



PIZZARO: Forensic analysis and restoration of image and video data



Jan Kamenicky^a, Michal Bartos^a, Jan Flusser^a, Babak Mahdian^a, Jan Kotera^a, Adam Novozamsky^a, Stanislav Saic^a, Filip Sroubek^a, Michal Sorel^a, Ales Zita^a, Barbara Zitova^{a,*}, Zdenek Sima^b, Petr Svarc^b, Jan Horinek^c

^aInstitute of Information Theory and Automation, The Czech Academy of Sciences, Pod vodárenskou veží 4, Prague, Czech Republic

^bInstitute of Criminalistics, Audio-Video Department, Prague, Czech Republic

^cNational Drug Headquarters of the Criminal Police and Investigation Service of the Police of the Czech Republic, Informatics Department, Prague, Czech Republic

ARTICLE INFO

Article history:

Available online 28 April 2016

Keywords:

Image forensic analysis
Image restoration
Image tampering detection
Image source identification

ABSTRACT

This paper introduces a set of methods for image and video forensic analysis. They were designed to help to assess image and video credibility and origin and to restore and increase image quality by diminishing unwanted blur, noise, and other possible artifacts. The motivation came from the best practices used in the criminal investigation utilizing images and/or videos. The determination of the image source, the verification of the image content, and image restoration were identified as the most important issues of which automation can facilitate criminalists work. Novel theoretical results complemented with existing approaches (LCD re-capture detection and denoising) were implemented in the PIZZARO software tool, which consists of the image processing functionality as well as of reporting and archiving functions to ensure the repeatability of image analysis procedures and thus fulfills formal aspects of the image/video analysis work. Comparison of new proposed methods with the state of the art approaches is shown. Real use cases are presented, which illustrate the functionality of the developed methods and demonstrate their applicability in different situations. The use cases as well as the method design were solved in tight cooperation of scientists from the Institute of Criminalistics, National Drug Headquarters of the Criminal Police and Investigation Service of the Police of the Czech Republic, and image processing experts from the Czech Academy of Sciences.

© 2016 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

In our world, digital images are an extensively used medium of communication that compactly and efficiently conveys a huge amount of information about our surrounding. However two important questions should be considered – how much we can trust all these photographs which are not necessarily obtained from a trustworthy source and do we notice everything contained in images, which are often acquired under non-ideal, inadequate conditions and thus possibly blurred, noisy, or with other types of unwanted artifacts.

Our research addressed two related topics – the quality of image data with respect to their credibility and origin, and image restoration aimed at diminishing unwanted blur, noise, and other possible artifacts. The motivation came from the best practices used in the criminal investigation utilizing images and/or videos. The *determination of the image source*, the *verification of the image content*, and possible *image restoration* were identified as important issues, of which automation can notably facilitate criminalist's work (see Fig. 1). Situations, when the origin of images (child pornography) or their authenticity (insurance frauds) is questioned, occur with increasing frequency nowadays.

Regarding the image source determination, the ability to link individual pictures to appropriate acquisition devices (camera, scanner, cell phone, etc.) can help investigators for example to indicate a particular camera type, or to detect cases, when a digital photo has been re-captured from an LCD screen. For the particular camera unit identification and model specification we have applied approaches based on photo response non uniformity (PRNU) and JPEG quantization tables, respectively. PRNU is known to be able to describe pixel sensitivity to light and is present in the sensor

* Corresponding author.

E-mail addresses: kamenik@utia.cas.cz (J. Kamenicky), bartos@utia.cas.cz (M. Bartos), flusser@utia.cas.cz (J. Flusser), mahdian@utia.cas.cz (B. Mahdian), kotera@utia.cas.cz (J. Kotera), novozamsky@utia.cas.cz (A. Novozamsky), ssaic@utia.cas.cz (S. Saic), sroubekf@utia.cas.cz (F. Sroubek), sorel@utia.cas.cz (M. Sorel), zita@utia.cas.cz (A. Zita), zitova@utia.cas.cz (B. Zitova), zdenek.sima1@pcr.cz (Z. Sima), pizzaro@utia.cas.cz (P. Svarc), jan.horinek@seznam.cz (J. Horinek).

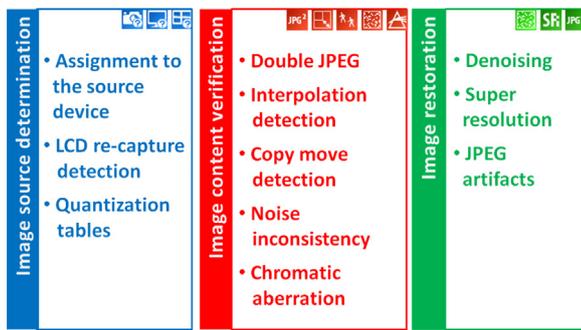


Fig. 1. An overview of the developed and implemented tools for image forensic analysis in PIZZARO.

pattern noise. The detection of LCD screen acquisition is done using structural and color abnormalities in recaptured images. This task is challenging due to advances in image capture technologies. It is possible to re-capture high quality images in such a way they are hard to be distinguished from the originals by an untrained eye.

Various approaches developed for detection of tampering and forgery traces in image content range from a simple metadata analysis to a complex shadow and scene lighting analysis using mathematical models. The simple metadata analysis often provides only a weak indicator of forgery and usually is not applicable in real cases. On the other hand, the manual scene analysis exploiting knowledge of scene lighting and/or geometry parameters creates strong forgery indicators but usually requires complex tuning for every case by professionals with deep expertise in the field. We have provided a set of methods that can help investigators in their decision making. These methods belong to the family of passive (blind) image forensic methods, which does not require any prior information about images under verification.

Specifically, we focused on methods for detecting double compression, interpolation, copy–paste action, and noise and chromatic aberration inconsistencies. Double compression is introduced to an image while altering the JPEG image in a photo-editing software, when the image is decompressed and after the manipulations are carried out the image is re-saved and thus compressed again. Interpolation, required when the image is geometrically transformed (i.e., rotation, scaling, skewing), is often applied on the image to be able to merge two or more images together to create high quality and consistent image forgeries. In copy–move forgeries, a part of the image is copied and pasted into another part of the same image, with the intention to hide an object or a region of the image. Methods for copy–move forgery detection search for similar overlapping image blocks, with possibly blurred borders (to achieve smooth transition between the original image and pasted blocks). Noise inconsistencies are often introduced while concealing the tampering traces, when locally random noise is added to the forged image regions. Finally, inconsistencies in chromatic aberrations across the image can indicate an image forgery, too. The chromatic aberrations are introduced to the image due to inability of a camera lens to focus all colors to the same convergence point. It is manifested as a regular pattern of slight blur and/or color shifts. Irregularities in this pattern could imply image tampering.

Finally, the last part of our work is related to improving image quality by reducing noise and blur. A paradigm for forensic specialists examining images and videos from surveillance cameras is to deal with cases that often exhibit insufficient resolution and compression artifacts, which prevents correct identification of subjects, such as human faces or car license plates. Moreover, omnipresent image noise is a frequent image degradation, typically visible in images taken in low-light conditions. We design a methodology for removing noise and compression

artifacts and even increasing data resolution, if applicable (e.g., several frames from a video sequence are available).

Proposed work is based on tight cooperation of scientists from the Institute of Criminalistics, National Drug Headquarters of the Criminal Police and Investigation Service of the Police of the Czech Republic, and image processing experts from the Czech Academy of Sciences. All novel theoretical results together with existing methods for LCD image re-capture and denoising, were implemented in the PIZZARO software [1] to provide required tools. In addition to the image processing functionality, we also paid attention to formal aspects of the image/video analysis, which play important role in the forensic practice. The PIZZARO software enables to create log files and reports to ensure the repeatability of image analysis procedures, when all taken steps with parameter settings and achieved outputs are recorded for future use/verification together with the operator identity and archiving information such as the file paths and dates.

After the review of other existing software solutions in Section 2 we will introduce methods proposed for solving the above mentioned three issues – source device analysis in Section 3, image content verification in Section 4, and image restoration in Section 5. Section 6 introduces main features of the resulting software package PIZZARO (<http://pizzaro.utia.cas.cz>), implementing proposed algorithms. Illustrative use cases with examples of the PIZZARO application are described in Section 7 and Section 8 concludes the paper.

2. Existing software solutions

Numerous software packages have been introduced in recent years for verifying the integrity of digital images. Some of them utilized sophisticated theoretical and scientific approaches and some others rather focused on practical and ad-hoc methods. In general, Forensic Toolkit (FTK) [2] has become the most popular solution in the digital forensic field and it has been widely used by digital forensic investigators. Fourandsix [3] has introduced an image forensic tool providing image authentication functionalities by using JPEG signatures. This software is mainly oriented on media publishers and forensic investigators. Verifeyed [4] has developed software focused on detecting manipulated photos and PDF files for insurance and corporate security markets. Belkasoft [5] introduced forensic software that enables to analyze the validity of digital images by using a set of separate basic image forensic methods. They also focused on insurance market. Forensic Pathways [6] has developed and introduced to the market an image ballistics software that aims to meet main needs of forensic investigators in the field. Last, but not least, we have to mention the Image Error Level Analyser [7] which is a free and easily accessible image forensic tool that has become very popular among the community despite its limitations and high false positive rates.

Also the area of image and video restoration and enhancement is experiencing a lot of new software packages. For example, we can mention Amped Five [8] (started as MIPE [9]), that developed an image and video enhancement (super-resolution) software. This software also provides image forensic and ballistics tools targeting the digital forensic experts. Another tool that can be mentioned is MotionDSP [10] that provides real-time super-resolution and video enhancement techniques. They also deliver GPU accelerated versions of their software. Cognitech [11] provides methods enhancing quality and resolution of (mainly facial) videos for the forensic market. Another software package providing super resolution, denoising, and deblurring has been developed by RTCVision [12]. Impress [13] is another company providing video enhancing tools accelerated by utilizing GPUs. The authors of the webpage [14] provide several alternative SR toolboxes.

Despite the fact that there are numerous image and video forensic and enhancement software packages available on market, we can note that none of them provides complete solution.

Inspecting and enhancing digital images and videos is complex process and often a combination of different methods and solutions is needed to derive meaningful and desired output.

3. Image source determination

Methods assigning a digital image to the source camera are typically based on the fact that image sensors suffer from several fundamental and technology related imperfections resulting in their performance limitations and noise. For example, if we take a picture of an absolutely evenly lit scene, the resulting digital image usually still exhibits small changes in intensity among individual pixels which is partly because of pattern noise, readout noise or shot noise. While readout noise or shot noise are random components, the pattern noise is deterministic and remain approximately the same if multiple pictures of the same scene are taken. As a result, pattern noise can be used as the sensor fingerprint of cameras.

Alternatively, if the digital image is the result of LCD recapturing (i.e., a photo displayed on an LCD is recaptured by camera), often we can observe periodic detectable artifacts in the recaptured photo. The last method for analysis of the image acquisition device is based on the evaluation of the camera and image attributes related to their resolution and the way how they were preprocessed.

3.1. Assignment to the source device

It has been shown that photo response non uniformity (PRNU, describing the pixel sensitivity to light) has a dominant presence in the sensor pattern noise [15,16] and can be utilized as the sensor fingerprint because of its stability and discrimination power. Most of the state-of-the-art source identification methods are based on the method proposed by Jessica Fridrich et al. (e.g., [15,17]). Here, the image acquisition process is modeled in the following way:

$$g = u + u \cdot \Gamma + Y, \tag{1}$$

where g denotes the image produced by the camera, u denotes the noise-free image (the perfect image of the scene), Γ denotes the PRNU noise and Y stands for all other additive or negligible noise components. We have proposed a modification of the standard approach, addressing the problems with the time demanding PRNU estimation and with the situation when the images are taken with various zoom.

For a given camera, PRNU noise can be estimated by averaging multiple images $g_i, i = 1, \dots, N$ captured by this camera. Prior to averaging, the scene content is suppressed from the image. This preprocessing reduces the required number of images that is needed to achieve a robust PRNU estimation. The suppression is realized by application of a denoising filter \mathcal{F} and averaging the noise residuals instead. We will denote these residuals by \hat{g}_i (i.e., $\hat{g}_i = g_i - \mathcal{F}(g_i)$). Hence, the deterministic components of the camera sensor noise are computed in the following way:

$$\Gamma_{\text{sensor}} = \frac{1}{N} \sum_{i=1}^N \hat{g}_i = \frac{1}{N} \sum_{i=1}^N (g_i - \mathcal{F}(g_i)). \tag{2}$$

Alternatively, maximum likelihood estimation (MLE) instead of simple averaging is often employed.

To reduce the false positive rate, sensor fingerprints are enhanced by Wiener filtering in the frequency domain (e.g., to reduce JPEG compression artifacts) as well as by using a linear pattern removal process through zero-mean operation (e.g., to remove traces of color filter array – CFA – interpolation) [16]. To summarize, Γ_{sensor} is denoted as the basic version of the sensor fingerprint of the camera.

The assignment test of a digital image to the possible source camera has been carried out by evaluating the similarity measure of two sensor fingerprints. One is obtained from the image under investigation and second from the set of camera reference images. Having available two different sensor fingerprints Γ_{s_1} and Γ_{s_2} , we can measure their similarity by employing a normalized correlation:

$$\text{corr}(\Gamma_{s_1}, \Gamma_{s_2}) = \frac{\langle (\Gamma_{s_1} - \bar{\Gamma}_{s_1}), (\Gamma_{s_2} - \bar{\Gamma}_{s_2}) \rangle}{\|\Gamma_{s_1} - \bar{\Gamma}_{s_1}\| \|\Gamma_{s_2} - \bar{\Gamma}_{s_2}\|}, \tag{3}$$

where \bar{X} denotes mean of the vector X , $\langle \cdot \rangle$ denotes inner product and $\|\cdot\|$ denotes L_2 norm. If the calculated correlation is above a certain threshold, we accept the hypothesis that the tested image has been acquired by the camera.

The implemented method differs from the one described in [15,18]. It only uses the central portions of images which leads to more robust results when images are strongly impacted by vignetting. The vignetting is typically more profound at non-central image parts thus filtering out outer parts of images increases the probability of achieving higher true positive (TP) rate.

In Fig. 2 we demonstrate on 43 images taken with different optical zoom how the correlation values increase if the central parts (bottom graph) are used instead of the full-size images (upper graph). The two histograms in Fig. 3 show that the correlations of the reference fingerprint with non-corresponding cameras are below the values with the corresponding camera even when the central part approach was applied, so the cropping does not cause increase in the number of false positives. The minimum correlation of the

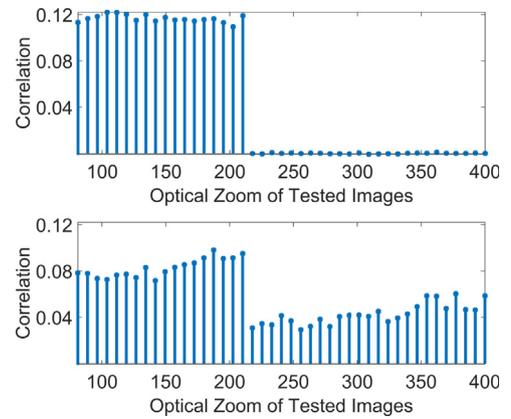


Fig. 2. Correlation values for images taken by the reference camera but with an increasing zoom; the Γ_{sensor} was created from images taken with the zoom 50. Top: using the full-size image approach; bottom: using the central part approach.

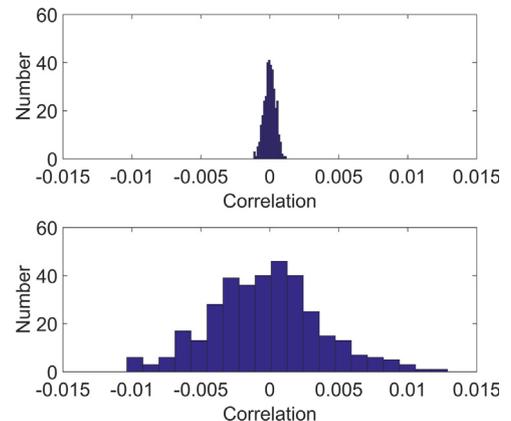


Fig. 3. Histogram of the correlation values from the false positive rate testing (10 different cameras). Top: using the full-size image approach; bottom: using the central part approach.

true test (Fig. 2 bottom) is more than 2 times higher than the maximum of false test in Fig. 3 bottom. Moreover, the central portion approach significantly reduce the computational time.

To increase the robustness of the correct identification we recommend to create several Γ_{sensor} with different zoom. It eliminates the need for the central part. Our software includes this feature that multiple Γ_{sensor} can be associated with a single camera.

3.2. LCD image re-capture

In order to be able to identify images which were created by re-capturing a photograph from a LCD screen, we have implemented three types of features which are able to reflect the abnormalities of recaptured image [19]. They are used together with a pre-trained support vector machine (SVM) classifier. Design of the feature set is focused on detection of certain image aberrations occurring during the re-capture process.

On LCD-originating images periodic texture pattern can be often observed. This is due to the aggregation of regular structures of LCD. Even though this pattern can be partially avoided by marginal image blurring or by higher quality data acquisition, related loss of fine details is inevitable and we exploit this fact in the first two types of features. The third type of features is based on color characteristic. The color space of the resulting recaptured image is usually less saturated comparing to the original due to the generally smaller color gamut of LCD screens than a typical camera.

The set of implemented features consists of:

- **LBP features** – 80 features are evaluated using the local binary patterns (LBP) operator $LBP_{P,R}^{\text{riu}2}$ with $P = \{8, 16, 24, 24\}$ and $R = \{1, 2, 3, 4\}$, respectively [20]. It is defined by means of a set of P neighbor pixels for the given pixel, equally distributed on a circle of radius R (see Fig. 4). To achieve gray scale and contrast invariance the operator is defined as

$$LBP_{P,R} = \sum_{p=0}^{P-1} H(g(p) - g(c)) 2^p, \quad (4)$$

where $g(c)$ is the intensity of the central pixel, $g(p)$ is the intensity of the p th neighbor and $H(\cdot)$ is Heaviside step function. Note, that by using 2^p binomial factor, each spatial structure is assigned unique $LBP_{P,R}$ number. To achieve rotation invariance:

$$LBP_{P,R}^{\text{riu}2} = \begin{cases} \sum_{p=0}^{P-1} H(g(p) - g(c)) & U_{P,R} \leq 2 \\ P + 1 & \text{otherwise,} \end{cases} \quad (5)$$

where $U_{P,R}$ is the uniformity operator, which corresponds to number of $LBP_{P,R}$ spatial transitions $U_{P,R} = |H(g(P-1) - g(c)) - H(g(0) - g(c))| + \sum_{p=1}^{P-1} |H(g(p) - g(c)) - H(g(p-1) - g(c))|$.

In this way, only the most uniform neighbor sets are taken into consideration.

- **Multi-scale wavelet statistics** – the first two geometric moments (the mean and the variance) of all the high-pass bands from the Haar wavelet decomposition up to the level 3, computed separately for R, G, and B channels – 54 features in total.
- **Color features** – Average pixel value of each color channel; RG, RB, and BG correlations between color channels; center of mass of histogram $h(i)$ of each color channel, where $h(i)$ is the number of pixels with intensities $i - 1$ or $i + 1$; RGB pairs energy ratios $E_1 = |G|^2/|B|^2$, $E_2 = |G|^2/|R|^2$, $E_3 = |B|^2/|R|^2$ [21]; mean, standard deviation and skewness, computed for the H, S, and V color channels, respectively [22].

The implemented SVM classifier is trained using the described features on a set of original images with their recaptured counterparts and then used for verification. We have used a database of 12,000 images, half of which were recaptured from numerous LCD/camera combinations. 2000 images were manually recaptured from our original photographs, whereas the rest were general images taken from Internet sources.

The recommended usage of the proposed approach starts with the classifier training using the samples from camera/LCD set in question. As is apparent from practice the information about the camera and LCD devices (e.g., devices owned by forgery suspect) is often available during the investigation.

3.3. Quantization tables

The last method aims to identify the set of camera models possibly used for the acquisition of the analyzed image. The proposed method is based on extraction of a certain set of features from the digital image file and matching them to a set of camera models. For instance, having a digital image of resolution of 1000×800 pixels and a claim that the digital image has been captured by a particular camera model (camera model name is stored in digital image metadata) and not been modified since, we can check if that particular camera device can produce digital images with such resolution. If we know that the camera produces only digital images with resolutions 2592×1944 , 1600×1200 , and 640×480 , we obviously can draw a conclusion that the above claim is false and the digital image was not produced by this camera. Otherwise, the tested model can be a potential source of the image.

The above example illustrates how the image resolution can be employed as a feature. Numerous similar features can be used for the image-camera matching process [23]. More formally, we say that a digital image has attributes, and the image metadata are their respective values, which characterize the digital image. Essentially, some attribute values are dependent on the camera with which the digital image has been taken. We refer to such camera associated features stored in the digital image as camera fingerprints. The properties that can explicitly characterize an acquisition device (camera) include, e.g., its producer and model,

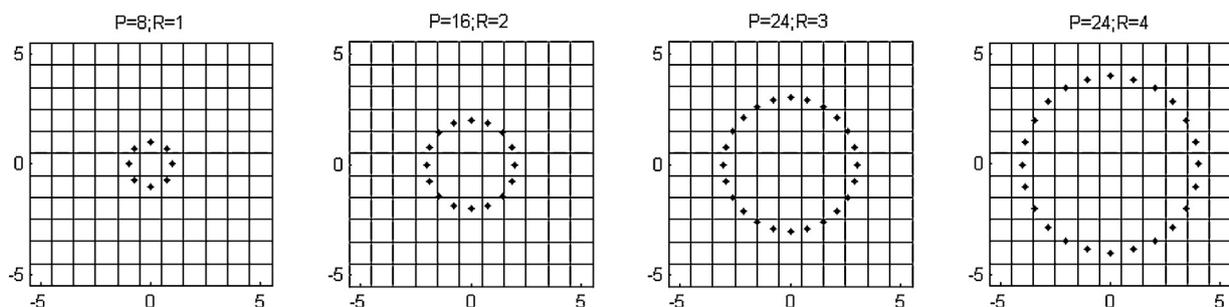


Fig. 4. Circularly symmetric neighbor sets for $LBP_{8,1}^{\text{riu}2}$, $LBP_{16,2}^{\text{riu}2}$, $LBP_{24,3}^{\text{riu}2}$ and $LBP_{24,4}^{\text{riu}2}$.

the output file format, imaging sensor properties, the digital zoom interpolation method, the color filter array interpolation method used to encode an image, etc. Some of these properties identify a camera uniquely, and some of them can be considered as camera model fingerprints. In the provided method, quantization tables (QTs), which encode digital images to JPEG format are used to match photos to their potential source camera models.

We assume a fixed tuple of properties sufficient for unique identification of any camera. We denote such a tuple by $c\vec{m}$, a camera ID vector. Next, we assume a tuple of camera attributes, whose values pose suitable fingerprints of available cameras in our database. We denote such a tuple of fingerprints by $\vec{\theta}$, a fingerprint vector.

Specifically, given a “testing” tuple $(c\vec{m}_0, \vec{\theta})$, our default position is that $\vec{\theta}$ cannot be a real fingerprint vector of $c\vec{m}_0$. Accordingly, we set out the following null hypothesis:

$$H_0: \text{“}\vec{\theta}\text{ cannot be a fingerprint of } c\vec{m}_0\text{”}. \quad (6)$$

To accept or reject this hypothesis, we downloaded over one million digital images from a typical photo sharing site and extracted a reference fingerprint data set. To discard non-original (i.e., manipulated) images and create a reliable reference data set, photos containing obvious traces of modifications were eliminated. To further eliminate non-original images, only those that form sufficiently big clusters of images with the same paired producer model and their QTs are retained and employed to accept or reject the null hypothesis – to identify a set of potential source camera models for given tested image. The classification is based on the luminance QTs only and a threshold-based test replaced originally used statistical testing.

4. Image content verification

Digital forensic methods for detection of forgery traces can be classified into two main categories – data hiding approaches and methods working with digital signatures. By data hiding we refer to methods embedding secondary data into the image such as digital watermarking approaches do [24]. Digital watermarking assumes an inserting of a digital watermark at the source side (e.g., camera) and verifying the mark integrity at the detection side. Watermarks are mostly inseparable from the digital image they are embedded in and they undergo the same transformations as the image itself. The latter is their main advantage – it is difficult to overcome them – but at the same time their disadvantage – they have to be designed robustly not to be degraded by expected user operations.

Methods using the digital signatures are based on extracting unique features from the image at the source side and encoding these features into the so-called digital signature. These signatures are then used to verify the image integrity. This research direction is popular thus numerous approaches based on one way hashing and digital signature methods have been introduced so far such as SHA, MD5, etc. [25].

Although in the past researchers preferred data hiding and digital watermarking algorithms, recently new passive approach which does not embed any secondary data into the image has become more popular. The passive (also called blind) methods in contrast to the active ones do not use any prior information about the analyzed image. Numerous algorithms have been proposed in this area, trying to detect image merging [26], traces of non-consistencies in color filter array interpolation [27], traces of geometric transformations, [28], cloning [29], traces of uses of computer graphics [30], JPEG compression inconsistencies [31], etc. All these methods are typically using the fact that digital image editing introduces specific detectable statistical changes into the analyzed image.

The proposed algorithms aim at the minimization of necessary computational power and human intervention. They apply passive

approach and are based on detection of traces of double compression and interpolation, and inconsistencies in chromatic aberrations and in omnipresent noise. We have included as well our method for detection of copy–paste forgeries, when some parts of an image are copied and pasted to other image parts to intentionally hide original image content.

4.1. Detecting double compression

When an image is intentionally altered, it is typically loaded into a photo-editing software and after manipulations are carried out the image is once again re-saved. If the original image was already compressed, during the loading procedure the data are uncompressed and at the re-saving step they are compressed again, potentially with other parameter setting of the compression procedure.

For JPEG compression method this double action can be analyzed and used as an indication of an image alternation, since this re-saving introduces specific changes into the altered image due to the difference of used quantizations matrices of unaltered (primary) and modified (secondary) images.

To understand the core of the proposed algorithm we need to show basics of the JPEG algorithm [32]. The image to be compressed is converted from RGB to YCbCr color channels (Y – luminance, Cb and Cr – chrominance components). Each channel is then split into adjacent blocks of 8×8 pixels, transformed by discrete cosine transform (DCT), and quantized. At the end all quantized coefficients are compressed by some variant of Huffman entropy encoding.

The key step is the quantization. Here the 8×8 quantization matrix $Q(u, v)$, which defines the quantization steps for each used DCT frequency, is applied in lossy manner. The quantized DCT coefficients $F^Q(u, v)$ are defined as:

$$F^Q(u, v) = \text{round}\left(\frac{F(u, v)}{Q(u, v)}\right), \quad u, v \in \{0, \dots, 7\}, \quad (7)$$

where $F(u, v)$ are computed DCT coefficients from particular block of individual color channels.

During the double JPEG action the compressions were realized with different quantization matrices Q_α (primary) and Q_β (secondary). The DCT quantized coefficient is said to be double quantized if $Q_\alpha(u, v) \neq Q_\beta(u, v)$. The double quantization is given by:

$$F^{Q^\beta}(u, v) = \text{round}\left(\frac{F^{Q^\alpha}(u, v)Q^\alpha(u, v)}{Q^\beta(u, v)}\right). \quad (8)$$

To determine the presence of double compression artifacts, the method uses a threshold-based quantitative measure. Histograms

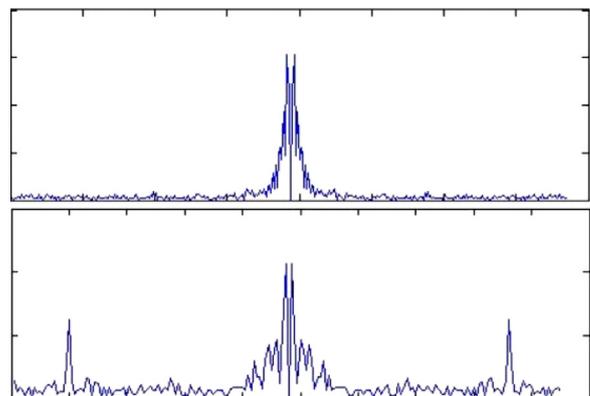


Fig. 5. The top image shows spectrum of a single JPEG image. In the bottom there is shown the spectrum of a double JPEG image.

of DCT coefficients are analyzed in the frequency spectrum (FFT). If their spectral representation exhibits characteristics peak, the image is classified as possibly double JPEG compressed image. The bottom image in Fig 5 demonstrates a typical FFT spectrum of double JPEG images. The image on the top is the spectrum of a single JPEG compressed image. Fig. 15(c) shows an application of the method.

Patterns introduced by double JPEG compression depend on particular compression quality parameters [33]. Proposed method extends the approach introduced in [33], by also allowing to detect double compression locally and thus detecting if two or more images were spliced together. The analyzed image is divided into patches and characteristic peaks are detected patch-wise. The sensitivity to the peak height is adjustable. By varying the sensitivity we can analyze the image and determine if double compression appears non-homogeneously.

It is important to note that detecting the traces of double compression does not necessarily imply the existence of malicious modifications in the image. Often images are re-compressed to achieve smaller size or only simple image adjustment operations such as contrast enhancing were applied. Nonetheless, detecting these changes plays a valuable role in identifying image forgeries.

4.2. Detection of interpolation

When two or more images are spliced together often geometric transformations such as scaling, rotation or skewing of these images are needed in order to create high quality and consistent forgeries. Geometric transformations typically require resampling and interpolation steps. Therefore by having proposed detectors for resampling/interpolation detectors altered images containing tampered portions can be easily identified.

There are two principal steps in geometric transformations. In the first step a spatial rearrangement of pixels of the image is done according to an appropriate transformation function, T , which maps the coordinates of the input image pixels to points in the output image:

$$x' = T_x(x, y), \quad y' = T_y(x, y). \quad (9)$$

The second step of the geometric transformation deals with an interpolation – resulting intensity values for individual pixel positions in the transformed image are assigned by means of a constructed low-pass interpolation filter w and rearranged pixels from the input image. Here, to compute signal values at arbitrary locations, discrete samples of the f_k are multiplied with proper filter weights when convolving them with w . We denote the result of interpolation operation by $f^w(x)$, respectively by $\mathcal{D}^n\{f^w\}(x)$ which would denote n th derivative of $f(x)$.

By assuming that ϑ is an integer, it can be shown that

$$\text{var}\{\mathcal{D}^n\{f^w\}(x)\} = \text{var}\{\mathcal{D}^n\{f^w\}(x + \vartheta\Delta_x)\}, \quad \vartheta \in \mathcal{Z}. \quad (10)$$

In other words it can be shown that interpolation brings into the signal and their derivatives a specific periodicity [28]. This periodicity is dependent on the applied interpolation kernel and can be used for detecting the traces of an interpolation. To this end, a derivative filter is applied to the investigated region, $b(x, y)$, $\mathcal{D}^n\{b(x, y)\}$ and a Radon transformation is employed in order to find traces of an affine transformation. The Radon transformation computes projections of magnitudes of $\mathcal{D}^n\{b(x, y)\}$ along specified directions determined by angle θ (the projection is a line integral in the given direction), resulting in 180 vectors ρ_θ . If the investigated region has been resampled, corresponding auto-covariance sequences defined as $R_{\rho_\theta}(k) = \sum_i (\rho_\theta(i+k) - \bar{\rho}_\theta)(\rho_\theta(i) - \bar{\rho}_\theta)$ will contain a specific strong periodicity (see Fig. 6 bottom-right). In contrast to [28], only the detection of scale and rotation is

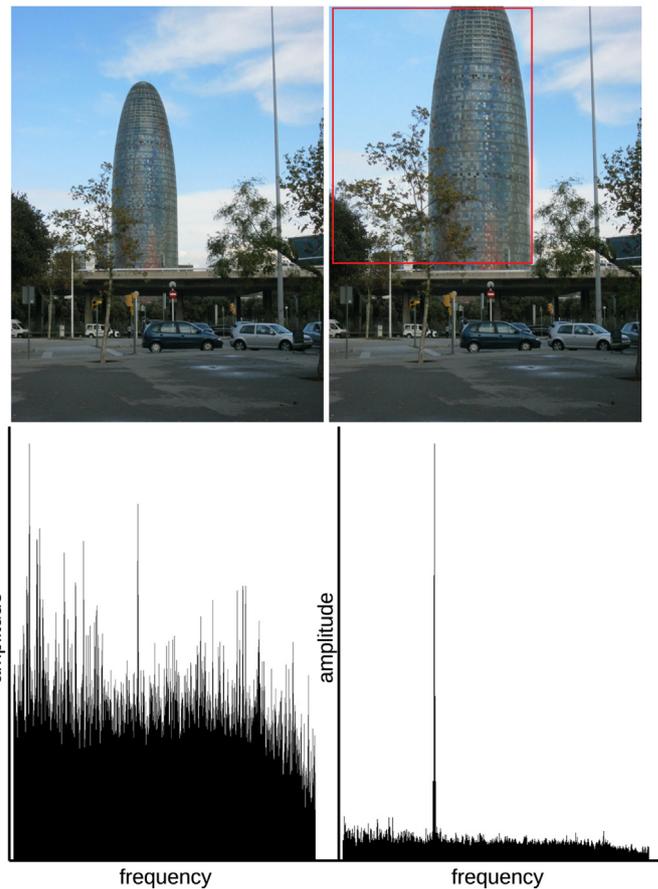


Fig. 6. Detection of interpolation: top-left: the original image stored in the TIFF format; top-right: the image with a re-sized area, denoted by a red box; bottom-left: the result of the method applied on the original image; bottom-right: the result of method applied on the modified image, with a distinctive peak.

supported and results are visualized to the forensic investigator in a single plot so he/she has a full control on decision making.

4.3. Detection of copy-move

One of common types of digital image forgeries is a copy-move forgery [29]. Here, a part of the image is copied, often blurred on its border and pasted into another part of the same image, with the intention to hide an object or a region of the image. Typically, ideal regions for using copy-move forgery are textured areas with irregular patterns, such as grass. Because the copied areas will likely blend with the background it is very difficult for the human eye to detect any suspicious artifacts. Another fact which complicates the detection of this type of tampering is that the copied regions come from the same image, therefore they have similar properties, such as the noise component or color palette. It makes the use of statistical measures to find irregularities in different parts of the image impossible.

Existing copy-move forgery detection methods are mostly based on matching of overlapping image blocks. For example, Fridrich et al. [34] have proposed a method which is based on matching the quantized lexicographically sorted discrete cosine transform (DCT) coefficients of overlapping image blocks. The lexicographical sorting of DCT coefficients is carried out mainly to reduce the computational complexity of the matching step.

The proposed method begins with tiling the image by blocks of $R \times R$ pixels. Blocks are assumed to be smaller than the size of the duplicated regions, which have to be detected. Blocks are horizontally slid by one pixel rightwards starting with the upper

left corner and ending with the bottom right corner. The next step is about the representation of overlapping blocks with a set of features. The method enables to choose between two different kinds of representation of blocks – either a DCT-based representation of blocks or our representation based on blur moment invariants. The DCT representation of blocks is faster and has a lighter requirement on RAM in comparison to moment invariants, while the blur invariants based method has an advantage in its robustness in scenarios where the copied areas have been intentionally blurred. They are functions of central moments [35,36]. Here, the two-dimensional $(p + q)$ th order central moment μ_{pq} of $f(x, y)$ is defined as

$$\text{var}\{\mathcal{D}^n\{f^w\}(x)\} = \text{var}\{\mathcal{D}^n\{f^w\}(x + \vartheta\Delta_x)\}, \quad \vartheta \in \mathcal{Z}, \quad (11)$$

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x-x_t)^p (y-y_t)^q f(x, y) dx dy, \quad (12)$$

where the coordinates (x_t, y_t) given by the relations $x_t = m_{10}/m_{00}$, $y_t = m_{01}/m_{00}$ denote the centroid or the center of gravity of $f(x, y)$. Blur invariants are then defined using the recursive relation:

$$B(p, q) = \mu_{pq} - \alpha \cdot \mu_{pq} - \frac{1}{\mu_{00}} \sum_{n=0}^K \sum_{i=m_1}^{m_2} \binom{p}{t-2i} \binom{q}{2i} \cdot B(p-t + 2i, q-2i) \mu_{t-2i, 2i}, \quad (13)$$

where

$$\begin{aligned} K &= \lfloor (p + q - 4)/2 \rfloor, & t &= 2(K - n + 1), \\ m_1 &= \max(0, \lfloor (t - p + 1)/2 \rfloor), \\ m_2 &= \min(t/2, \lfloor q/2 \rfloor), \\ \alpha &= 1 \Leftrightarrow p \wedge q \text{ are even}, & \alpha &= 0 \Leftrightarrow p \vee q \text{ are odd}. \end{aligned}$$

The proposed algorithm uses 24 blur invariants up to the seventh order to create the feature vector $B = \{B_1, B_2, B_3, \dots, B_{23}, B_{24}\}$ of each block. Using the principal component transformation (preserving the Euclidean distance among blocks) we reduce this dimension.

After the representation of blocks by either DCT or moment invariants, similar blocks are identified. The main assumption here is that a duplicated region consists of many neighboring duplicated blocks. If we find two similar blocks in the analyzed space and if their neighborhoods are also similar to each other, there is high probability that they are duplicated and thus they will be labeled. The output of the algorithm is a map with the same size as the input image (see Fig. 7 bottom), with values either zero (the block at this position is not duplicated) or one (the block at this position is duplicated). The implemented method is based on the optimized analysis of block similarity introduced in [29] resulting in comparable results but faster response.

4.4. Detection of noise inconsistency

Commonly used tool to conceal traces of tampering is the addition of locally random noise to the forged image regions [37]. The amount of noise in an authentic image is usually uniform across the entire image. Adding locally random noise may cause inconsistencies in the image's noise. Therefore, detection of variations of noise levels in an image may signify tampering. Our method is capable of dividing (segmenting) an investigated image into segments with different estimated noise levels. It is based on wavelet analysis followed by tiling high-frequency sub-band into non-overlapping blocks for which noise variation is individually estimated. In the first step a one-level wavelet decomposition [38] of the investigated image is carried out. The high frequency sub-band, HH_1 , gives diagonal details of the image at the highest resolution. Our method tiles this sub-band by non-overlapping blocks B_i of $R \times R$ pixels. Blocks are assumed to be much smaller than the size of the corrupted regions to be detected. The size of



Fig. 7. Detection of copy-move: Top: the original photo; middle: the tampered image with pasted areas; bottom: the method output based on blur moment invariants showing detected duplicated areas.

blocks can be interactively adjusted. For each block we estimate the standard deviation of the noise. It has been shown [39] that it can be robustly estimated from HH_1 using the median based estimator

$$\hat{\sigma} = \text{median}(|HH_1|)/0.6745.$$

The median measurement is insensitive to isolated outliers of potentially high amplitudes. Often $\text{median}(|HH_1|)$ is denoted as

MAD(HH_1) where MAD stands for median absolute deviation. This estimator is very popular and generally provides robust and precise outcomes. Estimated noise levels are visualized (see Fig. 14 right column) in gray level map for identification of potentially tampered areas. The method can be extended by clustering of blocks with similar estimated noise levels [37].

It has to be noted that the noise degradation is the main cause of failure of most existing blind forgery detection methods. These methods are able to work correctly only when the amount of present noise is small. For example, in copy–move forgery additive noise causes mismatches of duplicated regions. This significantly decreases the performance of copy–move forgery detection methods. The same effect can be observed in the methods based on the detection of resampling. Here the noise degradation causes the loss of detectable interpolation based correlation among neighboring pixels.

4.5. Chromatic aberration inconsistency

Camera optical system introduces several kinds of aberration (imperfections) into the image. One of them, chromatic aberration, is caused by failure of a lens to focus all colors to the same convergence point, because lenses have different refractive indices for different wavelengths of light. Although many modern cameras attempt to reduce the effect of chromatic aberration, it is still often manifested as a regular pattern of slight blur and/or color shifts. Deviations from this regularity in a digital photograph can be seen as an indication of tampering.

There are two types of chromatic aberration: longitudinal and lateral. Longitudinal aberration occurs because different wavelengths have different focal planes, therefore all colors cannot be focused (sharp) at once. Lateral aberration occurs because different wavelengths from the same scene point reach the sensor at different positions, therefore different color channels are shifted with respect to each other. The proposed method is based purely on lateral aberration.

It can be derived (e.g., [40]) that the relation between image points of two different color channels (e.g., red channel \mathbf{x}_r and green channel \mathbf{x}_g) corresponding to the same scene point is a simple affine transform with respect to the optical center \mathbf{x}_0 and some isotropic scale α

$$\mathbf{x}_r = \alpha(\mathbf{x}_g - \mathbf{x}_0) + \mathbf{x}_0. \quad (14)$$

The general principle of forgery detection using chromatic aberration is based on the estimation of the unknown parameters $\{\mathbf{x}_0, \alpha\}$. They are estimated both globally, using the entire image, and locally, using only a small patch of the image. If the two estimates are not sufficiently similar, the patch is marked as possibly having been tampered with.

Authors of [40] used mutual information as the similarity measure between color channels and found the transform parameters $\{\mathbf{x}_0, \alpha\}$ using an exhaustive search. Every iteration of such search requires interpolating the entire image and the whole procedure is therefore extremely slow. Motivated by our target application, we propose a method which is much faster and more practical.

Because shifts due to aberrations are small (α in (14) is very close to 1), the affine transform can be locally approximated by constant shift. We therefore set up a grid of regularly spaced positions throughout the entire image and estimate two sets of the shifts – red to green channel and blue to green channel – in these positions. If the estimated shifts locally deviate from the pattern suggested by Eq. (14), it may be a sign of tampering in this particular area of the image. We only display the estimated pattern of shifts in the form of arrows (see Fig. 15(d)). The final decision whether or not the pattern is sufficiently regular is left to the operator.

The key step of the proposed method is estimation of the shift between two color channels for a small patch around center (grid) pixel. Because the effect of chromatic aberration is relatively small, we must find the shift with high subpixel precision. Naive direct approach would require first upsampling the patch N times to reach $1/N$ pixel precision and then performing the registration, which is computationally quite demanding task. The suitable patch sizes depend on an image size, for common 10 megapixel image, our default patch size is 125×125 px, resulting in 29×21 patches. Processing such image with $1/100$ subpixel precision would then require performing two interpolations and the registration of $12,500 \times 12,500$ pixels image (156 megapixel) 609 times for just one of two combinations of color channels, which is computationally prohibitive.

To achieve better performance, we propose modification to the brute force method of [40]. First, we replace mutual information with direct correlation as the similarity measure for registration. Second, we calculate the correlation in the Fourier domain and use the two-step discrete Fourier method (DFT) method described in [41], in which the required upsampling can be done by virtually zero-padding the Fourier transforms of the two patches. The respective shift between two color channels is then estimated by the following procedure:

1. Calculate Fourier transforms of both color channels in the examined patch.
2. Perform correlation in the Fourier domain by pixel-wise multiplication.
3. Perform inverse Fourier transform taking into account the fact that the Fourier transform has been virtually zero-padded to N times their original size, where $1/N$ is the required subpixel precision.
4. Find coordinates of maximum, which is the sought shift (when divided by N).

For the inverse Fourier transform in the step 3 we do not use FFT, which would require physically zero-padding the patches, but rather express the 2D Fourier transform as a matrix multiplication of the patch in the form

$$p = RPC, \quad (15)$$

where p and P are the image patch (one of the color channels) and its Fourier transform, respectively, and R and C are the row and column inverse of the 2D DFT matrix. In the formation of the R and C matrices we take into account the fact that, firstly, we need the inverse FT only in the limited number of coordinates, expecting the inter-channel shift close to zero, and, secondly, that the FTs of the patches have been zero-padded so we include only non-zero input in the calculation.

Using the described procedure, the shift can be accurately estimated in a fraction of time (and memory) required for the naive approach. To further speed-up the process, we first estimate the shift with $1/4$ pixel precision and only in its neighborhood do we refine the estimate to the full $1/100$ precision.

5. Image and video restoration

The main goal of image and video restoration in forensic analysis is to increase the perceivable image content to the end user. This is a frequent situation since videos from surveillance cameras often do not have sufficient resolution and sharpness and are often blurred and noisy. The restoration module is a powerful tool that addresses these issues. It is capable of removing noise and compression artifacts, and if multiple images of the same object (e.g., several frames from a video sequence) are available it also increases spatial resolution. The module consists of three

algorithms that can be run independently or applied sequentially to corrupted input data. Below we describe each method and provide insights into the applied mathematical tools.

5.1. Denoising

Denoising is the most common image and video restoration task. In video surveillance two sources of noise impair the quality of acquired images. We refer to them as external and internal. Two examples of the external source are fog and haze. Various methods were proposed to remove such degradation, recently e.g., [42], but they are not considered here. Instead we focus on the internal source which is intrinsic to any acquisition device. The internal noise is a mixture of Gaussian and Poissonian noises, we approximate the noise for the image processing purposes by the additive Gaussian noise. We write this symbolically

$$g = u + n, \tag{16}$$

where u is the original image, g the observed noisy image and n the Gaussian noise with normal distribution $N(0, \sigma^2)$.

Our choice of denoising method was influenced by the generally valid trade-off between the quality of restoration and the time needed for computation. One group of the best performing fast methods is based on thresholding of wavelet coefficients [43]. In the implemented application we exploited special properties of the dual-tree complex wavelets DT-CWT described in [44]. This type of wavelets has more isotropic behavior than standard wavelets that emphasize vertical and horizontal edges, and weaken diagonal edges. While slightly more time-consuming than standard orthogonal wavelets, the time of computation is still linear in the number of pixels.

Formally, the estimated image equals

$$u = W^{-1} \arg \min_v \frac{1}{2\sigma^2} \|g - W^{-1}v\|^2 + \alpha \|v\|_1, \tag{17}$$

where W is the wavelet transform, W^{-1} its inverse, v the wavelet coefficients of the solution, σ^2 noise variance and $\alpha > 0$ a parameter controlling the level of smoothness. Thanks to the special property of complex wavelets being the Parseval frame, the solution can be computed in linear time with respect to the number of pixels by soft thresholding

$$S_\alpha(a) = \begin{cases} a - \alpha & a > \alpha \\ 0 & |a| \leq \alpha \\ a + \alpha & a < -\alpha, \end{cases} \tag{18}$$

applied on individual coefficients of the wavelet transform, i.e., $u = W^{-1} S_{\alpha\sigma^2}(Wg)$.

To achieve the optimal smoothness, the estimated noise variance σ^2 can be adjusted by user. Fig. 8 demonstrates our result for a very noisy input.

5.2. Super-resolution

Super-resolution (SR) algorithm fuses multiple input low-resolution images (video frames) and estimates the latent high-resolution image (frame). We model the acquisition process and then apply an inverse method to recover the latent image as we proposed originally in [45]. The nature of degradation in the acquisition process implies that this is an ill-posed problem and we need additional regularization. Appropriate numerical methods, such as [46], are necessary to solve non-linear equations, which are the outcome of the regularized functional.

The acquisition process models the observed low-resolution image g_k as the downsampled (D) and warped (T_k) high-resolution



Fig. 8. Denoising: Top: the original image taken in poor lighting conditions causing profound noise; bottom: the output of the proposed restoration.

image u

$$g_k = DT_k u + n, \tag{19}$$

where n is additive noise. The downsampling operator D mimics the sampling phenomenon taking place in the camera sensor by performing convolution with a sensor blur and subsampling the data. The operator D is the same for all the observed images g_k 's and is fully determined by user parameters, which are the sensor blur size and subsampling factor. In our case, we use gaussian blur as the sensor blur as proposed in [47]. The subsampling factor is equivalent to the SR factor and specifies by what ratio we want to increase the resolution of input images. The warping operator T_k geometrically transforms u to be aligned with g_k . The acquisition model assumes that all the observed images g_k 's differ only by the geometric transformations T_k 's, which are however unknown.

The critical step in the SR algorithm is to correctly estimate T_k 's. Typically we choose one input image g_r as a reference image, the corresponding T_r is identity and all other T_k 's are calculated towards the reference image. For this purpose we use an optical flow (OF) algorithm, which estimates local shifts (motion field) between two images with sub-pixel accuracy. The estimated motion field then fully defines the warping matrix T_k . There is a vast number of OF algorithms and several benchmarks that compare their performance [48]. For our purpose we have chosen a method in [49], which has the best ratio of precision to time complexity, and included modifications suggested in [50]. The implementation of the OF algorithm is pyramidal, i.e., calculating shifts on multiple scales, which allows accurate estimation of large and small shifts simultaneously. Smoothness of the estimated motion field is forced by the Total Variation (TV) regularization. If the input images contain artifacts, such as noise or compression artifacts, the motion field often contains outliers negatively influencing the SR step. We have improved the OF algorithm by adding an optional constraint for parametric motion field models. The user can choose between two parametric models: translation and affine. The translation model constrains the estimated motion field to give one global translation vector. Such scenario is typically

useful when we are interested in SR of a relatively small region where the geometric transformation among images is well approximated by translation, e.g., a license plate of a moving car. The affine model constrains the field to be represented by a six-parameter linear transform, which is the most general 2D linear transform. Note that this allows for rotation and scale changes in the input images.

Once the warping operators T_k 's are estimated, we proceed to solve (19). Following the Bayesian paradigm, the optimal solution is the maximum a posteriori (MAP) estimator. Let the noise n be normally distributed and the image prior be a sparsifying distribution $\exp\{-\lambda\|Cu\|_p^p\}$, where $p \leq 1$ and C is a filtering operator returning features that are assumed to be sparse. A common example of C is image gradient. The MAP estimator is then the minimum of the energy function

$$E(u) = \frac{1}{2} \sum_k \|DT_k u - g_k\|^2 + \lambda \|Cu\|_p^p. \quad (20)$$

The first term of E is called a data term and the second one is regularization. Due to the sparsity measure $\|Cu\|_p^p$, derivatives of E are non-linear and we thus apply a linearization method referred to as the half-quadratic algorithm [51]. The problematic term $\|Cu\|_p^p$ is replaced with a quadratic form $\langle u, Lu \rangle$, where L must be iteratively updated with previously calculated Cu . The action of L can be interpreted as space-variant convolution with a Laplacian-like filter, of which coefficients spatially vary based on Cu . In addition, the action of DT_k is also space-variant convolution with a filter (sensor blur) which is shifted by sub-pixel vectors in T_k . In the half-quadratic algorithm we thus iteratively solve for u a linear system

$$\left(\sum_k T_k^* D^* D T_k + \lambda L \right) u = \sum_k T_k^* D^* g_k, \quad (21)$$

for which we use a Conjugate Gradient (CG) method. A fast implementation using FFT is not possible in this case since $\sum_k T_k^* D^* D T_k + \lambda L$ is space-variant convolution. However CG methods are relatively efficient as they typically require a small number of iterations (around 10) for this type of problems.

In Fig. 9 we compare default camera demosaicing with standard SR reconstruction of JPEG images and our SR reconstruction of RAW images. The demosaicing algorithm applies on a single image whereas the SR algorithms take multiple images, therefore SR is expected to provide better results as more information are at disposal to process. The results are cropped to better see and compare the reconstruction quality of the car license plate. The camera demosaicing output in (b) was spatially interpolated by a factor of 2 to obtain the resolution of SR outputs. The demosaicing images are typically used in practice, since they are the default output of any digital camera. The demosaicing process removes color filter array (also called Bayer filter mosaic) and also performs other tasks, such as denoising and color correction, which alters the images in a way not suitable for SR. If we apply standard SR algorithms on such images, we obtain a suboptimal result in (c). In this case the SR factor was 2. However, our SR algorithm is adapted to use directly RAW images and work with Bayer mosaic, which outperforms standard SR as can be seen in (d). The resolution of RAW images is half the resolution of demosaicing images and therefore the SR factor in the second case (d) is 4 to obtain the same resolution as in the first case (c). The license plate is difficult to read after demosaicing and even after applying SR on these images. Our SR on RAW images shows dramatic improvement with all the letters and numbers clearly legible.



(a) RAW input images



(b) Camera demosaicing – JPEG



(c) Super-resolution of JPEG



(d) Super-resolution of RAW

Fig. 9. Super-resolution: (a) sequence of photos captured by a DSLR camera; (b) camera default demosaicing JPEG output of one photo; (c) super-resolution reconstruction from demosaicing JPEG images; (d) super-resolution reconstruction directly from RAW images.

5.3. JPEG artifact removal

Original JPEG standard [52] is based on compression of cosine transform coefficients. Since compression works independently on individual blocks of 8×8 pixels, higher compression ratios result in a visually disturbing checkerboard pattern and artifacts along strong edges (see Fig. 10) [53]. Although there is a newer compression standard JPEG 2000, the original version still prevails because of its simplicity and speed. For this reason, our algorithm was designed specifically to work with image compressed by the original standard.

For the removal of JPEG artifacts we proposed to use an approximation of the Bayesian MAP approach, minimizing



Fig. 10. JPEG restoration: top: the image compressed by JPEG with quality factor 20; bottom: the output of our reconstruction.

simultaneously the l_1 -norm of wavelet coefficients and discrepancy from the quantization constraint $QCu \in (QCg - 0.5, QCg + 0.5)$

$$\arg \min_u \frac{1}{2} \|QCg - QCu\|^2 + \alpha \|Wu\|_1, \quad (22)$$

where C is the orthogonal matrix of block discrete cosine transform and Q the diagonal matrix containing inverted elements of the

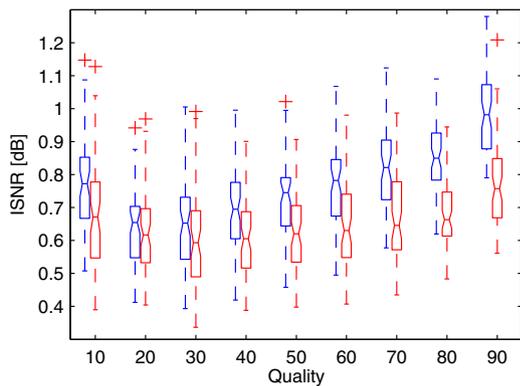


Fig. 11. Statistical comparison of our algorithm and [55] on a database of 60 images. All images were compressed into JPEG format and then decompressed/restored by both algorithms. This was done for nine JPEG quality levels, from 10 to 90. For each quality, our results is shown on the left (blue) and [55] on the right (red). The graph shows the median value, first and third quartiles, whiskers extending to the most extreme data points not considered outliers, and outliers plotted individually. The notches indicate the confidence intervals of the median.

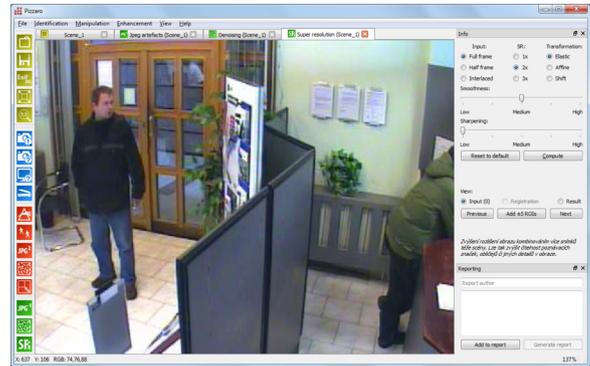


Fig. 12. An example of the PIZZARO user interface window.

quantization table defined by the JPEG standard [52]. For regularization, we used the same wavelet transform W as for image denoising [44], giving results superior to standard wavelets or total variation [54,55]. Fig. 11 shows a statistical comparison of the proposed novel method in terms of SNR with one of recent methods [55] based on total variation.

6. PIZZARO software

All proposed algorithms have been included to PIZZARO [1], the set of software tools for interactive image and video forensic analysis and an application of individual procedures (see Fig. 12 for the user interface). Software PIZZARO was developed in C++ using Qt toolkit for graphical user interface, OpenCV for image manipulation and several libraries for reading specific image/video formats (libpng, libjpeg, libxif, Ffmpeg), and can be run under any version of Windows® from XP up to 8.1 in either 32-bit or 64-bit mode. As no platform specific libraries are used, it is possible to compile the software for other platforms (e.g., OS X, or specific Linux distributions).

It is designed to allow efficient work with images and videos to quickly reveal and apply desired methods, with the possibility to



Fig. 13. Gas station super-resolution use case: top: an input video sequence recorded by a security camera with close-ups of suspect’s left and right hand; bottom: an estimated high-resolution image with two close-ups. Recovered image suggests that the suspect wore a ring and hold the handgun in an unconventional way using the middle finger on a trigger.

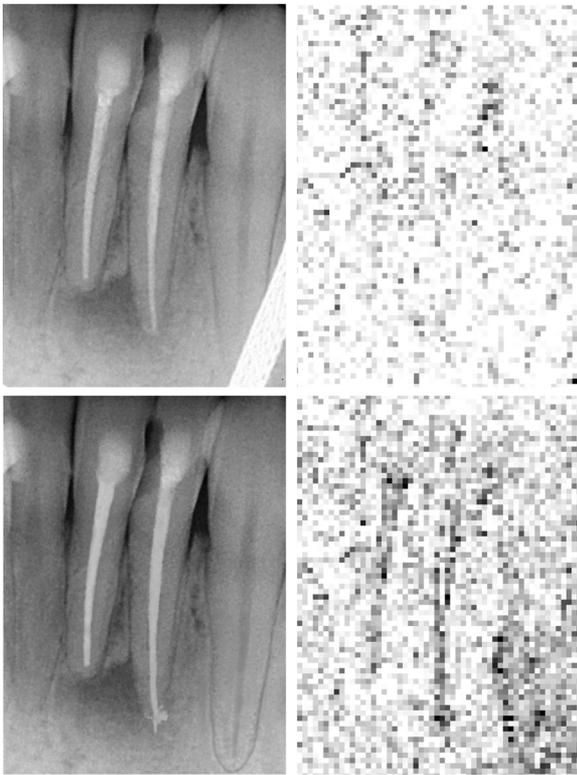


Fig. 14. X-ray scans image content verification use case: left column: dental X-rays from both parties to the lawsuit; right column: the output of the noise inconsistency detection evaluated on the X-rays, the bottom image manifests suspicious structures.

open many images or videos at the same time, compare results achieved using different parameters, view Exif metadata information for JPG images, etc.

The results can be immediately exported to HTML by the integrated reporting system for printing. This functionality enables to log and document how were the results achieved together with applied parameters, operator identity, file paths and dates and other possible settings. All this assures the repeatability of the applied procedures.

7. Use cases

The proposed software system has been successfully applied on several real cases of police investigation in cooperation with different police departments in the Czech Republic.

An example of the utilization of the image and video restoration module is the *Gas station* case. In Fig. 13 there is an output of the video restoration that assisted in a police investigation conducted by the Czech Police Department in Liberec, Division of Public Criminality, which resulted in convicting a suspect of burglary and homicide. Cropped frames from a video sequence recorded by a security camera are on left. The close-ups (top and bottom) of the left and right hand of the suspect show insufficient resolution of the video sequence, where details are missing. They were significantly restored after processing the video with proposed PIZZARO video restoration module (right), which increased the resolution and diminished the omnipresent noise. The most notable is the presence of a golden ring on suspect's left hand and an unconventional way of holding a handgun with a protruding index finger, which suggested that the suspect used his middle finger on a trigger.

Another example of the application of the proposed procedures is related to the demonstration that the photograph was tampered



(a)



(b)



(c)



(d)

Fig. 15. Billboard image content verification use case: (a) the original photo; (b) the image with a modified area; (c) the output of the double-compression detection; (d) the output of the chromatic aberration inconsistency detection.

in order to create false evidence. The analyzed image data were two dental X-ray scans capturing the same patient (Fig. 14, left column). They were supposed to be used to validate applied medical procedure and they were used by opposite parties to the

dispute. PIZZARO methods were here applied to investigate the originality of the data. We evaluated the noise consistency results of which are demonstrated in Fig. 14 in the right column. While the noise structure of the upper X-ray scan does not relate to the image content, the bottom X-ray noise is clearly related to the image content, which is very unlikely. The conclusion that the bottom X-ray is fraud was then even more significantly confirmed by the fact, that these two scans are geometrically identical apart from the areas where the noise inconsistencies were manifested (the procedure of the X-ray data acquisition makes it almost absolutely impossible to obtain two spatially identical scans).

Billboard, the last example of the PIZZARO application, comes from the area of investigation of tax evasion. Here, the question to be answered is whether the photographs of billboards justifying the tax payer costs on his marketing campaign are genuine, capturing real billboards (see Fig. 15(a)) and are not just results of Photoshop work when the fake billboard image content was copied into the real scenery (Fig. 15(b)). The latter case represents tax fraud, because the tax payer is claiming costs which were not spent in reality. Fig. 15(c) demonstrates the output of the double compression detection module, where the dark area marks parts of the photograph with suspicious parameters settings. The same area was identified by the procedure testing the consistency of chromatic aberrations (see Fig. 15(d), the arrows have random direction and size).

The experience with cases analyzed by PIZZARO software has shown that it is difficult to provide some defined manual how to proceed with various tasks. The variability of possible scenarios is large. Moreover, the three modules of the proposed PIZZARO toolbox are each oriented on different problems, so not too often methods from more modules are used on the same dataset. There are some limitations which should be taken into account not to decrease the efficiency of the methods. Denoising or JPEG artifact removal should not be applied before any of the other method, otherwise mathematical assumptions of the algorithms will be violated. Denoising can be achieved by superresolution (SR) too, so the denoising of the output of SR does not make sense. Even more difficult situation arises when the forensic case under investigation is to question the statement that the photograph was tampered. Methods verifying the authenticity of image content often provide only an indication that some parts of the photographs were tampered because they manifest suspicious characteristics. The recommended practice is to apply several methods for content verification in order to get stronger evidence.

8. Conclusion

We have introduced a set of methods for image and video forensic analysis. This work was based on tight cooperation of scientists from the Institute of Criminalistics, National Drug Headquarters of the Criminal Police and Investigation Service of the Police of the Czech Republic, and image processing experts from the Czech Academy of Sciences and thus reflecting best practices used in the criminal investigation utilizing images and/or videos. We have addressed two key problems identified as the most important issues of which automation can facilitate criminalists work. They are verification of image/video data with respect to their credibility and origin and image restoration aimed at diminishing unwanted artifacts and at the same time increasing the data quality using all available information.

The described methods include the novel approaches for: an assignment of the images to the source device, the image content verification methods based on various image characteristics, and finally, methodology for removing compression artifacts and even increasing the data resolution. The functionality of some methods was illustrated by means of the comparison to other existing

approaches and by demonstrating performance tests. Next to this, the detection of LCD image re-capture and noise removal were implemented in order to provide required functionality. Proposed methods were designed or modified in order to fulfill time, quality, and interactivity criteria. It should be noted that the output of the proposed methods has often an indicative nature thus the recommended practice is to apply several methods to get stronger evidence.

All theoretical results were implemented in the PIZZARO software tool (available for testing and licensing), which consists of the image processing functionality and reporting and archiving functions to ensure the repeatability of image analysis procedures, fulfilling formal aspects of the analysis work.

Acknowledgements

The work was supported by the Czech Ministry of the Interior, Grant No. VG20102013064 and by GACR Grant No. GA13-29225S.

References

- [1] PIZZARO – Tools for Imaging Device Identification, Authentication, and Image Reconstruction, <http://pizzaro.utia.cas.cz>.
- [2] Access Data, <http://accessdata.com/solutions/digital-forensics/forensic-toolkit-ftk>.
- [3] Fourandsix, <http://www.fourandsix.com/>.
- [4] Verifeyed, <http://www.verifeyed.com/>.
- [5] Belkasoft, <http://belkasoft.com/en/>.
- [6] Forensic Pathways, <http://www.forensic-pathways.com/products-and-services/forensic-image-analyser>.
- [7] Error Level Analysis, <http://www.errorlevelanalysis.com/>.
- [8] Amped, <http://ampedsoftware.com/>.
- [9] M. Jerian, S. Paolino, F. Cervelli, S. Carrato, A. Mattei, L. Garofano, A forensic image processing environment for investigation of surveillance video, *Forensic Sci. Int.* 167 (2–3) (2007) 207–212.
- [10] Motion DSP, <http://www.motiondsp.com/products/lkenalSR>.
- [11] Cognitech, <https://cognitech.com/>.
- [12] RTC Vision, <http://www.rtc-vision.com/>.
- [13] Impress, <http://www.impress4video.com/>.
- [14] Reproducible Research, <http://reproducibleresearch.net/super-resolution>.
- [15] J. Lukas, J. Fridrich, M. Goljan, Digital camera identification from sensor pattern noise, *IEEE Trans. Inf. Forensic Secur.* 1 (2) (2006) 205–214.
- [16] M. Chen, M. Goljan, J. Lukas, Determining image origin and integrity using sensor noise, *IEEE Trans. Inf. Forensic Secur.* 3 (1) (2008) 74–90.
- [17] M. Chen, J. Fridrich, M. Goljan, J. Lukas, Source digital camcorder identification using sensor photo response non-uniformity, in: *Proc. SPIE*, 6505, 2007, pp. 65051G–65051G-12.
- [18] B. Mahdian, A. Novozamsky, S. Saic, Determination of stop-criterion for incremental methods constructing camera sensor fingerprint, in: *13th Int. Workshop on Digital-Forensics and Watermarking*, Taipei, Taiwan, 2014.
- [19] H. Cao, A.C. Kot, Identification of recaptured photographs on LCD screens, in: *2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, IEEE, 2010, 1790–1793.
- [20] T. Ojala, M. Pietikinen, T. Menp, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (7) (2002) 971–987.
- [21] M. Kharrazi, H. Sencar, N. Memon, Blind source camera identification, in: *2004 International Conference on Image Processing, ICIP'04*, vol. 1, 2004, 709–712.
- [22] Y. Chen, Z. Li, M. Li, W.-Y. Ma, Automatic classification of photographs and graphics, in: *2006 IEEE International Conference on Multimedia and Expo*, 2006, 973–976.
- [23] B. Mahdian, R. Nedbal, S. Saic, Blind verification of digital image originality: a statistical approach, *IEEE Trans. Inf. Forensics Secur.* 8 (9) (2013) 1531–1540.
- [24] M. Arnold, M. Schmucker, S.D. Wolthusen, *Techniques and Applications of Digital Watermarking and Content Protection*, Artech House, Inc., Norwood, MA, USA, 2003.
- [25] J. Katz, Y. Lindell, *Introduction to Modern Cryptography* (Chapman & Hall/CRC Cryptography and Network Security Series), Chapman & Hall/CRC, 2007.
- [26] Z. Lint, R. Wang, X. Tang, H.-Y. Shum, Detecting doctored images using camera response normality and consistency, in: *CVPR'05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* – vol. 1, IEEE Computer Society, Washington, DC, USA, 2005, pp. 1087–1092.
- [27] A. Popescu, H. Farid, Exposing digital forgeries in color filter array interpolated images, *IEEE Trans. Signal Process.* 53 (10) (2005) 3948–3959.
- [28] B. Mahdian, S. Saic, Blind authentication using periodic properties of interpolation, *IEEE Trans. Inf. Forensics Secur.* 3 (3) (2008) 529–538.
- [29] B. Mahdian, S. Saic, Detection of copy–move forgery using a method based on blur moment invariants, *Forensic Sci. Int.* 171 (2–3) (2007) 180–189.

- [30] A.E. Dirik, S. Bayram, H.T. Sencar, N. Memon, New features to identify computer generated images, in: IEEE International Conference on Image Processing, ICIP'07, vol. 4, 2007, 433–436.
- [31] J. Fridrich, T. Pevny, Detection of double-compression for applications in steganography, IEEE Trans. Inf. Secur. Forensics 3 (2) (2008) 247–258.
- [32] W.B. Pennebaker, J.L. Mitchell, JPEG Still Image Data Compression Standard, Kluwer Academic Publishers, Norwell, MA, USA, 1992.
- [33] B. Mahdian, S. Saic, Detecting double compressed JPEG images, in: The 3rd International Conference on Imaging for Crime Detection and Prevention (ICDP-09), London, UK, 2009.
- [34] J. Fridrich, D. Soukal, J. Lukas, Detection of copy–move forgery in digital images, in: Proceedings of Digital Forensic Research Workshop, IEEE Computer Society, Cleveland, OH, USA, 2003, pp. 55–61.
- [35] J. Flusser, T. Suk, Degraded image analysis: an invariant approach, IEEE Trans. Pattern Anal. Mach. Intell. 20 (6) (1998) 590–603.
- [36] J. Flusser, T. Suk, S. Saic, Image features invariant with respect to blur, Pattern Recognit. 28 (11) (1995) 1723–1732.
- [37] B. Mahdian, S. Saic, Using noise inconsistencies for blind image forensics, Image Vis. Comput. 27 (10) (2009) 1497–1503.
- [38] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
- [39] D. Donoho, I. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Biometrika 8 (1994) 425–455.
- [40] M.K. Johnson, H. Farid, Exposing digital forgeries through chromatic aberration, in: Proceedings of the 8th Workshop on Multimedia and security, ACM, 2006, pp. 48–55.
- [41] M. Guizar-Sicairos, S.T. Thurman, J.R. Fienup, Efficient subpixel image registration algorithms, Opt. Lett. 33 (2) (2008) 156–158.
- [42] K.B. Gibson, T.Q. Nguyen, An analysis and method for contrast enhancement turbulence mitigation, IEEE Trans. Image Process. 23 (7) (2014) 3179–3190, <http://dx.doi.org/10.1109/TIP.2014.2328180>.
- [43] S. Mallat, A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way, 3rd ed., Academic Press, 2008.
- [44] N. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, Appl. Comput. Harmon. Anal. 10 (3) (2001) 234–253.
- [45] F. Sroubek, G. Cristobal, J. Flusser, A unified approach to superresolution and multichannel blind deconvolution, IEEE Trans. Image Process. 16 (2007) 2322–2332.
- [46] J. Eckstein, D.P. Bertsekas, On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators, Math. Program. 55 (1992) 293–318.
- [47] D. Capel, Image Mosaicing and Super-Resolution (Cphc/Bcs Distinguished Dissertations), SpringerVerlag, 2004.
- [48] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, R. Szeliski, A database and evaluation methodology for optical flow, Int. J. Comput. Vis. 92 (1) (2011) 1–31.
- [49] C. Liu, Beyond pixels: Exploring new representations and applications for motion analysis (Ph.D. thesis), Massachusetts Institute of Technology, 2009.
- [50] D. Sun, S. Roth, M. Black, Secrets of optical flow estimation and their principles, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, 2432–2439.
- [51] G. Aubert, P. Kornprobst, Mathematical Problems in Image Processing, Springer Verlag, New York, 2002.
- [52] JPEG File Interchange Format (JFIF), Tech. rep., ECMA TR/98, Ecma International, 2009.
- [53] B. Gunturk, Y. Altunbasak, R. Mersereau, Multiframe resolution-enhancement methods for compressed video, IEEE Signal Process. Lett. 9 (6) (2002) 170–174.
- [54] F. Alter, S. Durand, J. Froment, Adapted total variation for artifact free decomposition of JPEG images, J. Math. Imaging Vis. 23 (2005) 199–211.
- [55] K. Bredies, M. Holler, A total variation-based jpeg decompression model., SIAM J. Imaging Sci. 5 (1) (2012) 366–393.