenario, the underlying rationale, and state-of-the-art methodologies are elaborated. Moreover, the major limitations of the current image forensics approaches are discussed in order to point out some possible research directions or focuses in these areas.

1. Introduction

In the past decade, digital images have become more and more popular in our daily life. Compared with traditional text content, images are more intuitive and can convey much more information. Despite these benefits, the easy accessibility of digital images has resulted in significant security problems—that is, how to examine the authenticity of digital images and how to detect malicious modification. With advanced image processing software, it is becoming easier and easier to manipulate images without leaving visually noticeable traces, making the aforementioned problems more challenging [1].

To authenticate image contents and detect image forgeries accurately and robustly, researchers have proposed various approaches in digital image forensics. Generally speaking, there are two major categories of digital image forensics approaches: active approaches and passive approaches [2]. Active forensics approaches usually involve designing various kinds of watermarks or fingerprints of the image content and embedding them into the digital image. In the authentication stage, the former embedded watermarks or fingerprints are extracted and examined to determine whether the original image has been tampered with and, if so, where the tampered location is [3]. These active approaches can detect digital image tampering accurately; however, they have not been widely used because it is not possible to require all the digital images on the Internet to be watermarked before distribution. Hence, passive forensics approaches have become a more popular choice. These approaches detect digital image forgeries by analyzing specific inherent clues or patterns that occur during the creation/modification stage of digital images [4]. Compared with active forensics approaches, passive approaches do not rely on any prior or preset information; thus, they can have a broader application in image forensics.

For passive digital image forensics, various kinds of traces can be exploited to differentiate tampered images from natural ones

* Corresponding author.
E-mail address: wsl@sjtu.edu.cn (S.-L. Wang).

https://doi.org/10.1016/j.eng.2018.02.008

[5]. In this paper, we have categorized these traces into three groups: traces left in image acquisition, traces left in image storage, and traces left in image editing. For each kind of traces, we will briefly review the corresponding passive digital image forensics approaches by clarifying the following issues:

- What are these traces and how do they form?
- What are the relevant recent and state-of-the-art approaches in image forensics?
- Why can these approaches detect these particular traces?

The paper is organized as follows. In Sections 2–4, we introduce various passive digital image forensics approaches, respectively: detecting the traces in image acquisition, the traces in image storage, and the traces in image editing. Section 5 concludes with a discussion of the major limitations of current techniques, and presents some possible future directions.

2. Traces left in image acquisition

When a digital image is captured, several processing steps are performed before storage (Fig. 1). Before entering the imaging device, the natural light usually goes through a series of lenses. The imaging device then conveys it to the color filter array (CFA), where a specific color mosaic is performed that only permits a certain component of the light to pass through a specific area. In most digital cameras, only one main color (red, green, or blue, or RGB) is allowed for each pixel. After CFA filtering, the light reaches the imaging sensor, which is the crucial part of the digital camera. At present, there are two widely used sensors: the charge-coupled device (CCD) and the complementary metal-oxide semiconductor (CMOS). A number of photo detectors are contained in the imaging sensor and each of their output is related to a pixel of the image. In each detector, the filtered light is transformed into a corresponding voltage; thus, the output of the sensor is a mosaic of RGB pixels with various intensity values. In order to obtain the integrated color information for every pixel in the image, a demosaicing procedure has been adopted. An interpolation process has been performed on all the color channels; thus, the missing color components can be estimated.

Each of the above stages will introduce specific imperfections or patterns into the final image; these can be adopted as useful clues in image source identification and tampering detection. The following subsections present a brief introduction to the traces left by the lens, sensor, and CFA interpolation; typical image forensics approaches using these traces; and the state-of-the-art forensics approaches used in this case.

2.1. Traces left by acquisition artifacts

Owing to the design and manufacturing process, distortions or aberrations will be introduced by the lens into the captured images. There are two kinds of common distortions in digital cameras: chromatic aberration (CA) and spherical aberration. The former distortion will appear when the lenses cannot focus colors with different wavelengths to the same convergence position on the sensor. The latter aberration will appear when lights passing through the lens do not converge at the focal point.

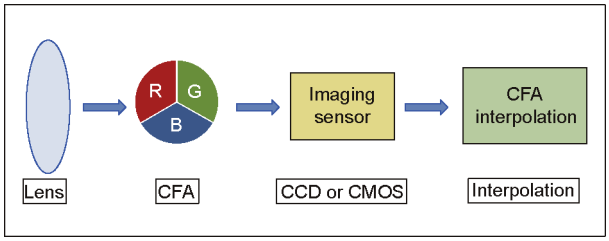

Fig. 1. Flowchart of the digital image acquisition procedure.

Sensor noise is another important acquisition characteristic. Among the various kinds of sensor noises, noise caused by the photo response non-uniformity (PRNU) is of the greatest importance. Various kinds of digital image forensics approaches have been proposed based on the PRNU noise; these approaches cover the applications of source identification, processing history recovery, image forgery detection, and more. The PRNU is a distinguishing characteristic that is related to each imaging sensor. Hence, by analyzing the PRNU in each captured image, the traces of the capturing device can be obtained.

Another aspect that cannot be ignored is the demosaicing of artifacts in the CFA. In order to estimate the corresponding values of each pixel in the image, an interpolation procedure has been applied to the three color channels. The interpolation process inevitably introduces certain correlations among the pixels, and such correlations can be adopted as an intrinsic "fingerprint" of the capturing device. By analyzing the demosaicing patterns, traces of the capturing devices can be obtained.

2.2. Image forensics using acquisition artifacts

2.2.1. Image forensics based on sensor noise

Using the aberrations produced by the camera lens, we can relate an image to a specific device or examine whether an image has been tampered. For example, radial distortion is usually observed in an image, as the straight lines of an object appear to be curved. In order to deal with radial distortion, digital camera manufacturers usually adopt various methods to compensate for the distortion; these methods produce different artifacts accordingly. Hence, by analyzing such artifacts, the camera manufacturer and even the camera model can be identified.

Sensor pattern noise is commonly used in image forensics. For the image I, the pattern noise can be formulated as follows:

$$R = I - F(I) = I \times P + \varphi \tag{1}$$

where R is the overall residual, which can be obtained from the original image by deducting the counterpart after passing the image through a de-noising filter F; P is the PRNU factor; and φ is the summation of all the other kinds of noises in the image.

Assuming that we have N images captured by the same camera—that is, images I_1 to I_N—the corresponding residual R_k can be calculated using Eq. (1). The PRNU factor P can be estimated following the maximum likelihood criterion and is formulated as follows:

$$P = \frac{\sum_{k=1}^{N} R_k I_k}{\sum_{k=1}^{N} (I_k)^2} \tag{2}$$

In camera identification, supposing that there are M devices, the PRNU factor should be calculated M times with a specific P_i value being recorded for each device ($i = 1, 2, \ldots, M$). In the test stage, the residual term is first calculated using Eq. (1) as $R_t = I_t - F(I_t)$ for the test image I_t. Then the correlation between the PRNU factors and this residual, R_t, is computed as follows:

$$\tilde{n}_i = I_t P_i \otimes R_t \tag{3}$$

where $\otimes$ denotes normalized correlation. The capturing device is identified as the one with maximum $\tilde{n}_i$.

2.2.2. Image forensics based on CFA patterns

The underlying rationale of the forensics approaches that are based on CFA artifacts is that the original image will have specific CFA artifacts, whereas the tampered region will probably have a

different kind of artifact. Hence, the CFA artifacts are computed for each block in the image that is being investigated, with different CFA patterns indicating the existence and location of forgery. In addition, since a different demosaicing algorithm introduces different correlations between neighboring pixels in a specific color channel, there are two major directions in image forensics based on CFA patterns. The first direction aims to predict the interpolation parameters and identify the type of capturing device, while the second aims to examine the demosaicing traces so as to locate the region that has potentially been tampered with.

2.3. State-of-the-art acquisition artifacts detection approaches

2.3.1. Source identification based on lens aberrations

As discussed in Section 2.1, different cameras have different kinds of lens aberrations, so lens aberrations can be adopted as a unique "fingerprint" in source identification. Choi et al. [6] proposed a pioneering work in this area: Considering the characteristics of the radial distortion that forces straight lines to become curved, they proposed two kinds of features, based on the pixel intensities and the distortion measurements. Choi et al. [6] then performed source identification as a classification procedure and achieved a 4% detection accuracy gain compared with approaches that only use image intensities.

2.3.2. Tampering detection based on lens artifacts

The basic idea behind using lens artifacts to detect image-splicing forgery is that the original image and the inserted image patch are probably captured using different devices; thus, detecting and locating image regions with different lens artifacts may identify image forgery. Yerushalmy and Hel-Or [7] proposed a new type of lens artifact, which is referred to as the "purple fringing aberration" (PFA); they also presented a corresponding extraction method. The PFA directions are adopted as unique "fingerprints" to determine whether there is any inconsistency in the image being tested. The algorithm achieved good performance in both image forgery detection and tampering detection.

2.3.3. Source identification based on sensor pattern noise

Owing to deficiencies in the sensor manufacturing process, different pixels will have dissimilar light sensitivities; therefore, the sensor pattern noise, and especially the PRNU, can be used to differentiate various kinds of sensors and camera types. Lukas et al. [8] proposed a source identification approach based on the PRNU that can identify nine camera models. Kulkarni and Mane [9] noticed that an estimation of the sensor noise may not be very accurate in the edge region, and proposed a preprocessing step before feature extraction. Two edge detectors, the Canny and Laplace operators, were adopted to detect the edge regions; these regions are then removed for further processing. After calculating the sensor noise by thresholding, a number of statistical features were extracted from the discrete wavelet transform (DWT) domain in the form of the gray-level co-occurrence matrix (GLCM). Finally, the *k*-nearest neighbor (*k*-NN) is adopted as the classifier. Considering the applications for camera identification in mobile phones, Sandoval Orozco et al. [10] proposed a specific approach based on sensor imperfections. Their features were also extracted from the wavelet domain, and this approach achieved good results in digital camera identification in mobile phones.

2.3.4. Tampering detection based on sensor fingerprint inconsistencies

Similar to the lens artifacts, the sensor noise can be used to detect image forgery. The underlying idea is that there will be inconsistencies in the sensor noise/fingerprints in the tampered region, allowing a suspicious tampered region to be detected. Fridrich [11] proposed an image-tampering detection approach based on the PRNU information. A statistical model was built to describe the PRNU factor. The PRNU factor of the image being tested was extracted and used to identify the capturing device. The experimental results showed that this approach can achieve almost 100% accuracy for 100 different types of cameras.

2.3.5. Source identification based on CFA artifacts

As mentioned earlier, the CFA and the demosaicing process vary among various cameras; Gao et al. [12] proposed a source identification method based on this information. A 69 dimensional feature was designed to describe the abovementioned artifacts. The experiments were carried out on the Dresden Image Database, and a very high detection accuracy of 99.88% was achieved in differentiating seven camera models.

2.3.6. Tampering detection based on CFA artifacts

The basic idea of using CFA artifacts in tampering detection is straightforward: The tampered region will exhibit different CFA/demosaicing artifacts than those of the original image. Prasad [13] proposed a feature to describe the demosaicing artifacts in images. If an abnormal region is present (i.e., one without the original artifacts or having different artifacts), it is regarded as a tampered region. Katre and Chandel [14] proposed an approach that can both detect image forgeries and locate the tampered region. The artifacts caused by demosaicing were modeled, which helped to reveal the image forgery. Their approach achieved good performance for uncompressed images; however, the question of how to deal with Joint Photographic Experts Group (JPEG) compression remained a challenging one.

3. Traces left in image storage

JPEG is the most widely used format for image transmission and storage. Since it is a lossy compression standard, JPEG will inevitably introduce certain compression patterns during each image storage. By analyzing these patterns, it is possible to deduce important forensic cues, such as ① how many times the image has been compressed and ② whether all the regions in the image have been compressed the same number of times. The following subsections present a brief introduction to the patterns left by JPEG compression, typical scenarios in JPEG compressed image forensics, and state-of-the-art forensics approaches in this area.

3.1. Patterns left by JPEG compression

A standard JPEG compression procedure for grayscale images runs as follows (note that this procedure can be extended to color images by performing a similar approach in each channel of the YCbCr color space): A non-overlapping 8×8 block division is performed on an original image. For each block, a two-dimensional discrete cosine transform (2D-DCT) is applied on the grayscale values to transform all the pixels to the frequency domain. Next, the amplitudes of the frequency components are quantized by a preset quantization table. Fig. 2 shows a typical quantization table with a quality factor of 50, where larger quality factors represent a higher image quality and lower compression ratio. Finally, an entropy coding technique (e.g., Huffman coding) is adopted to turn the quantized frequency amplitude into a binary sequence.

The JPEG compression will introduce three kinds of bias to the original image: the quantization error, truncation error, and rounding error. The quantization error is caused by the quantization process in the frequency domain. After quantization, the original value of a specific DCT component will be represented by the closest integer multiples of the corresponding quantization step. For example, given an original direct current (DC) value of 86 and a

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig. 2. A quantization table with a quality factor of 50.

quantization step of 16 (as shown in Fig. 2), the DC value will be changed to 80 after quantization and the difference is denoted as the quantization error (i.e., 86 – 80 = 6). The truncation error and rounding error are introduced in the inverse discrete cosine transform (IDCT) transform. As we know that the grayscale value should be an integer ranging from 0 to 255, any value greater than 255 or less than 0 will be truncated to 255 or 0, correspondingly, which leads to the truncation error. On the other hand, most values after IDCT are not integers, so a rounding process must be performed, which leads to rounding errors. Generally speaking, the quantization error is much greater than the other two errors, especially when the quality factor is medium or low (< 75). By analyzing the quantization error, some clues about JPEG compression can be derived.

3.2. Image-tampering detection using clues from JPEG compression

Consider a simple image-tampering scenario. We crop the small Patch I from Image B and insert it into Image A to generate a composite Image C (Fig. 3). If all the images—A, B, and C—are stored using JPEG compression, the following issues can be observed.

3.2.1. Aligned double JPEG compression

In Image C, all the regions except Patch I are compressed twice (during the storing of Image A and the storing of Image C), whereas Patch I is compressed once. Note that although Image B has also been JPEG compressed, the 8 × 8 block division structure of Image B is very likely (with a probability of 63/64) to be different from that of Image A with regards to Patch I. Thus, Patch I in Image C is compressed only once according to the 8 × 8 block division structure of Image C (which is the same as that of Image A). Hence, the inserted Patch I can be located by examining the aligned double JPEG compression effects in all the regions. Here, the term "aligned" refers to the first and second JPEG compression using the same 8 × 8 block division structure.

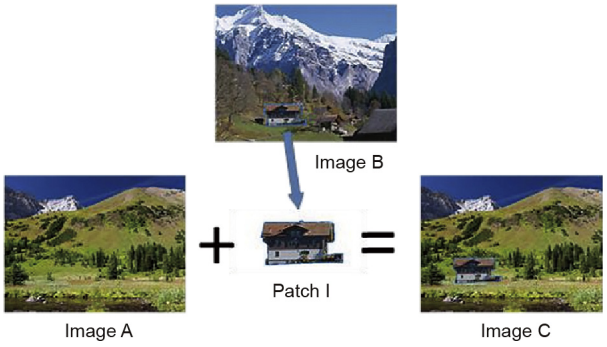

Fig. 3. An illustration of copy-paste image tampering.

3.2.2. Shifted (non-aligned) double JPEG (SDJPEG) compression

The image in Patch I is focused on in this part. The image in Patch I is first compressed when storing Image B using a specific block division structure; it is then compressed again during the storing of Image C, using another structure. Double JPEG compression with different block division structures is referred to as a "shifted" or "non-aligned" case. By examining all the regions in Image C, the inserted Patch I can be located based on the SDJPEG compression effects.

3.3. State-of-the-art double JPEG compression detection approaches

For most tampered images, at least two JPEG compressions have been applied to the original image. Hence, double JPEG compression detection is a crucial part in digital image forensics. The representative aligned double JPEG compression detection approaches can be found in Refs. [15–21]. One of the earliest works was proposed by Lukas and Fridrich [15], who noticed that double JPEG compression would generate two peaks in the DCT histogram; this phenomenon was adopted as a clue to detect double JPEG compressed regions. Fu et al. [16] proposed a double JPEG compression detection approach based on the generalized Benford's law. They assumed that the first digits of the DCT coefficients after JPEG compression form a distribution that follows the generalized Benford's law; thus, any image region violating this assumption is determined to be a double JPEG compressed region. Pevny and Fridrich [17] extracted a number of low-frequency DCT coefficient histograms as distinguishing features and used the support vector machine (SVM) as the classifier. Farid [18] stated that if the image was double JPEG compressed, recompression with the same quality factor would cause the minimum reconstruction error. He referred to such a local minimum as a "JPEG ghost," and tried to detect double compression by searching for JPEG ghosts. Lin et al. [19] proposed a tampered region-locating algorithm based on double JPEG compression detection. The distribution of DCT coefficients after single JPEG compression was assumed to follow the Laplace distribution; the distribution after double JPEG compression can then be derived accordingly. By using an expectation maximization (EM) optimization algorithm, each region was assigned a probability of being doubly compressed. A graph cut algorithm was then adopted to avoid false alarms. This algorithm usually works well for double compression with different quality factors. However, when two compressions use the same quantization table, most of the existing approaches cannot achieve a high detection accuracy. In view of this, Huang et al. [20] designed an algorithm only for detecting double JPEG compression with the same quality factor. A random perturbation strategy was adopted based on the fact that the differences between the first and second compressions are much greater than those between the second and third compressions. In a recent work, Yang et al. [21] extended the idea in Ref. [20] and comprehensively analyzed the error blocks in JPEG compression. The rounding errors and truncation errors were analyzed, and a set of features describing the differences between single and double JPEG compressions was extracted. By using the SVM as the classifier, their algorithm [21] can accurately detect double compression with the same quality factor when the quality factor is relatively high.

When the original image (i.e., Image A in Fig. 3) is uncompressed, aligned double JPEG compression effects will not exist in the composite image (i.e., in Image C in Fig. 3). In order to detect image forgeries in such a case, the researchers try to investigate the SDJPEG compression effects [22–27]. Luo et al. [22] detected SDJPEG compression effects by introducing a specific feature called

the blocking artifact characteristics matrix (BACM). For single JPEG compressed images, it is observed that the corresponding BACMs are symmetric, whereas the BACMs for SDJPEG compressed images are no longer symmetric. However, the BACM feature is related to the image content to some extent; thus, different image contents may lead to different kinds of BACM features, which may degrade the double JPEG compression detection performance. In order to solve the content-related problem, an extended BACM feature was proposed by Chen and Hsu [23] by considering the inter-block correlation. Qu et al. [24] proposed a convolutive mixing model for SDJPEG compression, and solved it using blind signal separation. They analyzed the independent value map (IVM) to examine whether the image had undergone SDJPEG compression, with the assumption that SDJPEG compression will break the symmetry of the IVM. Bianchi and Piva [25] tried to disclose SDJPEG compressed traces from the DCT coefficient histograms. A complete search was performed to examine each region with a specific size in order to determine whether it was SDJPEG compressed or not. When the searching region matches the block division of the first JPEG compression, an integer periodicity pattern can be observed in its DCT coefficient histogram. Bianchi and Piva [26] also proposed a statistical model for the DCT coefficient distributions caused by SDJPEG compression. They simulated the SDJPEG compression by adding a zero-mean Gaussian noise to each DCT coefficient; they also provided a variance estimation method to approximate the noise variance. Recently, Wang et al. [27] extended the idea in Ref. [26] and provided a complete theoretical proof on how the SDJPEG compression affects the DCT coefficient distributions.

The major limitations of the current double JPEG compression detection algorithm are threefold:

(1) Most techniques are based on statistical features or models. When the tampered region is small enough (i.e., smaller than 64 × 64), most techniques cannot provide accurate results, since the data for constructing the statistical features (models) are quite limited.

(2) Most techniques can achieve good results when the first quantization table is known or can be accurately estimated. However, such an assumption cannot be easily achieved. Due to the error propagation, the overall detection accuracy will decrease to some extent during the exhaustive search for the first quantization table.

(3) For SDJPEG compression detection, when the quality factor of the second quantization is much less than that of the first quantization, most state-of-the-art detection approaches show a performance that is similar to random guessing.

4. Traces left in image editing

4.1. Inconsistency in lighting

Copy-paste image tampering is the most common method of creating an image forgery. These forgeries are usually deceptive to the human eye; however, they may have some inconsistencies in lighting, shadows, perspective, and so forth, which can be detected by proper analysis. Fig. 4 [28] shows a well-known case of lighting inconsistencies. When a picture is captured, the objects in the scene are illuminated by a light source coming from a certain direction (Fig. 5 [28]). If two objects originally come from different images, it is unlikely that their light sources will be similar in direction and distance. The following subsections present a brief introduction to image forgery detection by analyzing lighting consistencies.

4.1.1. Lighting traces

When an image is captured, the objects captured in the scene are illuminated by a light source. This light source will leave traces on the objects, such as intensity differences on the surface, and shadows. Hence, the light direction can be inferred from these traces. If the objects in an image are illuminated by different light sources, these objects are very unlikely to belong to the same original scene—which indicates forgery.

4.1.2. Image-tampering detection using a lighting pattern

The lighting environment in the real world is complex. Light sources are three dimensional (3D), and sometimes there are multiple light sources. Hence, assumptions are made to simplify the problem: the Lambertian assumption of the surface of interest, the constant assumption of the object's reflectance, and the assumption that the light source is located infinitely far away. With these assumptions, Johnson and Farid [28] described the image intensity, I_s, as follows:

$$I_s(x,y) = R_f(\boldsymbol{N}(x,y) \cdot \boldsymbol{L}) + C \tag{4}$$

where R_f is the object's constant reflectance value, $\boldsymbol{L}$ is the light source direction, $\boldsymbol{N}(x,y)$ is the surface normal direction at the

Fig. 4. The light source directions of the two people in this image are different, which may be evidence of forgery [28].

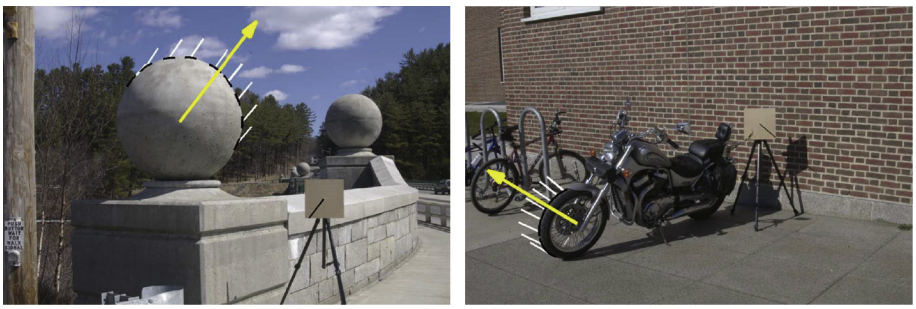

Fig. 5. In different images, the light source directions are different [28].

coordinate (x, y), and C is the constant ambient light. Moreover, R_f can be regarded as a unit value when only the light direction is required for estimation. Given knowledge of 3D surface normals from at least p distinct points ($p \geqslant 4$) on a surface with the same reflectance, the least-squared estimation is adopted to calculate the light direction, which is formulated as follows:

$$E(\boldsymbol{L}, C) = \left\| M \begin{pmatrix} L_x \\ L_y \\ L_z \\ C \end{pmatrix} - \begin{pmatrix} I_s(x_1, y_1) \\ I_s(x_2, y_2) \\ \vdots \\ I_s(x_p, y_p) \end{pmatrix} \right\|^2 = \|M\upsilon - \boldsymbol{b}\|^2 \tag{5}$$

where $\|\|$ represents the vector norm, $\boldsymbol{L}$ is the light source direction containing three components (L_x, L_y, and L_z), and

$$M = \begin{pmatrix} N_x(x_1, y_1) & N_y(x_1, y_1) & N_z(x_1, y_1) & 1 \\ N_x(x_2, y_2) & N_y(x_2, y_2) & N_z(x_2, y_2) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ N_x(x_p, y_p) & N_y(x_p, y_p) & N_z(x_p, y_p) & 1 \end{pmatrix} \tag{6}$$

where N_x, N_y, and N_z are the three components of the surface normal direction. Setting the partial derivative at zero, the least-squared estimate can be calculated as follows:

$$\upsilon = (M^{\mathrm{T}}M)^{-1}M^{\mathrm{T}}\boldsymbol{b} \tag{7}$$

However, with only a single image and without knowing the geometry of the objects in the scene, it is unlikely that the object surface normal $\boldsymbol{N}(x, y)$ can be obtained. A good solution is to simplify a 3D scene to a 2D one, which means that we only need to estimate two components, (L_x, L_y), of the light source direction from a digital image.

In some scenes, the simplifying assumptions are too strict, such that forgeries cannot be detected. In these cases, we can relax some simplifying assumptions. For example, we can relax the constant reflectance assumption; different light source directions on each patch of the surface must then be estimated. Moreover, when the light source is local, the assumption that the light source is infinitely far away is invalid. In that case, one more assumption must be added: that the light source direction is constant for a local patch.

4.1.3. State-of-the-art lighting inconsistency detection approaches

Lighting traces are quite complex in the real world. For indoor scenes, there may be multiple local light sources. For outdoor scenes, there is very likely a one-point light source from infinitely far away. In some scenes, we cannot calculate the 3D components of the light direction due to unknown geometry. Most of the detection approaches are based on particular scenes or have some defects. Johnson and Farid [28] proposed one of the earliest attempts at detection using lighting inconsistencies. Although this method had some difficulties in calculating the object surface normal in many cases, it was observed that for images containing human faces, the light source in the scene can be estimated from the highlights on human eyes. Johnson and Farid [29] then estimated the light direction from this highlight based on a 3D model of a human eye, and proposed a lower-dimensional model in Ref. [30] to deal with a complex lighting environment. Kee and Farid [31] further built a 3D head model to improve the estimation performance for a complex lighting environment. Nillius and Eklundh [32] presented an automatic light source direction estimation algorithm from a single image. The algorithm in Ref. [32] required at least one occluding object contour with isotropic surface reflectance in the image—a requirement that can be easily satisfied in many images with various contents. In the algorithm, the occluding contours were first extracted based on the color and edge information. A shading model was then adopted to estimate the light source direction for each contour, and a Bayesian network was designed to fuse all the estimation results and output the most likely estimation. It is worth noting that in Ref. [32], the shadows of the objects could also be used to detect forgery. Koenderink et al. [33] modeled the illumination process as a parallel light beam from random directions projecting on random Gaussian surfaces, and proposed an illumination direction estimation method accordingly. Zhang et al. [34] adopted the planar homology and shadow matte to describe the color distribution/characteristics and relationship of the shadows in the image. A framework based on photometry and geometry of the image shadow was proposed to detect image forgeries. Fan et al. [35] designed a straightforward counter-forensics strategy to outdo the 2D lighting consistency-based forensics approaches; this strategy disposed the shortcomings of the existing approaches in this area and provided new challenges for future research. Moreover, it was shown that lighting traces are more effective for outdoor scenes due to the simple lighting environment.

4.2. Local filtering traces

4.2.1. Median filtering detection

Median filtering (MF) is a common image post-processing method to filter out image noise. With its nonlinear property, it also can be used in image forgeries to remove specific modification traces. The MF result is obtained by the median value of a small window surrounding a specific pixel. Generally speaking, the square window size is set to odd values, such as 3×3 or 5×5. Since the MF takes the median value as the filtered result, it usually generates many constant or nearly constant patches in the filtered image; these patches become intrinsic footprints and make the MF procedure traceable and detectable.

The general form of a 2D median filter is given as follows:

$$y_i = \mathrm{median}(x_{i+r,j+c}), \quad r, c \in [-z/2, z/2] \tag{8}$$

where y_i is the output of the median filter on a pixel (i, j) with a squared window of the size $[z, z]$.

The nonlinear nature of MF makes it difficult to build up an explicit expression describing the relationship between its input and output. However, Bovik et al. [36] demonstrated that MF has a good property of edge preservation. In addition, constant or nearly constant image patches are usually found in an image after MF; this is referred to as the streaking artifact [37], as shown in Fig. 6. Kirchner and Fridrich [38] first tried to exploit the streaking artifact of MF images and proposed a pair of MF detectors using this artifact. The histogram in the first-order difference image was analyzed to examine the streaking artifacts, and the approach showed good results for uncompressed images. In order to detect MF in JPEG compressed images, the subtractive pixel adjacency matrix (SPAM) features [38], which show good results in steganalysis, were adopted for MF detection. However, since the SPAM is a complex statistical model, the detection performance will degrade when the number of pixels in the image region is reduced to a relatively small value (e.g., 64 or 128). Similarly, Cao et al. [39] proposed an MF detection algorithm based on the streaking artifacts. After computing the difference images of the texture region in both the horizontal and vertical directions, the probabilities of zero values were recorded. Moreover, the algorithm developed by Cao et al. [39] can differentiate MF from other local image processes such as rescaling, Gaussian low-pass filtering, and averaging filtering. Yuan [40] observed that MF results have a local dependency artifact because the local filtering windows of adjacent pixels are overlapped with each other during

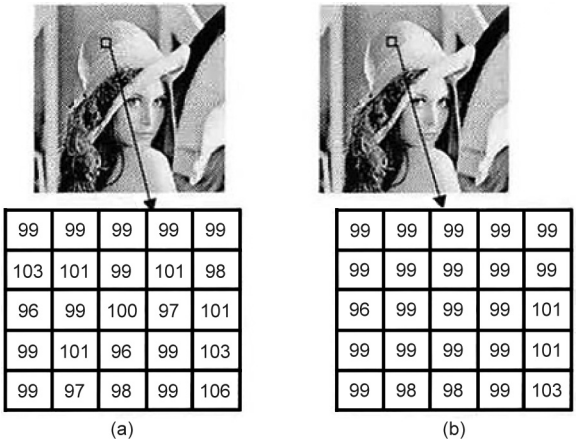

Fig. 6. Pixel distribution (a) before and (b) after MF.

MF, and a number of pixels are shared between adjacent local windows. Based on this observation, 44 features measuring the intrinsic relationships among the adjacent pixels were proposed for MF detection, as what is referred to as the median filtering feature (MFF). The MFF can detect MF very accurately for uncompressed images and can achieve similar results when compared with the SPAM feature in JPEG compressed images. Moreover, for JPEG compressed images with a low quality factor, the MFF usually outperformed the SPAM feature.

The major drawback of using streaking artifacts for MF detection is that these approaches are not robust against certain types of image post-processing steps, such as JPEG compression. Hence, other kinds of MF traces should be exploited. Chen and Ni [41] observed that images after MF show different image characteristics in the edge region, compared with the original images. Such characteristics are represented in the specific correlation between neighboring pixels and the relationship between image noise and edges [41]. Taking the above characteristics into consideration, the edge-based prediction matrix (EBPM) feature was proposed for MF detection; this feature is extracted from various edge regions in the image, and the SVM is adopted for classification. Compared with previous methods, the EBPM showed better performance in distinguishing MF from other filtering processes such as Gaussian low-pass filtering and averaging filtering. Kang et al. [42] adopted the median filter residual (MFR) as evidence to detect MF. The residual was defined as the difference between the image and its median filtered output. The authors pointed out that when an image is median filtered once more, the MFR will be reduced; they then proposed an MF detection approach based on this observation. An auto-regressive (AR) model was employed to construct an MF detection feature set based on the MFR. Compared with SPAM feature and MFF, the MFR feature has a lower dimension (it is 10 dimensional) and achieves comparable detection results, even when the quality factor of the JPEG compressed image is low (e.g., 30). Chen et al. [43] exploited both the global information and local information for MF detection. In the global sense, a number of cumulative distributions of various orders of the difference images were adopted as global features. In the local sense, the correlations between various neighboring pixel pairs were adopted as local features. The final feature was constructed by concatenating the global and local features, and had a dimensionality of 56. Their approach is successful at detecting MF in low-resolution and JPEG compressed images with low quality factors.

Some modern techniques, including the local texture descriptor and deep learning, have recently been adopted for MF detection. Zhang et al. [44] proposed a local texture descriptor, referred to as the second-order local ternary pattern (LTP), for MF detection. The proposed feature combined the merits of the LTP feature and those of the local derivative pattern feature; thus, it can better describe the local characteristics caused by MF. Moreover, kernel principal component analysis (KPCA) was employed for feature dimension reduction and discriminative feature extraction. Using the approaches in Ref. [44], median filtered images can be detected accurately and efficiently. Chen et al. [45] aimed to detect MF artifacts in a challenging small-sized image patch scenario. A convolutional neural network (CNN) was designed for both feature extraction and classification. The image patch is adopted as the input of the network, and the detection result is the network output. In Ref. [45], the features were learned automatically from training samples, and no manual feature extraction procedure was required. This approach showed significant performance improvements in MF detection, especially when the image patch is small.

4.2.2. Unsharp masking sharpening detection

Unsharp masking (USM) sharpening is a technique that is widely used in daily life: It enhances edge contrast in order to improve image quality. However, in many image forgeries, USM sharpening can also be employed to cover the traces of image manipulation to some extent. Hence, USM sharpening detection has become a hot research topic in digital image forensics. The USM sharpening process usually contains the following two steps:

Step 1: Perform Gaussian high-pass filtering.

$$H(x,y) = I_i(x,y) - I_i(x,y) \otimes G_\sigma \tag{9}$$

where H is the high pass filter; I_i is the original image; (x, y) represents the horizontal and vertical coordinates; and G_σ stands for the Gaussian high-pass filter, and σ is the standard deviation (SD) of G, which controls the sharpening range.

Step 2: Add the unsharpened mask to the original image.

$$O(x,y) = I_i(x,y) + \lambda H(x,y) \tag{10}$$

where O is the final image after sharpening and λ is the scale coefficient that can control the strength of the sharpening.

Fig. 7 [46] shows the change in grayscale values after USM sharpening. The original edge is a side-plain edge. It is observed in Fig. 7 that there are two jumps enlarging the edge effects in the edge region. This phenomenon is defined as the overshoot artifact, which is a significant clue for the USM sharpening procedure. It is caused by the superposition of high-frequency signals.

In recent years, many researchers have proposed several approaches to detect USM sharpening [46–49]. The edge-modeling-based [47,48] and local-texture-based [46,49] approaches are two widely used techniques in USM sharpening detection. Cao et al. [47] first proposed the idea of USM sharpening detection. They discovered the histogram aberration caused by

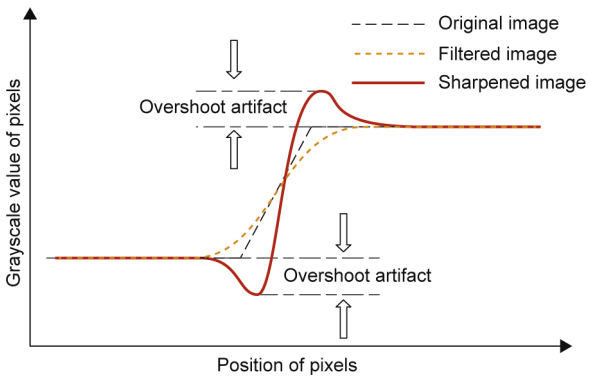

Fig. 7. Overshoot artifacts of USM sharpening [46].

sharpening, and built edge models to measure it. However, in their later work [48], the method in Ref. [47] was proved to be effective only when the image has a wide pixel value histogram. Cao et al. [48] proposed another method for USM sharpening detection. First, the edge position of the original image was detected. Next, the overshoot strength around the edge was measured, and the average overshoot strength over the whole image was calculated. Finally, a threshold was set for overshoot artifacts in order to determine whether an image has been USM sharpened. Although this approach in Ref. [48] solves some of the shortcomings of the method presented in Ref. [47], it is very sensitive to image noises.

Local-texture-based approaches apply different local textures in edge regions. Local binary pattern (LBP) and edge perpendicular binary coding (EPBC) [46] textures are two widely used local textures. Ding et al. [49] adopted LBP texture. They first located the edge region using a Canny operator. They then applied LBP texture around the edge pixel. The overall histogram of the whole image was calculated and an SVM classifier was utilized to differentiate the sharpened image from the original images. Experimental results showed that the LBP-based approach outperforms the edge modeling approaches. Later, Ding et al. [46] discovered that overshoot artifacts mainly appeared in the perpendicular direction of the edge. Therefore, they proposed a novel texture—EPBC texture—for USM sharpening detection. Compared with LBP texture, the EPBC texture feature uses a $1 \times N$ rectangle window, which is on the edge pixel and along the edge normal. A new binary coding strategy was applied for the rectangle window. Compared with other approaches, the EPBC-based approach has better detection accuracy with efficiency. This approach is also robust against JPEG compression and other noises, to some extent.

4.3. Detection of copy-move attacks

The aim of a copy-move attack is to deliberately forge or remove one or more objects in the source image. For object creation, the new object is generated by copying a source object and moving it to another position in the same image as shown in Fig. 8. For object removal, the object region being removed is replaced by some background patches in the same image. In order to deceive the human eye, fundamental affine transforms such as scaling, rotation, and so forth are usually applied before pasting.

Considering the mechanism of the copy-move attack, it can be concluded that in the tampered image, there is at least one pair of regions in the object or background region with extremely similar color, shape, and texture. By analyzing these similar image patch pairs, a copy-move attack can be detected. Accuracy and efficiency are two key issues in copy-move detection. Refs. [50,51] are the pioneering works in copy-move detection. Fridrich et al. [50] discussed several major requirements of the copy-move detection algorithm, including: ① allowing for an approximate match of small image segments; and ② having few false alarms and an acceptable processing time or complexity. They then proposed a detection algorithm based on block matching. The image investigated was first divided into a series of overlapping small blocks, and specific features were extracted from each block. By comparing the features, similar small blocks were detected; the test image was regarded as an image forgery when there were more than a preset number of block pairs gathering together in the spatial domain. Popescu and Farid [51] adopted principle component analysis (PCA) for feature extraction, and the eigenvalues of the small block were adopted as its corresponding feature. The major computation complexity of Refs. [50,51] is dominated by the lexicographic sorting, which is $O(F \times N \times \log N)$ [51], where F is the feature dimension and N is the number of pixels in the image. However, the above approaches may not achieve good results when the duplicated region is scaled or rotated greatly.

In order to deal with the problems caused by scaling and rotation, Bayram et al. [52] exploited the properties of the Fourier-Mellin transform, which is robust against translation, rotation, and scaling. Furthermore, they adopted the practice of counting bloom filters in order to reduce the computational complexity caused by lexicographic sorting. The experimental results showed that their features can resist a rotation of 10% and a scaling of 10%. Li and Yu [53] extended the idea in Ref. [52] and proposed the vector erosion filter to solve the problem of vector counters. Their algorithm is shown to be able to detect region duplication with a large rotation angle. Recently, Zandi et al. [54] proposed an adaptive copy-move forgery detection (CMFD) approach. Different thresholds were adopted for various image contents. The adaptive threshold of a specific block was determined by its SD; thus, the corresponding CMFD can detect duplications in both smooth and textured regions. Christlein et al. [55] performed a comprehensive evaluation of various kinds of CMFD approaches. An image database containing 48 base images was adopted, and the copy-move forgeries were carefully produced without leaving visually noticeable traces. The results in Ref. [55] demonstrated that although keypoint-based approaches are very efficient, the detection results will be affected by low-contrast regions and repetitive objects [55]. On the other hand, block-based methods provide high detection accuracies at the cost of higher computational complexity. Of all the features, Zernike features are the recommended choice.

4.4. Resampling detection

Generally speaking, in many image-splicing scenarios, the splicing image region usually undergoes a resizing and/or a rotation

(a) (b)

Fig. 8. A typical example of copy-move attack. (a) Original image; (b) tampered image.

procedure to cover the forgery traces and to make the final image forgery appear more realistic. Moreover, in the resizing and rotation procedure, the pixels in the modified image patch must be resampled to fit the new sample lattice. Hence, resampling detection can help to detect possible image forgeries and locate suspicious splicing regions.

In 2005, Popescu and Farid [56] proposed their pioneer work in resampling detection. They observed that for resampled signals, there are strong correlations between neighboring/adjacent samples. An EM algorithm was employed to estimate the underlying correlation parameters. With the above parameters, a probability map can be generated to describe the probability of a specific pixel to be correlated with its neighboring pixels. This approach can provide high detection accuracy for uncompressed images, although the detection performance will be degraded for JPEG compressed images because the block discrete cosine transform (BDCT) in JPEG compressed images leads to a specific kind of resampling that may confuse the detector.

Gallagher [57] proposed a simple algorithm to detect two widely used interpolation operators: the linear interpolation and the cubic interpolation. He observed that after linear/cubic interpolation, images would have a periodic pattern in the second derivative signal, and that the period of such a pattern can indicate the interpolation factor. The experiments showed that the approach in Ref. [57] can detect the zoom factor from 1.1 to 3.0 in increments of 0.1. Mahdian and Saic [58] extended this idea and showed that interpolation will introduce a specific periodicity when considering the covariance of the signal's derivatives. However, the detection performance of the above approaches will be degraded with medium or strong JPEG compressions.

In order to deal with JPEG compressed images, Kirchner and Gloe [59] modified the algorithm in Ref. [56] by removing the artifacts caused by JPEG compression to some extent. Their results demonstrated that the detection performance degrades greatly when the post-compression is stronger than the pre-compression. Moreover, upscaling is usually more detectable than downscaling.

Vázquez-Padín et al. [60] recently proposed a simple approach to detect upsampled images or image patches. The singular value decomposition (SVD) approach was adopted to characterize the linear dependencies of a resampled image. A specific measure describing the degrees of saturated pixels was then proposed in order to differentiate upsampled images from genuine images. This approach showed good performance in detecting small resampled patches.

4.5. Blind image-splicing detection

This kind of method attempts to detect various image-tampering and manipulation procedures based on the fact that such procedures will inevitably introduce certain artifacts that do not exist in the original images. Generally speaking, these approaches lead to a binary decision regarding whether a test image is original or not. However, even if the image is examined and determined to not be original, it is not possible to tell what kind of tampering has been performed, or what region has been tampered with. Nonetheless, blind image-tampering detection algorithms can be used as a preprocessing step for many image forensics systems in order to detect suspicious images without determining the editing/manipulation tools that were used.

One of the first blind image-tampering detection methods was proposed by Avcibas et al. [61], who aimed to detect image manipulations caused by affine transformations (scaling and rotation), brightness and contrast adjustments, and so forth. Various image quality metrics were adopted as image features, and linear regression was employed as the classifier. The forgery detection was formulated as a two-class classification problem, which allowed a feasible solution for most of the subsequent blind detection approaches. The detection accuracy of this method was reported as about 70%. Shi et al. [62] proposed an image-splicing detection method based on a DCT coefficient model. They observed that the distribution and correlation between neighboring BDCT coefficients are useful to differentiate natural images from spliced images. The Markov features extracted from the BDCT 2D arrays were adopted as the discriminative features, and the SVM was employed as the classifier. The experimental results showed that this approach can achieve over 90% detection accuracy on the Columbia Image Splicing Detection Evaluation Dataset [63]. Wang et al. [64] stated that the chromatic information is more discriminative than the grayscale information. They adopted the GLCM of the edge map in a chromatic channel as the discriminative feature, and employed SVM as the classifier. Similar to the Markov features in Ref. [62], the GLCM features represent the second-order statistics of the data and usually provide comparable results. He et al. [65] extended the idea from Ref. [62] and extracted Markov features in both the DCT and DWT domains. An SVM with recursive feature elimination (SVM-RFE) was employed to extract highly discriminative features and provide higher detection accuracy. This approach also achieved high detection accuracy in the Columbia Image Splicing Detection Evaluation Dataset [63]. Recently, Zhao et al. [66] proposed a new blind image-splicing detection method. In contrast to the traditional causal Markov features [62,64,65], they employed a 2D non-causal Markov model to describe the underlying characteristics of the natural and spliced images in both the DCT and DWT domains. The proposed non-causal model provided more information and better modeled the 2D image. The model parameters were regarded as the features, and SVM was adopted as the classifier. The experimental results revealed their detection accuracies to be above 90% in the Columbia Image Splicing Detection Evaluation Dataset, which demonstrated the feasibility of the blind detection approaches in image-splicing detection.

5. Conclusion and future work

Owing to various kinds of traces in image acquisition, image storage, and image editing stages, digital image forgeries can be detected without any prior information or knowledge. Among the various kinds of passive digital image forensics approaches, machine-learning-based techniques play an important role, in which forgery detection is formulated as a two-class (i.e., natural image or composite image) classification problem. Various features that can effectively disclose specific underlying traces have been proposed in digital image forensics.

The most critical issue and challenging problem in digital image forensics stem from the discriminative power and generalization ability of the extracted features. On the one hand, the extracted features should be sensitive to specific forgery operations and be robust against variations caused by different image contents. In most cases, these two requirements conflict with each other, leaving the question of how to design a highly discriminative, content-adaptive image forensics approach as an unsolved problem. On the other hand, most of the current features are handcrafted or hand-designed features that are usually based on certain assumptions or on a specific simplified model of the forgery procedure. However, due to the complexity in real images that are caused by variations in image content, image manipulation techniques, and so forth, handcrafted features have difficulty handling most forgery cases effectively and comprehensively.

With the development of sophisticated artificial intelligence techniques, deep learning suggests a promising solution for digital image forensics. In many deep learning structures, such as the CNN and the deep residual network, features are automatically learned

from training samples rather than being manually designed. With sufficient training samples, deep-learning-based approaches can provide a more comprehensive description for specific forgery operations, compared with traditional approaches. Recent deep-learning-based approaches have shown great performance improvement in digital image forensics applications such as MF detection. However, thus far, deep-learning-based approaches have not shown as much performance gain in digital image forensics as they have in image recognition and understanding. This is because the current network structures are being learned from structures that are adopted in image recognition; these structures are more or less related to image content, which leads to performance degradation in many digital image forensics applications. As a result, although deep-learning-based approaches are promising, they are not yet mature in digital image forensics; a considerable amount of work remains to be done in this area [67–71].

Acknowledgements

The work described in this paper was supported by National Key Research and Development Program of China (2016QY01W0104) and National Natural Science Foundation of China (61771310).

Compliance with ethics guidelines

Xiang Lin, Jian-Hua Li, Shi-Lin Wang, Alan-Wee-Chung Liew, Feng Cheng, and Xiao-Sa Huang declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Peraica A. Image science: Iconology, visual culture, and media aesthetics. Leonardo 2016;49(3):285.
[2] Farid H. A survey of image forgery detection. IEEE Signal Proc Mag 2009;26(2):16–25.
[3] Zhou G, Lv D. An overview of digital watermarking in image forensics. In: Proceedings of 2011 Fourth International Joint Conference on Computational Sciences and Optimization; 2011 Apr 15–19; Yunnan, China. Washington, DC: IEEE Computer Society; 2011. p. 332–5.
[4] Farid H. How to detect faked photos. Am Sci 2017;105(2):77–81.
[5] Birajdar GK, Mankar VH. Digital image forgery detection using passive techniques: A survey. Digital Invest 2013;10(3):226–45.
[6] Choi KS, Lam EY, Wong KKY. Source camera identification using footprints from lens aberration. In: Sampat N, DiCarlo JM, Martin RA, editors. Proceedings of SPIE—Electronic Imaging 2006: Digital Photography II; 2006 Jan 16–19; San Jose, CA, USA. Bellingham: International Society for Optics and Photonics; 2006. p. 172–9.
[7] Yerushalmy I, Hel-Or H. Digital image forgery detection based on lens and sensor aberration. Int J Comput Vis 2011;92(1):71–91.
[8] Lukas J, Fridrich J, Goljan M. Digital camera identification from sensor pattern noise. IEEE Trans Inf Foren Sec 2006;1(2):205–14.
[9] Kulkarni N, Mane V. Improvements on sensor noise based on source camera identification using GLCM. In: Proceedings of International Conference on Advances in Science and Technology; 2014 Oct 29–31; Ota, Nigeria. New York: International Journal of Computer Applications; 2015. p. 1–4.
[10] Sandoval Orozco AL, Arenas González DM, Rosales Corripio J, García Villalba LJ, Hernandez-Castro JC. Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections. Computing 2014;96(9):829–41.
[11] Fridrich J. Digital image forensics using sensor noise. IEEE Signal Proc Mag 2009;26(2):26–37.
[12] Gao S, Xu G, Hu RM. Camera model identification based on the characteristic of CFA and interpolation. In: IWDW'11 Proceedings of the 10th International Conference on Digital-Forensics and Watermarking; 2011 Oct 23–26; Atlantic City, NJ, USA. Berlin: Springer-Verlag; 2012. p. 268–80.
[13] Prasad P. Image forgery localization via CFA based feature extraction and Poisson matting. Int J Sci Res 2014;3(10):1273–8.
[14] Katre Y, Chandel GS. Image forgery detection using analysis of CFA artifacts. Int J Adv Technol Eng Sci 2014;2(1):381–9.
[15] Lukas J, Fridrich J. Estimation of primary quantization matrix in double compressed JPEG images. In: Proceedings of Digital Forensic Research Workshop; 2003 Aug 5–8; Cleveland, OH, USA. p. 5–8.
[16] Fu D, Shi YQ, Su W. A generalized Benford's law for JPEG coefficients and its applications in image forensics. In: Delp EJ, Wong PW, editors. Proceedings of SPIE—Electronic Imaging 2007: Security, Steganography, and Watermarking of Multimedia Contents IX; 2007 Jan 28–Feb 1; San Jose, CA, USA. Bellingham: International Society for Optics and Photonics; 2007. 65051L1–11.
[17] Pevny T, Fridrich J. Detection of double-compression in JPEG images for applications in steganography. IEEE Trans Inf Foren Sec 2008;3(2):247–58.
[18] Farid H. Exposing digital forgeries from JPEG ghosts. IEEE Trans Inf Foren Sec 2009;4(1):154–60.
[19] Lin Z, He J, Tang X, Tang CK. Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis. Pattern Recognit 2009;42(11):2492–501.
[20] Huang F, Huang J, Shi YQ. Detecting double JPEG compression with the same quantization matrix. IEEE Trans Inf Foren Sec 2010;5(4):848–56.
[21] Yang J, Xie J, Zhu G, Kwong S, Shi YQ. An effective method for detecting double JPEG compression with the same quantization matrix. IEEE Trans Inf Foren Sec 2014;9(11):1933–42.
[22] Luo W, Qu Z, Huang J, Qiu G. A novel method for detecting cropped and recompressed image block. In: Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing; 2007 Apr 15–20; Honolulu, HI, USA. Piscataway: IEEE; 2007. p. 217–20.
[23] Chen YL, Hsu CT. Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection. IEEE Trans Inf Foren Sec 2011;6(2):396–406.
[24] Qu Z, Luo W, Huang J. A convolutive mixing model for shifted double JPEG compression with application to passive image authentication. In: Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing; 2008 Mar 30–Apr 4; Las Vegas, NV, USA. Piscataway: IEEE; 2008. p. 1661–4.
[25] Bianchi T, Piva A. Detection of nonaligned double JPEG compression based on integer periodicity maps. IEEE Trans Inf Foren Sec 2012;7(2):842–8.
[26] Bianchi T, Piva A. Image forgery localization via block-grained analysis of JPEG artifacts. IEEE Trans Inf Foren Sec 2012;7(3):1003–17.
[27] Wang SL, Liew AWC, Li SH, Zhang YJ, Li JH. Detection of shifted double JPEG compression by an adaptive DCT coefficient model. EURASIP J Adv Signal Process 2014;2014:101.
[28] Johnson MK, Farid H. Exposing digital forgeries by detecting inconsistencies in lighting. In: Proceedings of the 7th Workshop on Multimedia and Security; 2005 Aug 1–2; New York, NY, USA. New York: ACM Press; 2005. p. 1–10.
[29] Johnson MK, Farid H. Exposing digital forgeries through specular highlights on the eye. In: Proceedings of the 9th International Conference on Information Hiding; 2007 Jun 11–13; Saint Malo, France. Berlin: Springer-Verlag; 2007. p. 311–25.
[30] Johnson MK, Farid H. Exposing digital forgeries in complex lighting environments. IEEE Trans Inf Foren Sec 2007;2(3):450–61.
[31] Kee E, Farid H. Exposing digital forgeries from 3-D lighting environments. In: Proceedings of 2010 IEEE International Workshop on Information Forensics and Security; 2010 Dec 12–15; Seattle, WA, USA. Piscataway: IEEE; 2010. p. 1–6.
[32] Nillius P, Eklundh JO. Automatic estimation of the projected light source direction. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2001 Dec 8–14; Kauai, HI, USA. Piscataway: IEEE; 2001. p. 1076–83.
[33] Koenderink JJ, van Doorn AJ, Pont SC. Light direction from shad(ow)ed random Gaussian surfaces. Perception 2004;33(12):1405–20.
[34] Zhang W, Cao X, Zhang J, Zhu J, Wang P. Detecting photographic composites using shadows. In: Proceedings of 2009 IEEE International Conference on Multimedia and Expo; 2009 Jun 28–Jul 3; New York, NY, USA. Piscataway: IEEE; 2009. p. 1042–5.
[35] Fan W, Wang K, Cayre F, Xiong Z. 3D lighting-based image forgery detection using shape-from-shading. In: Proceedings of the 20th European Signal Processing Conference; 2012 Aug 27–31; Bucharest, Romania. Piscataway: IEEE; 2012. p. 1777–81.
[36] Bovik AC, Huang TS, Munson DC. The effect of median filtering on edge estimation and detection. IEEE Trans Pattern Anal Mach Intell 1987;9(2):181–94.
[37] Bovik AC. Streaking in median filtered images. IEEE Trans Acoust Speech Signal Process 1987;35(4):493–503.
[38] Kirchner M, Fridrich J. On detection of median filtering in digital images. In: Proceedings of SPIE—Electronic Imaging 2010: Media Forensics and Security II; 2010 Jan 17–21; San Jose, CA, USA. Bellingham: International Society for Optics and Photonics; 2010. p. 7541101–12.
[39] Cao G, Zhao Y, Ni R, Yu L, Tian H. Forensic detection of median filtering in digital images. In: Proceedings of 2010 IEEE International Conference on Multimedia and Expo; 2010 Jul 19–23; Singapore, Singapore. Piscataway: IEEE; 2010. p. 89–94.
[40] Yuan HD. Blind forensics of median filtering in digital images. IEEE Trans Inf Foren Sec 2011;6(4):1335–45.
[41] Chen C, Ni J. Median filtering detection using edge based prediction matrix. In: Shi YQ, Kim HJ, Pérez-González F, editors Proceeding of 10th International Workshop on Digital Forensics and Watermarking; 2011 Oct 23–26; Atlantic City, NJ, USA. Berlin: Springer-Verlag; 2012. p. 361–75.
[42] Kang X, Stamm MC, Peng A, Ray Liu KJ. Robust median filtering forensics using an autoregressive model. IEEE Trans Inf Foren Sec 2013;8(9):1456–68.
[43] Chen C, Ni J, Huang J. Blind detection of median filtering in digital images: A difference domain based approach. IEEE Trans Image Process 2013;22(12):4699–710.

[44] Zhang Y, Li S, Wang S, Shi YQ. Revealing the traces of median filtering using high-order local ternary patterns. IEEE Signal Proc Lett 2014;21(3):275–9.
[45] Chen J, Kang X, Liu Y, Jane Wang Z. Median filtering forensics based on convolutional neural networks. IEEE Signal Proc Lett 2015;22(11):1849–53.
[46] Ding F, Zhu G, Yang J, Xie J, Shi YQ. Edge perpendicular binary coding for USM sharpening detection. IEEE Signal Proc Lett 2015;22(3):327–31.
[47] Cao G, Zhao Y, Ni R. Detection of image sharpening based on histogram aberration and ringing artifacts. In: Proceedings of the 2009 IEEE International Conference on Multimedia and Expo; 2009 Jun 28–Jul 3; New York, NY, USA. Piscataway: IEEE; 2009. p. 1026–9.
[48] Cao G, Zhao Y, Ni R, Kot AC. Unsharp masking sharpening detection via overshoot artifacts analysis. IEEE Signal Proc Lett 2011;18(10):603–6.
[49] Ding F, Zhu G, Shi YQ. A novel method for detecting image sharpening based on local binary pattern. In: Shi Y, Kim HJ, Pérez-González F, editors. Proceedings of 12th International Workshop on Digital Forensics and Watermarking; 2013 Oct 1–4; Auckland, New Zealand. Berlin: Springer-Verlag; 2013. p. 180–91.
[50] Fridrich AJ, Soukal BD, Lukas AJ. Detection of copy-move forgery in digital images. Int J 2003;3(2):652–63.
[51] Popescu AC, Farid H. Exposing digital forgeries by detecting duplicated image regions. Technical report. Hanover: Department of Computer Science, Dartmouth College; 2004. Report No.: TR2004-515.
[52] Bayram S, Sencar HT, Memon N. An efficient and robust method for detecting copy-move forgery. In: Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing; 2009 Apr 19–24; Taipei, Taiwan, China. Washington, DC: IEEE Computer Society; 2009. p. 1053–6.
[53] Li W, Yu N. Rotation robust detection of copy-move forgery. In: Proceedings of 2010 IEEE International Conference on Image Processing; 2010 Sep 26–29; Hong Kong, China. Piscataway: IEEE; 2010. p. 2113–6.
[54] Zandi M, Mahmoudi-Aznaveh A, Mansouri A. Adaptive matching for copy-move Forgery detection. In: Proceedings of 2014 IEEE International Workshop on Information Forensics and Security; 2014 Dec 3–5; Atlanta, GA, USA. Piscataway: IEEE; 2014. p. 119–24.
[55] Christlein V, Riess C, Jordan J, Riess C, Angelopoulou E. An evaluation of popular copy-move forgery detection approaches. IEEE Trans Inf Foren Sec 2012;7(6):1841–54.
[56] Popescu AC, Farid H. Exposing digital forgeries by detecting traces of resampling. IEEE Trans Signal Process 2005;53(2):758–67.
[57] Gallagher AC. Detection of linear and cubic interpolation in JPEG compressed images. In: Proceedings of the 2nd Canadian Conference on Computer and Robot Vision; 2005 May 9–11; Victoria, BC, Canada. Washington, DC: IEEE Computer Society; 2005. p. 65–72.
[58] Mahdian B, Saic S. Blind authentication using periodic properties of interpolation. IEEE Trans Inf Foren Sec 2008;3(3):529–38.
[59] Kirchner M, Gloe T. On resampling detection in re-compressed images. In: Proceedings of the 1st IEEE International Workshop on Information Forensics and Security; 2009 Dec 6–9; London, UK. Piscataway: IEEE; 2009. p. 21–5.
[60] Vázquez-Padín D, Comesana P, Pérez-González F. An SVD approach to forensic image resampling detection. In: Proceedings of the 23rd European Signal Processing Conference; 2015 Aug 31–Sep 4; Nice, France. Piscataway: IEEE; 2015. p. 2112–6.
[61] Avcibas I, Bayram S, Memon N, Ramkumar M, Sankur B. A classifier design for detecting image manipulations. In: Proceedings of the International Conference on Image Processing; 2004 Oct 24–27; Singapore, Singapore. Piscataway: IEEE; 2004. p. 2645–8.
[62] Shi YQ, Chen C, Chen W. A natural image model approach to splicing detection. In: Proceedings of the 9th Workshop on Multimedia and Security; 2007 Sep 20–21; Dallas, TX, USA. New York: ACM Press; 2007. p. 51–62.
[63] Ng TT, Hsu J, Chang SF. Columbia image splicing detection evaluation dataset [Internet]. Available from: http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataSet/dlform.html.
[64] Wang W, Dong J, Tan T. Effective image splicing detection based on image chroma. In: Proceedings of the 16th IEEE International Conference on Image Processing; 2009 Nov 7–10; Cairo, Egypt. Piscataway: IEEE; 2009. p. 1257–60.
[65] He Z, Lu W, Sun W, Huang J. Digital image splicing detection based on Markov features in DCT and DWT domain. Pattern Recognit 2012;45(12):4292–9.
[66] Zhao X, Wang S, Li S, Li J. Passive image-splicing detection by a 2-D noncausal Markov model. IEEE Trans Circuits Syst Video Techn 2015;25(2):185–99.
[67] Bayar B, Stamm MC. A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security; 2016 Jun 20–22; Vigo, Spain. New York: ACM Press; 2016. p. 5–10.
[68] Bondi L, Güera D, Baroffio L, Bestagini P, Delp EJ, Tubaro S. A preliminary study on convolutional neural networks for camera model identification. In: Proceedings of IS&T International Symposium on Electronic Imaging: Media Watermarking, Security, and Forensics; 2017 Jan 29–Feb 2; San Francisco, CA, USA. Washington, DC: Society for Imaging Science and Technology; 2017. p. 67–76.
[69] Bayar B, Stamm MC. On the robustness of constrained convolutional neural networks to JPEG post-compression for image resampling detection. In: Proceedings of 2017 IEEE International Conference on Acoustics, Speech and Signal Processing; 2017 Mar 5–9; New Orleans, LA, USA. Piscataway: IEEE; 2017. p. 2152–6.
[70] Chen J, Kang X, Liu Y, Wang ZJ. Median filtering forensics based on convolutional neural networks. IEEE Signal Proc Lett 2015;22(11):1849–53.
[71] Liu Y, Guan Q, Zhao X, Cao Y. Image forgery localization based on multi-scale convolutional neural networks. 2017. arXiv: 1706.07842v3.