Image Forgery Localization via Block-Grained Analysis of JPEG Artifacts

Image Forgery Localization via Block-Grained Analysis of JPEG Artifacts

Tiziano Bianchi*, *Member, IEEE,* and Alessandro Piva, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a forensic algorithm to discriminate between original and forged regions in JPEG images, under the hypothesis that the tampered image presents a double JPEG compression, either aligned (A-DJPG) or non-aligned (NA-DJPG). Unlike previous approaches, the proposed algorithm does not need to manually select a suspect region in order to test the presence or the absence of double compression artifacts. Based on an improved and unified statistical model characterizing the artifacts that appear in the presence of both A-DJPG or NA-DJPG, the proposed algorithm automatically computes a likelihood map indicating the probability for each 8×8 discrete cosine transform block of being doubly compressed. The validity of the proposed approach has been assessed by evaluating the performance of a detector based on thresholding the likelihood map, considering different forensic scenarios. The effectiveness of the proposed method is also confirmed by tests carried on realistic tampered images. An interesting property of the proposed Bayesian approach is that it can be easily extended to work with traces left by other kinds of processing.**

***Index Terms*—Image forensics, JPEG artifacts, double JPEG compression, forgery localization**

I. Introduction

The diffusion of tampered visual contents through the digital world is becoming increasing and worrying, due to the large availability of simple and effective image and video processing tools. Because of this issue, in many fields like insurance, law and order, journalism, and medical applications, there is an increasing interest in tools allowing to grant the credibility of digital images as sources of information. As a possible answer to this request, many image forensic techniques have been proposed to detect the presence of forgeries in digital images through the analysis of the presence or of the perturbation of some traces that remain in digital content, during the creation process or any other successive processing [1].

The JPEG format is adopted in most of the digital cameras and image processing tools; many forensic tools have thus been studied to detect the presence of tampering in this class of images. In general, the manipulation is detected by analyzing proper artifacts introduced by JPEG recompression occurring when the forged image is created; in particular, such artifacts can be categorized into two classes, according to whether the second JPEG compression adopts a discrete cosine transform (DCT) grid aligned with the one used by the first compression or not. The first case will be referred to as aligned double JPEG (A-DJPG) compression, whereas the second case will be referred to as non aligned double JPEG (NA-DJPG) compression.

Approaches belonging to the first category include [2]–[8], whereas the presence of non aligned double JPEG compression has been investigated in [9]–[12].

Based on the observation that the distribution of the first digit of DCT coefficients in single JPEG compressed images follows the generalized Benford distribution, in [3], [4] the statistical distribution of first digits in quantized DCT coefficients is used as feature set for detecting double JPEG compression. Their performance however does not seem adequate, and are outperformed by later works: e.g. in [6], starting from the observation that recompression induces periodic artifacts and discontinuities in the image histogram, a set of features is derived from the pixels histogram to train a support vector machine (SVM) allowing to detect an A-DJPG compression; in [13], the histogram of a subset of 9 DCT coefficients is also used to train a SVM and make the same detection. These two last approaches, however, have been tested only for secondary quality factors set to 75 or 80.

A promising approach is the one introduced by Popescu et al. in [2]: here, it is proposed to detect the presence of double aligned JPEG compression by observing that consecutive quantizations introduce periodic artifacts into the histogram of DCT coefficients; these periodic artifacts are visible in the Fourier domain as strong peaks in medium and high frequencies. Their seminal work has been the basis of the work presented in [7], where double JPEG compression is detected by computing a tampering probability map of the image according to a proper statistical model of DCT coefficients. In [8], an improved version of the model proposed in [7] is presented, leading to a significant improvement of the accuracy of the probability map estimation and consequently of the algorithm performance.

In [5], a different approach to detect areas which have undergone a double JPEG compression is proposed. The method works by comparing differently compressed versions of the image with the possibly tampered one; when the same quality factor of the tampered area is adopted, a spatial local minima, the so-called JPEG ghosts, will appear in correspondence of the forgery. This method works only if the tampered region has a lower quality factor than the rest of the image.

Concerning the methods for the detection of NA-DJPG, a well known approach is the one presented in [9]: starting from

The authors are with the Department of Electronics and Telecommunications, University of Florence, and with CNIT Research Unit at University of Florence, Via S. Marta 3, I-50139, Firenze, Italy (phone: +39 055 4796380, fax: +39 055 4796497, e-mail: {tiziano.bianchi, alessandro.piva}@unifi.it).

This work was partially supported by the REWIND Project funded by the Future and Emerging Technologies (FET) programme within the 7FP of the European Commission, under FET-Open grant number: 268478, and by the project LivingKnowledge - Facts, Opinions and Bias in Time funded by the European Commission under contract no. 231126.

an idea proposed in [14] to detect blocking artifacts, a 8×8 blocking artifact characteristics matrix (BACM) is computed in the pixel domain to measure the symmetrical property of the blocking artifacts in a JPEG image; an asymmetric BACM will reveal the presence of misaligned JPEG compressions. The previous algorithm is modified in [10] to localize the tampered regions, according to two possible approaches: in a block-wise approach each 8×8 image block is processed independently by analyzing the statistics of a rectangular region surrounding it, whereas in the region-wise approach the image is first segmented into homogeneous regions and then each region is analyzed separately. In [11] the blocking artifacts in the pixel domain are again investigated, but in this case their periodic property is measured by devising two proper sets of features fed to a SVM; this method shows higher performance with respect to [9]. In [12], the authors propose a method based on a single feature which depends on the integer periodicity of the DCT coefficients when the DCT is computed according to the grid of the previous JPEG compression, and whose experimental results are superior to the previous works.

In [15] a detection method which is able to detect either block-aligned or misaligned recompression is proposed; in particular the authors propose to compute a set of features to measure the periodicity of blocking artifacts, perturbed in presence of NA-DJPG compression, and a set of features to measure the periodicity of DCT coefficients, perturbed when an A-DJPG compression is applied; this set of nine periodic features is used to train a classifier allowing to detect if an image has undergone a double JPEG compression. Experimental results show that this method outperforms the scheme proposed in [9] for the NA-DJPG case, and the schemes proposed in [4], [7].

Most of the above algorithms rely on the hypothesis to know the right location of the possibly manipulated area, for example by applying a segmentation of the image under test before the forensic analysis as done in [10], or they are just designed to analyse the whole image, so that the correct localization of the forgery in a tampered image is still an open issue. To the best of our knowledge, only the forensic algorithms presented in [7] and [8] have been designed to localize in an automatic way the tampered regions with a fine-grained scale of 8×8 image blocks; let us note that these methods work only in the presence of aligned double JPEG compression, whereas no similar methods exist in the case of non-aligned double JPEG compression.

In this paper, we move a step forward in the localization of tampering in JPEG images, by proposing a forensic algorithm that can reveal a tampering at a local level, without any prior information about the location of the manipulated area, in the presence of double JPEG compression, either aligned or non-aligned. The output of the algorithm, as the methods proposed in [7] and [8], is a map that gives the probability, or the likelihood, for each 8×8 block to be tampered. Starting from the analysis of the probability models of DCT coefficients carried out in [8], we propose an improved and unified statistical model characterizing the artifacts that appear in the presence of either A-DJPG or NA-DJPG, allowing to derive algorithms working for any kind of recompression. As a further contribution of this paper, the proposed forgery localization method, following a Bayesian approach, can be easily generalized to work with similar features derived by any other footprints left by a tampering operation, in the spatial domain or in another image domain.

The rest of this paper is organized as follows. In Section II, the possible scenarios where the algorithm for the detection of double JPEG compression can be applied are briefly reviewed. In Section III, the statistical models used to characterize both A-DJPG and NA-DJPG artifacts are explained. Starting from these models, in Section IV the proposed algorithm is described. The results of the experimental analysis carried out to evaluate the performance of the proposed method are discussed in Section V, whereas in Section VI, some conclusions are drawn.

II. Forensic Scenarios

One of the most evident traces of tampering in JPEG images is the presence of artifacts due to double compression. Basically, such artifacts can be categorized into two classes, according to whether the second JPEG compression uses a DCT grid aligned with the one of the first compression or not. However, in order to correctly interpret the presence or the absence of these artifacts, different scenarios have to be considered.

A first scenario is that in which an original JPEG image, after some localized forgery, is saved again in JPEG format. We can assume that during forgery an image processing technique which disrupts JPEG compression statistics is applied. Examples of this kind of manipulation are a cut and paste from either a non compressed image or a resized image, or the insertion of computer generated content. In this case, DCT coefficients of unmodified areas will undergo a double JPEG compression thus exhibiting double quantization (DQ) artifacts, while DCT coefficients of forged areas will likely not have the same DQ artifacts.

A second scenario is that of image splicing. Here, it is assumed that a region from a JPEG image is pasted onto a host image that does not exhibit the same JPEG compression statistics, and that the resulting image is JPEG recompressed. In this case, the forged region will exhibit double compression artifacts, whereas the non manipulated region will not present such artifacts.

In the first scenario, the most common artifacts will be the ones due to A-DJPG compression. This is the same scenario considered in [7]. However, it is still possible that the original image is randomly cropped before being recompressed in JPEG format, thus creating NA-DJPG artifacts. Conversely, in the second scenario, assuming a random placement of the spliced region, there is a probability of $63/64$ that the forged region will exhibit NA-DJPG artifacts. This is the same scenario considered in [9] [16].

It is worth noting that the above scenarios do not require one part of the image to be singly compressed. For example, if a portion from a JPEG image is pasted onto another JPEG image, we could have an A-DJPG part and a NA-DJPG part, two A-DJPG parts, or two NA-DJPG parts. Nevertheless, as

TABLE I
NOTATION

$\mathbf{I}$	image
$\mathbf{C}$	quantized DCT coefficients
$\mathbf{U}$	unquantized DCT coefficients
$\mathbf{D}_{rc}$	8×8 block DCT matrix, grid aligned to (r, c) pixel position
$\mathcal{Q}$	quantization according to 8×8 quantization matrix
$\mathcal{D}$	dequantization according to 8×8 quantization matrix
x	generic DCT coefficient (either quantized or not)
(r, c)	grid shift
Q	quantization step
k	DCT coefficient frequency index
(i, j)	DCT block position within the image
$(\cdot)_k$	select kth DCT coefficient from each 8×8 block
$\mathcal{L}(i, j)$	likelihood map

long as different kinds of artifacts can be detected on original and forged regions, forensic tools able to identify either A-DJPG or NA-DJPG compression could be very useful to discern original regions from tampered ones.

III. Double JPEG Compression Models

In this section, we will review the statistical models used to characterize both A-DJPG and NA-DJPG artifacts. Then, we will introduce some simplifications that will be useful in defining the proposed detection algorithm, as well as some modifications needed to take into account the effects of rounding and truncation errors between the first compression and the second compression. The notation used hereafter is summarized in Table I.

A. JPEG Compression Model

The JPEG compression algorithm can be modeled by three basic steps [17]: 8×8 block DCT of the image pixels, uniform quantization of DCT coefficients with a quantization matrix whose values depend on a quality factor QF, entropy encoding of the quantized values. The image resulting from decompression will be obtained by applying the inverse of each step in reverse order: entropy decoding, dequantization, inverse block DCT. In the following analysis, we will consider that quantization is achieved by dividing each DCT coefficient by a proper quantization step Q and rounding the result to the nearest integer, whereas dequantization is achieved by simply multiplying by Q.

Let us then assume that an original uncompressed image $\mathbf{I}$ is JPEG compressed with a quality factor QF, and then decompressed. Since entropy encoding is perfectly reversible, the image obtained after JPEG decompression can be modeled as follows:

$$\mathbf{I}_1 = \mathbf{D}_{00}^{-1}\mathcal{D}(\mathcal{Q}(\mathbf{D}_{00}\mathbf{I})) + \mathbf{E}_1 = \mathbf{I} + \mathbf{R}_1. \tag{1}$$

In the above equation, $\mathbf{D}_{00}$ models an 8×8 block DCT with the grid aligned with the upper left corner of the image, $\mathcal{Q}(\cdot)$ and $\mathcal{D}(\cdot)$ model quantization and dequantization processes, respectively, and $\mathbf{E}_1$ is the error introduced by rounding and truncating the output values to eight bit integers. The last quantity $\mathbf{R}_1$ can be thought of as the overall approximation error introduced by JPEG compression with respect to the original uncompressed image. In the above chain, if rounding/truncation (R/T) errors are neglected, the only operation causing a loss of information is the quantization process $\mathcal{Q}(\cdot)$.

In the following, we will analyse the artifacts that appear in presence of a double JPEG compression. These artifacts, caused by the interaction of successive quantization and dequantization processes, will depend on the alignment or misalignment of the respective block DCTs.

B. A-DJPG Compression

In the case of A-DJPG compression, we consider that the original image $\mathbf{I}$ has been JPEG compressed with a quality factor QF_1, decompressed, and again compressed with a quality factor QF_2, with the respective block DCTs perfectly aligned.

Let us consider the quantized DCT coefficients obtained after entropy decoding the doubly compressed image. Such coefficients can be modeled as

$$\mathbf{C}_2 = \mathcal{Q}_2(\mathbf{D}_{00}\mathbf{I}_1) = \mathcal{Q}_2(\mathcal{D}_1(\mathcal{Q}_1(\mathbf{U})) + \mathbf{D}_{00}\mathbf{E}_1) \tag{2}$$

where $\mathbf{U} = \mathbf{D}_{00}\mathbf{I}$ are the unquantized DCT coefficients of $\mathbf{I}$ and $\mathcal{Q}_1$, $\mathcal{Q}_2$ denote that different quantization matrices may be used. In the above equation, we can recognize a first term taking into account quantization effects and a second term modeling R/T errors.

Since the JPEG standard uses 64 different quantization steps, one for each of the 64 frequencies within a 8×8 DCT, the quantized coefficients will be distributed according to 64 different probability distributions. According to (2), each of these distributions can be expressed as

$$p_{\text{DQ}}(x; Q_1, Q_2) = \sum_{v=Q_2x-Q_2/2}^{Q_2x+Q_2/2} p_1(v; Q_1) * g_{\text{DQ}}(v) \tag{3}$$

where Q_1 and Q_2 are the quantization steps of the first and last compression, $g_{\text{DQ}}(v)$ models the distribution of the R/T error in the DCT domain, given by the term $\mathbf{D}_{00}\mathbf{E}_1$, $*$ models convolution, and

$$p_1(v; Q_1) = \begin{cases} \sum_{u=v-Q_1/2}^{v+Q_1/2} p_0(u) & v = kQ_1 \\ 0 & \text{elsewhere} \end{cases} \tag{4}$$

models the distribution of the DCT coefficients after quantization by Q_1 and dequantization, given by the term $\mathcal{D}_1(\mathcal{Q}_1(\mathbf{U}))$. In the above equation, $p_0(u)$ represents the distribution of the unquantized coefficients.

The R/T error in the DCT domain is obtained as a linear combination of R/T errors on the pixel values, where the latter can be assumed i.i.d. random variables. Hence, thanks to the central limit theorem, it can be assumed Gaussian distributed with mean μ_e and variance σ_e^2, i.e.

$$g_{\text{DQ}}(v) = \frac{1}{\sigma_e\sqrt{2\pi}} e^{-(v-\mu_e)^2/\sigma_e^2}. \tag{5}$$

When A-DJPG is not present, the only visible effect is a single quantization with quality factor QF_2 and the quantized DCT coefficients can be expressed as

$$p_{\text{NDQ}}(x; Q_2) = \sum_{v=Q_2x-Q_2/2}^{Q_2x+Q_2/2} p_0(v). \tag{6}$$

The above model holds also if the image was previously JPEG compressed but the DCT grids were nor aligned, since this usually destroys the effects of the primary quantization [18].

C. NA-DJPG Compression

In the case of NA-DJPG compression, we can assume that the original image $\mathbf{I}$ has been JPEG compressed using a DCT grid shifted by $(r, c) \neq (0, 0)$, $0 \leq r \leq 7$ and $0 \leq c \leq 7$, with respect to the upper left corner, so that the image obtained after JPEG decompression can be represented as

$$\mathbf{I}_1 = \mathbf{D}_{rc}^{-1}\mathcal{D}_1(\mathcal{Q}_1(\mathbf{D}_{rc}\mathbf{I})) + \mathbf{E}_1 \tag{7}$$

where $\mathbf{D}_{rc}\mathbf{I}$ are the unquantized DCT coefficients of $\mathbf{I}$ and $\mathcal{Q}_1$, $\mathcal{D}_1$ denote that a proper quantization matrix corresponding to the quality QF_1 was used[1].

We then assume that the image has been again JPEG compressed with a quality factor QF_2, but now with the block grid aligned with the upper left corner of the image. However, if we consider the image after the second decompression, i.e., $\mathbf{I}_2 = \mathbf{I_1} + \mathbf{R}_2$, and we apply a block DCT with alignment (r, c), we have

$$\mathbf{D}_{rc}\mathbf{I}_2 = \mathcal{D}_1(\mathcal{Q}_1(\mathbf{D}_{rc}\mathbf{I})) + \mathbf{D}_{rc}(\mathbf{E}_1 + \mathbf{R}_2). \tag{8}$$

According to the above equation, the unquantized DCT coefficients obtained by applying to the doubly compressed image $\mathbf{I}_2$ a block DCT with alignment (r, c) (i.e., the same alignment of the first compression) can be expressed as the sum of a first term depending from prior quantization (analogous to the first term in (2)) and a second term modeling the approximation error due to the second JPEG compression. Hence, the distribution of DCT coefficients of a particular frequency can be modeled as

$$p_{\mathrm{Q}}(x; Q_1) = p_1(x; Q_1) * g_{\mathrm{Q}}(x) \tag{9}$$

where $g_{\mathrm{Q}}(x)$ models the distribution of the JPEG approximation error due to the second compression in the DCT domain. If we model the approximation error as the sum of the R/T error plus the quantization error due to uniform quantization with step Q_2[2], by invoking the central limit theorem we can assume that the approximation error is Gaussian distributed with mean μ_e and variance $\sigma_e^2 + Q_2^2/12$, i.e.

$$g_{\mathrm{Q}}(x) = \frac{1}{\sqrt{2\pi(\sigma_e^2 + Q_2^2/12)}} e^{-(x-\mu_e)^2/(\sigma_e^2+Q_2^2/12)}. \tag{10}$$

In the absence of NA-DJPG compression, that is the image did not undergo a first JPEG compression with alignment (r, c), the unquantized DCT coefficients obtained by applying a shifted block DCT can be assumed distributed approximately as the original unquantized coefficients, that is

$$p_{\mathrm{NQ}}(x) = p_0(x) \tag{11}$$

since a misalignment of the DCT grids usually destroys the effects of quantization [18].

[1]With a slight abuse of notation, in (2) and (7) we use the same symbols $\mathcal{Q}_1(\cdot)$ and $\mathcal{D}_1(\cdot)$ to denote slightly different operators, since the respective quantization matrices are aligned to different grids.

[2]This is a strong approximation, since quantization error depends on the whole quantization matrix and the shift (r, c). However, it allows us to keep the model reasonably simple.

D. Simplified Models

Although the models in (3) and (9) are quite accurate, they require the knowledge of the distribution of the unquantized coefficients $p_0(u)$, which may not be available in practice. However, it is possible to make some simplifications in order to obtain models that are less dependent from the image content.

In the case of A-DJPG, it was already observed in [2] that by neglecting the effects of R/T errors the distribution of the quantized DCT coefficients is given by

$$p_{\mathrm{DQ}}(x; Q_1, Q_2) = \sum_{u=L(x)}^{R(x)} p_0(u). \tag{12}$$

where, denoting as $\lceil x \rceil$ and $\lfloor x \rfloor$ the ceiling and floor function, respectively, $L(x) = Q_1 \left(\left\lceil \frac{Q_2}{Q_1} \left(x - \frac{1}{2} \right) \right\rceil - \frac{1}{2} \right)$, and $R(x) = Q_1 \left(\left\lfloor \frac{Q_2}{Q_1} \left(x + \frac{1}{2} \right) \right\rfloor + \frac{1}{2} \right)$ [7].

A simplified version of the above model can be obtained by introducing the following approximation [8]

$$\begin{aligned} &\frac{1}{R(x) - L(x)} \sum_{u=L(x)}^{R(x)} p_0(u) \\ &\qquad \approx \frac{1}{Q_2} \sum_{u=Q_2 x - Q_2/2}^{Q_2 x + Q_2/2} p_0(u) = \frac{1}{Q_2} p_{\mathrm{NDQ}}(x; Q_2). \end{aligned} \tag{13}$$

The above approximation holds whenever the function $n_{\mathrm{DQ}}(x) = (R(x) - L(x))/Q_2 > 0$ and the histogram of the original DCT coefficient is locally uniform. In practice, we found that for moderate values of Q_2 this is usually true, except for the center bin ($x = 0$) of the AC coefficients, which have a Laplacian-like distribution. According to the above approximation, we have

$$p_{\mathrm{DQ}}(x; Q_1, Q_2) \approx n_{\mathrm{DQ}}(x) \cdot p_{\mathrm{NDQ}}(x; Q_2), \quad x \neq 0. \tag{14}$$

i.e., we can approximate the distribution of the quantized DCT coefficients in the presence of A-DJPG compression by multiplying the distribution of the single compressed DCT coefficients by a periodic function $n_{\mathrm{DQ}}(x)$.

The above model can be further modified to take into account the effects of R/T errors. According to (3), R/T errors will cause every bin of $p_0(u)$ to spread over the adjacent bins proportionally to σ_e, where σ_e is the standard deviation of the R/T error. Hence, after quantization by Q_2, such a spread will be proportional to σ_e/Q_2. As a first approximation, this can be modeled by convolving $n_{\mathrm{DQ}}(x)$ with a Gaussian kernel having standard deviation σ_e/Q_2.

Besides these effects, truncation often introduces a bias on R/T errors, which will affect the statistics of the DC coefficients. If the bias is equal to μ_e, the relationship between the unquantized DC coefficient u and the doubly quantized DC coefficient x becomes $x = \left\lfloor \left(\left\lfloor \frac{u}{Q_1} \right\rceil Q_1 + \mu_e \right) \frac{1}{Q_2} \right\rfloor$. Hence, only in the case of DC coefficients, the function $n_{\mathrm{DQ}}(x)$ must be conveniently modified by redefining $L(x) = Q_1 \left(\left\lceil \frac{Q_2}{Q_1} \left(x - \frac{\mu_e}{Q_2} - \frac{1}{2} \right) \right\rceil - \frac{1}{2} \right)$, and $R(x) = Q_1 \left(\left\lfloor \frac{Q_2}{Q_1} \left(x - \frac{\mu_e}{Q_2} + \frac{1}{2} \right) \right\rfloor + \frac{1}{2} \right)$.

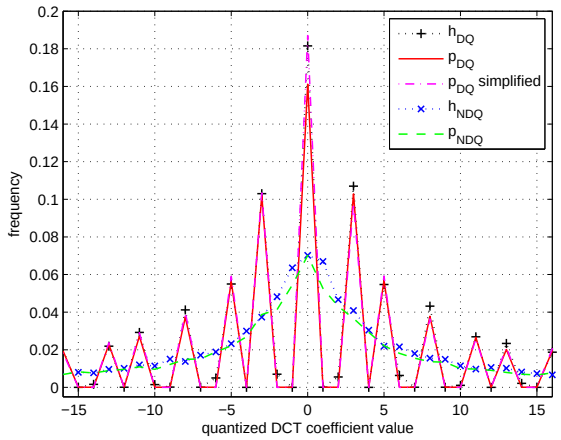


Fig. 1. Example of A-DJPG compression model: h_{DQ} and h_{NDQ} denote the histograms of quantized DCT coefficients of an A-DJPG compressed and a singly compressed image, respectively, for a single DCT frequency. The distributions obtained according to equations (3), (14), and (11) are in good agreement with the data.

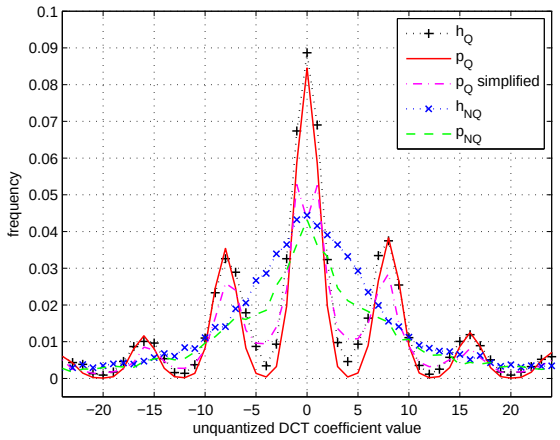


Fig. 2. Example of NA-DJPG compression model: h_{Q} and h_{NQ} denote the histograms of unquantized DCT coefficients of a NA-DJPG compressed and a singly compressed image, respectively, for a single DCT frequency. The distributions obtained according to equations (9), (16), and (6) are in good agreement with the data.

An analogous simplification can also be applied in the case of NA-DJPG. Indeed, by making similar assumptions regarding the smoothness of $p_0(u)$, we have

$$p_1(x) \approx \begin{cases} Q_1 p_0(x) & x = kQ_1 \\ 0 & \text{elsewhere} \end{cases} \tag{15}$$

Hence, if we assume that the JPEG approximation error due to the last compression is smaller than Q_1, and thanks to (11), we have that (9) can be simplified to

$$p_{\mathrm{Q}}(x; Q_1) \approx n_{\mathrm{Q}}(x) \cdot p_{\mathrm{NQ}}(x), \quad x \neq 0. \tag{16}$$

where $n_{\mathrm{Q}}(x) = n_{\mathrm{Q},0}(x) * g_{\mathrm{Q}}(x)$ and

$$n_{\mathrm{Q},0}(x) \triangleq \begin{cases} Q_1 & x = kQ_1 \\ 0 & \text{elsewhere} \end{cases} \tag{17}$$

Since the above approximations may not be accurate when $x = 0$, in practice we choose to empirically correct the function $n_{\mathrm{Q}}(x)$ by setting $n'_{\mathrm{Q}}(0) = n_{\mathrm{Q}}(0)^{1-R_Z}$, where R_Z is the percentage of DCT coefficients equal to zero. The rationale of the above approach is that the more DCT coefficients are equal to zero, the less informative is $n_{\mathrm{Q}}(x)$ when $x = 0$.

As an illustrative example, in Fig. 1 the models proposed in (3), (14), and (11) are compared with the histograms of quantized DCT coefficients of a A-DJPG compressed and a singly compressed image, whereas in Fig. 2 the models proposed in (9), (16), and (6) are compared with the histograms of unquantized DCT coefficients of a NA-DJPG compressed and a singly compressed image, considering a single DCT frequency in both cases. The figures show that there is a good agreement between the proposed models and the real distributions and that the distributions of singly compressed and doubly compressed DCT coefficients can be effectively separated.

IV. Double JPEG Localization Algorithm

In this section, we will describe the proposed algorithm for localizing double JPEG artifacts. Since A-DJPG and NA-DJPG artifacts are characterized by different statistical models defined on different domains – quantized DCT coefficients given by (2) for A-DJPG, unquantized DCT coefficients given by (7) for NA-DJPG – we will actually have two similar algorithms specialized on the respective type of double JPEG compression. A pseudo-code summarizing the proposed algorithms can be found in Algorithm 2 and Algorithm 3.

A. Likelihood Map

In the following, we will assume that for each element x of an image, being x either a pixel value, a DCT coefficient, or the value of any other kind of representation, we know both the probability distributions of x conditional to the hypothesis of being doubly compressed, i.e., $p(x|\mathcal{H}_1)$, and the probability distributions of x conditional to the hypothesis of being singly compressed, i.e., $p(x|\mathcal{H}_0)$.

When dealing with double JPEG compression traces, the above conditional distributions are given by either (3) and (6) or (9) and (11), according to whether we are considering A-DJPG or NA-DJPG artifacts: in the case of A-DJPG artifacts we have $p(x|\mathcal{H}_1) = p_{\mathrm{DQ}}(x; Q_1, Q_2)$ and $p(x|\mathcal{H}_0) = p_{\mathrm{NDQ}}(x; Q_2)$, whereas in the case of NA-DJPG artifacts we have $p(x|\mathcal{H}_1) = p_{\mathrm{Q}}(x; Q_1)$ and $p(x|\mathcal{H}_0) = p_{\mathrm{NQ}}(x)$.

Given $p(x|\mathcal{H}_1)$ and $p(x|\mathcal{H}_0)$, an element x can be classified as belonging to one of the two models according to the value of the likelihood ratio

$$\mathcal{L}(x) = \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)}. \tag{18}$$

As a natural choice, one could assume the element x to be doubly compressed if $\mathcal{L}(x) > 1$, i.e., if $p(x|\mathcal{H}_1) > p(x|\mathcal{H}_0)$. In practice, the likelihood ratio is compared with an appropriate threshold, defined in such a way that the detector fulfills some

optimality condition (e.g., a given probability of false alarm). It is worth noting that the above approach can be easily extended to any trace of tampering, even to traces that are not left by JPEG recompression, provided that the appropriate probability models for both $p(x|\mathcal{H}_0)$ and $p(x|\mathcal{H}_1)$ are defined.

In the case of JPEG images, an element is a DCT coefficient corresponding to a 8×8 block. If multiple DCT coefficients within the same block are considered, by assuming that they are independently distributed we can express the likelihood ratio corresponding to the block at position (i, j) as

$$\mathcal{L}(i,j) = \prod_k \mathcal{L}(x_k(i,j)) \tag{19}$$

where $x_k(i,j)$ denotes the kth DCT coefficient within the block at position (i,j)[3]. Such values form a likelihood map of the JPEG image with resolution 8×8 pixels, which can be used to localize possibly forged regions within the image. Since the areas of interest are usually connected regions with area greater than 8×8 pixels, the likelihood map can be further processed by cumulating the likelihoods on a local window, by assuming that if neighboring blocks are doubly compressed, the likelihood that the reference block is also doubly compressed will increase.

1) Simplified Map: By using the simplified models described in Section III-D, it is possible to approximate the likelihood ratio as $\mathcal{L}(x) = n(x)$, where the form of $n(x)$ depends from the kind of artifacts considered. The likelihood map obtained using such simplifications can be expressed as

$$\mathcal{L}(i,j) \approx \prod_k n(x_k(i,j)) \tag{20}$$

and depends only on compression parameters, i.e., Q_1, Q_2, having removed any dependencies from $p_0(u)$. Hence, provided that the estimated primary compression parameters are the correct ones, the simplified likelihood map will be less dependent from the image content.

B. Estimation of Model Parameters

For the localization of forgeries, the estimation of some parameters of the models described in Section III is required. Among these parameters, $p_0(u)$, Q_2, μ_e, and σ_e are common to both $p(x|\mathcal{H}_1)$ and $p(x|\mathcal{H}_0)$, whereas Q_1 is required only to characterize the distribution of doubly quantized coefficients. Moreover, in the case of NA-DJPG compression we should determine the shift (r, c) between the first compression and the last compression in order to compute the unquantized DCT coefficients as in (8).

As to Q_2, we will assume that it is available from the JPEG image header. As to the other parameters, they can be estimated according to the following procedures.

1) Estimation of Q_1: the estimation of the quantization step of the primary compression is crucial for the correct modeling of doubly compressed regions. When dealing with a possibly forged image, usually there is no prior knowledge regarding the location of such regions. In general, the distribution of the DCT coefficients of a tampered image can thus be modeled as a mixture of the two hypotheses $p(x|\mathcal{H}_1)$ and $p(x|\mathcal{H}_0)$, i.e.,

$$p(x; Q_1, \alpha) = \alpha \cdot p(x|\mathcal{H}_0) + (1-\alpha) \cdot p(x|\mathcal{H}_1; Q_1) \tag{21}$$

where α is the mixture parameter and we have highlighted the dependence of $p(x|\mathcal{H}_1)$ from Q_1.

Based on the above model, the maximum likelihood estimate of Q_1 can be obtained as

$$\hat{Q}_1 = \arg\max_{Q_1} \prod_x [\alpha(Q_1) p(x|\mathcal{H}_0) + (1-\alpha(Q_1)) p(x|\mathcal{H}_1; Q_1)] \tag{22}$$

where $\alpha(Q_1)$ is the optimal mixture parameter for a given Q_1. For each Q_1, the optimal mixture parameter can be estimated using the expectation-maximization (EM) algorithm [19]. Since Q_1 is a discrete parameter with a limited set of possible values, the idea is to run in parallel the EM algorithm over a set of candidate Q_1 values and then to choose the Q_1 maximizing the likelihood function according to (22). The proposed implementation of the EM algorithm is described in Algorithm 1. In order to estimate the complete quantization matrix, the above maximization problem has to be separately solved for each of the 64 DCT coefficients within a block.

Algorithm 1 Pseudocode of the EM algorithm for estimating the quantization step of the primary compression Q_1. N is the number of DCT coefficients.

input $p(x|\mathcal{H}_0)$, $p(x|\mathcal{H}_1; \tilde{Q}_m)$, $\tilde{Q}_m$, $m = 1, \ldots, M$, α_0
set $\tilde{\alpha}_m = \alpha_0$, $m = 1, \ldots, M$
set $L_{max} = -\infty$
repeat
  for $m = 1 \rightarrow M$ **do**
    {Expectation step}
    for all x **do**
      $\beta(x) = \frac{\tilde{\alpha}_m p(x|\mathcal{H}_0)}{\tilde{\alpha}_m p(x|\mathcal{H}_0) + (1-\tilde{\alpha}_m) p(x|\mathcal{H}_1; \tilde{Q}_m)}$
    end for
    {Maximization step}
    $\tilde{\alpha}_m = \frac{1}{N} \sum_x \beta(x)$
    $L = \prod_x (\tilde{\alpha}_m p(x|\mathcal{H}_0) + (1-\tilde{\alpha}_m) p(x|\mathcal{H}_1; \tilde{Q}_m))$
    if $L > L_{max}$ **then**
      $L_{max} = L$
      $Q_1 = \tilde{Q}_m$
    end if
  end for
until convergence
return Q_1, L_{max}

2) Estimation of (r, c): the shift (r, c) in the case of NA-DJPG can be estimated using a similar technique. The idea is to use the previously proposed EM algorithm to solve the maximization in (22) for every possible grid shift, excepting $(r, c) = (0, 0)$, and then to choose the (r, c) values for which the likelihood function is globally maximized. In order to limit the complexity, for each grid shift we consider the DC DCT coefficients only, following an approach similar to [20]. The proposed technique is described in Algorithm 3.

[3]With a slight abuse of notation, we use the same symbol $\mathcal{L}(x)$ even if for different k we have different likelihood functions. The same convention is used in (20) and (28) when referring to $n(x)$ and $h(x)$, respectively.

3) Estimation of $p_0(u)$: obtaining a reliable estimate of the distribution of the unquantized DCT coefficients from the corresponding quantized or doubly quantized coefficients may be a difficult task. In [21], it is proposed to use a parametric model based on the hypothesis that AC coefficients are Laplacian distributed. However, previous studies show that a simple Laplacian distribution may not be sufficiently accurate [22]. Moreover, such a hypothesis can not be applied to DC coefficients.

In this work, we resort to a non parametric modeling of $p_0(u)$ based on the histogram of the observed DC coefficients. It has already been observed [18] that the DCT coefficients obtained by applying a slight shift to the grid used for computing the block DCT usually do not exhibit quantization artifacts. Hence, we propose to approximate the distribution of the unquantized DCT coefficients using the histogram of the DCT coefficients of the decompressed image computed after the DCT grid is suitably shifted with respect to the upper left corner. In the case that we are detecting the presence of A-DJPG artifacts, we will use a shift $(1,1)$. In the case of detection of NA-DJPG artifacts, we will use a shift of ± 1 with respect to the estimated shift (r,c) of the primary compression, where the sign of the increment is chosen so as to keep the shift values between 0 and 7 and to avoid the case $(0,0)$.

Given the histogram $h(u)$ of the DCT coefficients computed with the above procedure, the desired probability distribution is estimated as

$$p_0(u) = \frac{h(u)+1}{N+N_{bin}} \tag{23}$$

where N is the number of DCT coefficients and N_{bin} is the number of bins of the histogram. The above estimate uses Laplace's rule of succession [23] to avoid assigning a zero probability when $h(u)=0$.

4) Estimation of μ_e and σ_e: the true values of both μ_e and σ_e should be estimated by relying on the primary JPEG compression, which in general is not available when observing the tampered image. In practice, we found that they can be well approximated by measuring the R/T error on the tampered image. The rationale is that both μ_e and σ_e are mainly determined by the coarse-grained statistics of the image content, which usually are little affected by tampering.

According to equation (1), we can model the tampered image as $\mathbf{I}_2 = \mathbf{D}_{00}^{-1}\mathcal{D}(\mathcal{Q}(\mathbf{D}_{00}\mathbf{I_1})) + \mathbf{E}_2$. Hence, given as input the quantized DCT coefficients of the observed image $\mathbf{C}_2$ we can think to compute the term $\mathbf{E}_2$ by reconstructing the tampered image with infinite precision as $\mathbf{D}_{00}^{-1}\mathcal{D}(\mathbf{C}_2)$, which can be approximated by using floating point arithmetic, and taking the difference with the image $\mathbf{I}_2$ which is obtained by rounding and truncating to 8 bit precision the floating point values. The values of μ_e and σ_e relative to the kth DCT coefficient within a block can be estimated as

$$\mu_e = E[(\mathbf{D}_{00}\mathbf{E}_2)_k] \tag{24}$$

and

$$\sigma_e^2 = Var[(\mathbf{D}_{00}\mathbf{E}_2)_k] \tag{25}$$

where $()_k$ indicates that we are taking the kth DCT coefficient from each 8×8 block. Usually, μ_e is computed only for the DC coefficient, since for the AC coefficients we have $\mu_e = 0$.

Algorithm 2 Pseudocode of the algorithm for generating the likelihood map in the A-DJPG case.

```
input C_2, Q_2, N_coeff, model, α_0
set L(i,j) = 1
for k = 1 → N_coeff do
  input Q̃_m, m = 1..., M_k
  set x = (C_2)_k
  estimate p_0(u), μ_e, σ_e^2
  for all x do
    p(x|H_0) = p_NDQ(x; Q_2)
    p(x|H_1; Q̃_m) = p_DQ(x; Q̃_m, Q_2), m = 1..., M_k
  end for
  Q_1 ⇐ Algorithm 1
  if model = 0 then
    L(i,j) = L(i,j) · p(x(i,j)|H_1;Q_1) / p(x(i,j)|H_0)
  else
    L(i,j) = L(i,j) · n_DQ(x(i,j); Q_1)
  end if
end for
return L(i,j)
```

C. Method of [7]

In the case of A-DJPG artifacts, the authors of [7] propose an alternative method for deriving the posterior probability of a block being doubly compressed. Starting from the observation that $n_{\mathrm{DQ}}(x)$ is a periodic function with period $P = Q_1/\gcd(Q_1,Q_2)$, they propose to estimate the conditional probabilities as

$$p(x|\mathcal{H}_0) \propto 1/P \tag{26}$$

and

$$p(x|\mathcal{H}_1) \propto h(x)/\sum_{t=0}^{P-1} h(\lfloor x/P \rfloor P + t) \tag{27}$$

where $h(x)$ is the histogram of the DCT coefficients and the period P can be estimated from $h(x)$ (see [7] for the details). The above estimates stem from the observation that in the absence of DQ artifacts x will be approximately uniformly distributed over a histogram period, whereas in the presence of DQ artifacts x will be concentrated around some histogram peaks. According to the above model, the likelihood map can be approximated as

$$\mathcal{L}(i,j) \approx \prod_k \frac{Ph(x_k(i,j))}{\sum_{t=0}^{P-1} h(\lfloor x_k(i,j)/P \rfloor P + t)}. \tag{28}$$

V. Experimental Results

In this section we firstly describe the experimental methodology we followed in order to evaluate the performance of the proposed forgery detectors; then, experimental results for both A-DJPG and NA-DJPG are shown, considering different scenarios, and the use of both standard likelihood maps and simplified maps. As to A-DJPG, results are also compared with the method of [7].

Algorithm 3 Pseudocode of the algorithm for generating the likelihood map in the NA-DJPG case.

input $\mathbf{I}_2, N_{coeff}, model, \alpha_0$
set $\mathcal{L}(i,j) = 1$
estimate μ_e, σ_e^2
{Estimate grid shift}
set $L_{max} = -\infty$
for all (r', c') **do**
  input $\tilde{Q}_m, m = 1 \ldots, M_1$
  set $x = (\mathbf{D}_{r'c'}\mathbf{I}_2)_1$
  estimate $p_0(u)$
  for all x **do**
    $p(x|\mathcal{H}_0) = p_{\text{NQ}}(x)$
    $p(x|\mathcal{H}_1; \tilde{Q}_m) = p_{\text{Q}}(x; \tilde{Q}_m), m = 1 \ldots, M_1$
  end for
  $L \Leftarrow$ Algorithm 1
  if $L > L_{max}$ **then**
    $(r, c) = (r', c')$
    $L_{max} = L$
  end if
end for
for $k = 1 \rightarrow N_{coeff}$ **do**
  input $\tilde{Q}_m, m = 1 \ldots, M_k$
  set $x = (\mathbf{D}_{rc}\mathbf{I}_2)_k$
  estimate $p_0(u), \mu_e, \sigma_e^2$
  for all x **do**
    $p(x|\mathcal{H}_0) = p_{\text{NQ}}(x)$
    $p(x|\mathcal{H}_1; \tilde{Q}_m) = p_{\text{Q}}(x; \tilde{Q}_m), m = 1 \ldots, M_k$
  end for
  $Q_1 \Leftarrow$ Algorithm 1
  if $model = 0$ **then**
    $\mathcal{L}(i,j) = \mathcal{L}(i,j) \cdot \frac{p(x(i,j)|\mathcal{H}_1; Q_1)}{p(x(i,j)|\mathcal{H}_0)}$
  else
    $\mathcal{L}(i,j) = \mathcal{L}(i,j) \cdot n_{\text{Q}}(x(i,j); Q_1)$
  end if
end for
return $\mathcal{L}(i,j)$

A. Experimental Methodology

For the experimental validation of the proposed work, we have built an image dataset composed by 100 non-compressed TIFF images, having heterogeneous contents, coming from three different digital cameras (namely Nikon D90, Canon EOS 450D, Canon EOS 5D) and each acquired at its highest resolution; each test has been performed by cropping a central portion with size 1024×1024[4].

Starting from this dataset, we have created manipulated images exhibiting either A-DJPG or NA-DJPG artifacts, according to different scenarios. As to the simulation of A-DJPG artifacts, each original image is JPEG compressed with a given quality factor QF_1 (using the Matlab function `imwrite`); the area that should not exhibit A-DJPG artifacts is replaced with the corresponding area from the original TIFF image; finally, the overall "manipulated" image is JPEG compressed (again with Matlab) with another given quality factor QF_2. As to the simulation of NA-DJPG artifacts, we repeat the above steps, except that the primary JPEG compressed image is randomly cropped by removing a number of rows and columns between 0 and 7 before inserting the uncompressed area. In all the datasets, QF_1 and QF_2 are taken from the sets $[50, 55, \ldots 95]$ and $[50, 55, \ldots 100]$, respectively, achieving 110 possible combinations of (QF_1, QF_2) for each of the 100 tampered images.

We considered three different scenarios corresponding to different percentages of doubly compressed blocks. In the first scenario, double compression is present in all the image but the central portion of size 256×256, corresponding to $15/16$ of the image area; in the second scenario, double compression is present only in the right half of the image, corresponding to $1/2$ of the image area; in the third scenario double compression is present only in the central portion of size 256×256, corresponding to $1/16$ of the image area. In the following, the above scenarios are referred to as $15/16$ scenario, $1/2$ scenario, and $1/16$ scenario, respectively.

As to the parameters of the proposed algorithm, in all experiments we used $\alpha_0 = 0.95$, while the possible Q_1 values for each DCT frequency have been chosen in the set $\{1, \ldots, Q^{(50)}\}$, where $Q^{(50)}$ denotes, for each DCT frequency, the quantization step corresponding to $QF_1 = 50$. Moreover, the logarithm of likelihood maps has been processed by a 3×3 mean filter in order to cumulate neighboring likelihood values.

The selection of a proper performance metric is fundamental for comparing the different methods. The considered algorithms provide as output, for each analyzed image, a likelihood map that represents the probability of each 8×8 block to be doubly compressed (i.e. for each 1024×1024 image a 128×128 likelihood map is given). After a thresholding step, a binary detection map is achieved, that locates which are the blocks detected as recompressed. In our experiments, we assume to know for each analyzed image the position of doubly compressed areas, so that it is possible to associate to any manipulated image a corresponding 128×128 binary mask indicating the positions of doubly compressed blocks.

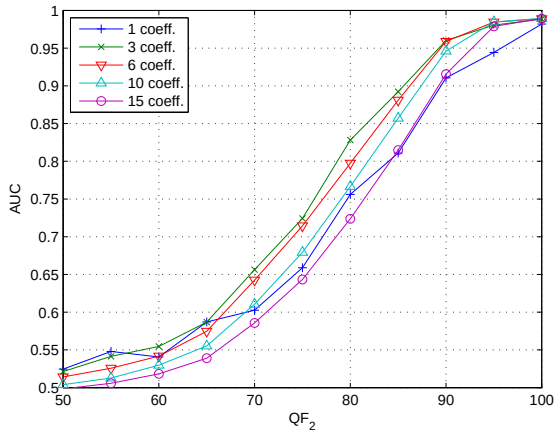
A comparison between the algorithm output detection map and the known tampering mask will allow to estimate the error rates of the forensic schemes, measured as false alarm probability P_{fa} and missed detection probability P_{md}. These two probabilities can be computed by measuring the following parameters: n_{EDQ}: number of blocks erroneously detected as doubly compressed; n_{ESQ}: number of blocks erroneously detected as singly compressed; n_{I}: number of blocks in the image; n_{M}: number of doubly compressed blocks. Starting from these figures, the error probabilities are given by:

$$P_{fa} = \frac{n_{\text{EDQ}}}{n_{\text{I}} - n_{\text{M}}} \qquad P_{md} = \frac{n_{\text{ESQ}}}{n_{\text{M}}}$$

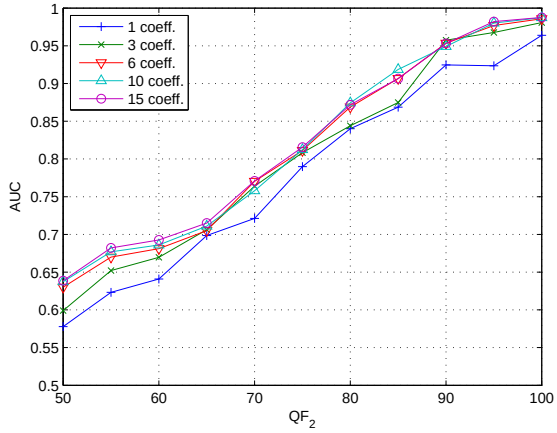
and the correct detection probability is: $P_d = 1 - P_{md}$.

For depicting the tradeoff between the correct detection rate P_d and the false alarm rate P_{fa} the receiver operating characteristic (ROC) curve is considered. Since the ROC curve is a two dimensional plot of P_d versus P_{fa} as the decision threshold of the detector is varied, we adopt the area under

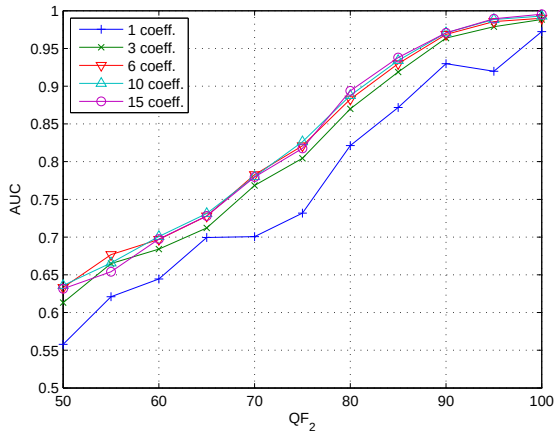
[4]In the case of NA-DJPG, the size is increased by 7 pixels in either dimension so that we can still have a 1024×1024 image after random cropping.



(a)

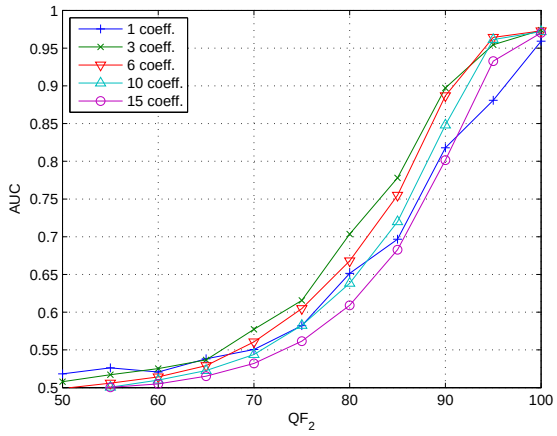


(b)

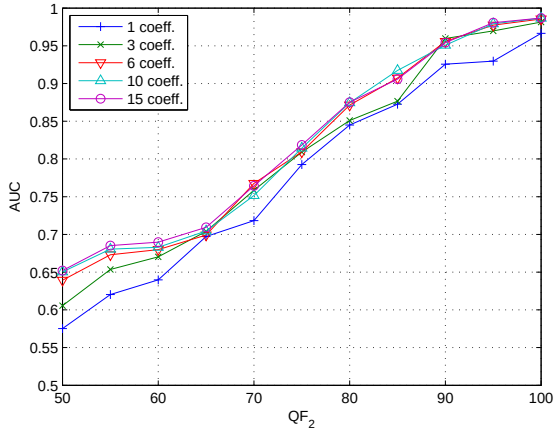


(c)

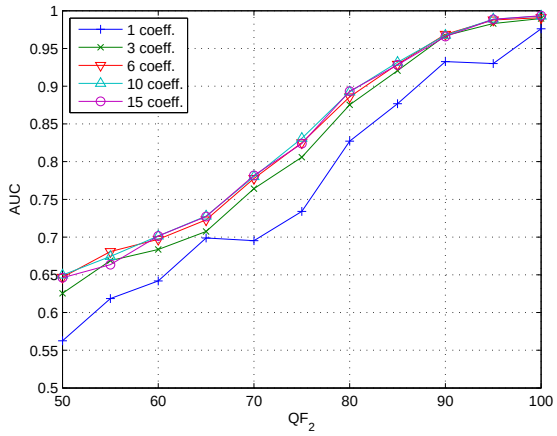
Fig. 3. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of A-DJPG compression and $15/16$ scenario: (a) algorithm in [7]; (b) proposed algorithm with standard map; (c) proposed algorithm with simplified map.



(a)



(b)



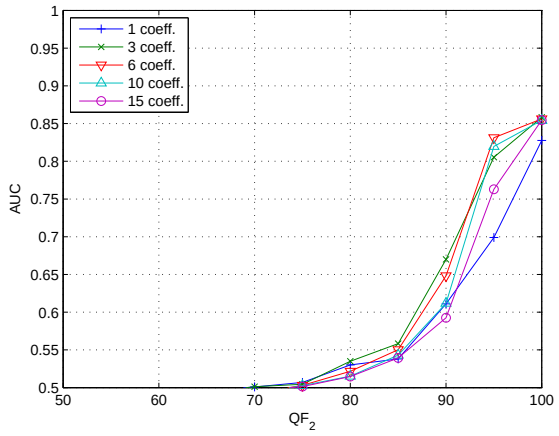
(c)

Fig. 4. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of A-DJPG compression and $1/2$ scenario: (a) algorithm in [7]; (b) proposed algorithm with standard map; (c) proposed algorithm with simplified map.

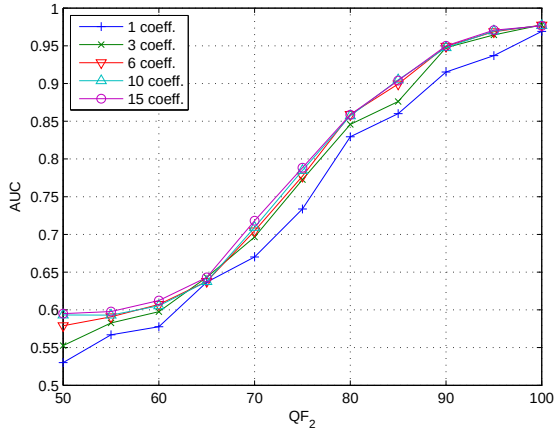
the ROC curve (AUC) in order to summarize the performance with a unique scalar value representing the general behavior of the detector. It is known that AUC should assume values between 0.5 and 1 for realistic and effective, i.e. no random, detectors.

B. A-DJPG

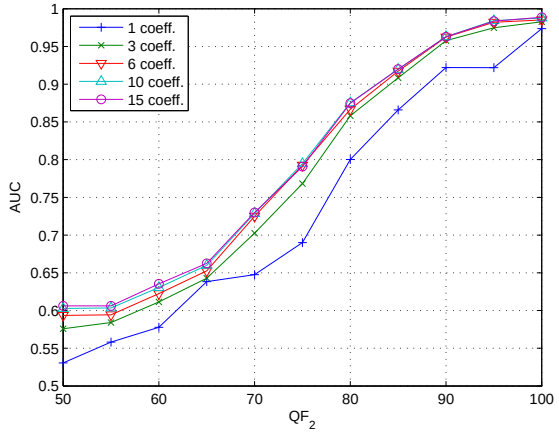
In the case of A-DJPG compression, we have compared the AUC values obtained using the standard map in (19) and the simplified map in (20) with those obtained using the algorithm in [7]. In all cases, likelihood maps are obtained by cumulating



(a)



(b)



(c)

Fig. 5. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of A-DJPG compression and $1/16$ scenario: (a) algorithm in [7]; (b) proposed algorithm with standard map; (c) proposed algorithm with simplified map.

a different number of DCT coefficients for each block, starting from the DC coefficient and scanning the coefficients in zig-zag order.

The AUC values achieved for different QF_2 values in the three scenarios, averaged over all QF_1 values, are shown in Figs. 3-5. As can be seen, the AUC values achieved by the proposed approach are higher than those obtained with the method of [7], especially for lower QF_2 and in the $1/2$ and $1/16$ scenarios. It is worth noting that the proposed algorithm is able to detect traces of A-DJPG compression even in the $1/16$ scenario, achieving a performance only slightly lower than that obtained in the other two scenarios, whereas the method of [7] is almost useless in the $1/16$ scenario.

When using the proposed approach, the detection performance tends to increase with the number of DCT coefficients used to generate the likelihood map, even though 6 coefficients are usually enough to obtain the best performance. Conversely, with the algorithm in [7] we have the best performance when using only few DCT coefficients: when more than 6 coefficients are used, the AUC values tend to decrease, highlighting the fact that the information contained in the higher frequencies can not be reliably exploited by the method of [7].

In order to assess the effects of different QF_1 values, the AUC values obtained for different combinations of (QF_1, QF_2) in the $1/2$ scenario, using the first 6 DCT coefficients to compute the likelihood map, are reported in Tables II-IV. For ease of reading, for each combination of (QF_1, QF_2) the highest AUC value among the three considered approaches is highlighted in bold. As can be seen, the proposed approach is always the best one, being able to achieve AUC greater the 0.5 even when $QF_2 < QF_1$. The standard map and

TABLE II
AUC ACHIEVED BY THE ALGORITHM IN [7] IN THE CASE OF A-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	0.506	0.579	0.709	0.845	0.978	0.994
60	0.492	0.503	0.635	0.814	0.969	0.993
70	0.484	0.509	**0.502**	0.686	0.943	0.992
80	0.526	0.485	0.512	0.501	0.897	0.981
90	0.505	0.499	0.502	0.500	0.499	0.948

TABLE III
AUC ACHIEVED BY THE PROPOSED ALGORITHM USING THE STANDARD MAP IN THE CASE OF A-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	0.504	0.942	**0.995**	**0.999**	**0.999**	**0.999**
60	**0.745**	0.499	0.949	**0.997**	0.998	0.999
70	0.783	0.810	0.502	0.986	0.998	0.998
80	0.602	0.592	0.712	**0.501**	0.989	0.990
90	0.529	0.515	0.534	0.584	0.499	0.976

TABLE IV
AUC ACHIEVED BY THE PROPOSED ALGORITHM USING THE SIMPLIFIED MAP IN THE CASE OF A-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	**0.509**	**0.948**	0.995	0.998	0.999	0.998
60	0.734	**0.503**	**0.957**	0.997	**0.999**	**0.999**
70	**0.784**	**0.828**	0.496	**0.988**	**0.999**	**0.999**
80	**0.659**	**0.640**	**0.778**	0.499	**0.995**	**0.993**
90	**0.540**	**0.530**	**0.549**	**0.621**	**0.505**	**0.987**

TABLE V
PERCENTAGE (%) OF ERRONEOUSLY ESTIMATED Q_1 VALUES IN THE CASE OF A-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	100	8	7	5	2	1
60	25	100	18	8	2	1
70	32	25	100	7	2	0
80	40	46	22	100	1	0
90	99	99	98	79	98	1

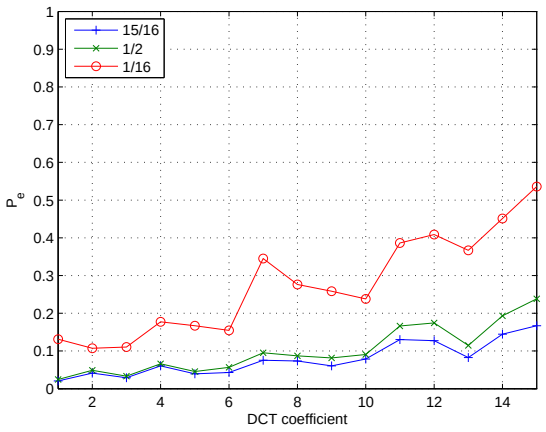


Fig. 6. Percentage of erroneously estimated Q_1 values for each DCT coefficient in the A-DJPG case. The index on the x-axis corresponds to zig-zag ordering. Values are averaged over all (QF_1, QF_2) such that $QF_2 > QF_1$.

the simplified map achieve very similar results: the standard approach seems slightly more robust when $QF_2 \gg QF_1$, whereas the simplified map is usually better when QF_2 is similar to QF_1.

The influence of the reliability of the estimated Q_1 values on the above results can be assessed from Table V, where the percentage of erroneously estimated Q_1 values in the $1/2$ scenario is shown. As can be seen, there is a strong dependence between the detection performance and the estimation accuracy, suggesting that the main reason of detection errors is the inaccurate estimation of the parameters of the underlying probability models. From Fig. 6, where the above percentages are averaged over all (QF_1, QF_2) such that $QF_2 > QF_1$, it is also evident that the estimated Q_1 values are more reliable for lower DCT frequencies and in scenarios where we have a higher percentage of doubly compressed blocks.

C. NA-DJPG

In the case of NA-DJPG compression, we compare only the AUC values obtained using the standard map in (19) and the simplified map in (20), since, to the best of our knowledge, these are the first methods that permit to localize possibly forged areas by relying on non-aligned double JPEG compression. As with the A-DJPG case, likelihood maps are obtained by cumulating different number of DCT coefficients for each block.

The AUC values achieved for different QF_2 values in the three scenarios, averaged over all QF_1 values, are shown in

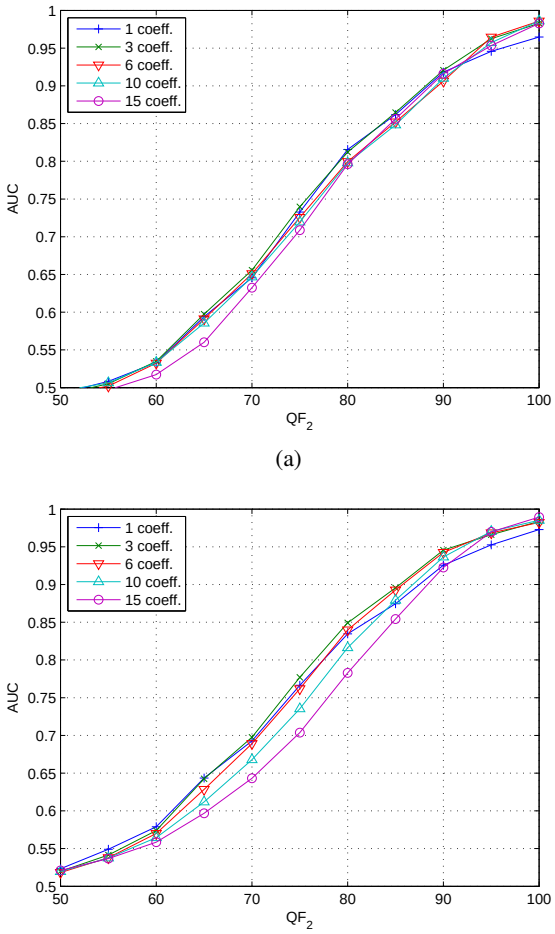
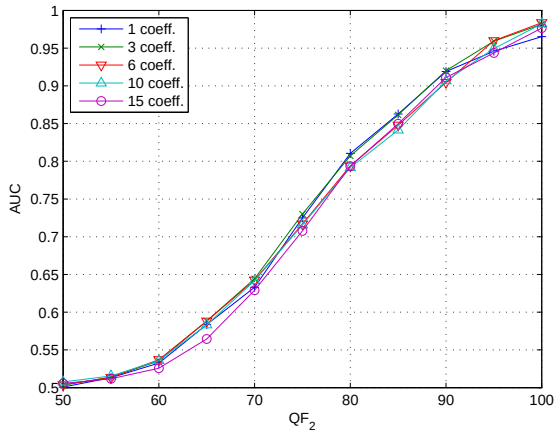


Fig. 7. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of NA-DJPG compression and $15/16$ scenario: (a) proposed algorithm with standard map; (b) proposed algorithm with simplified map.

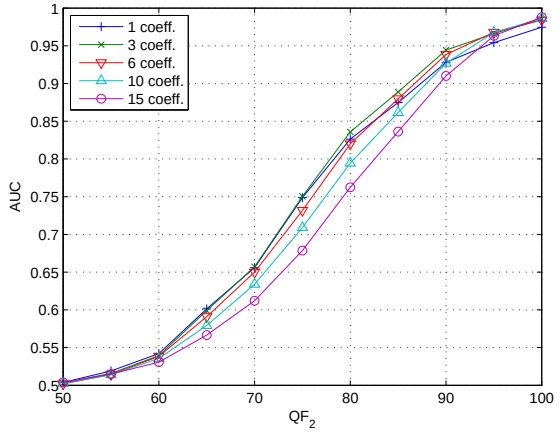
Figs. 7-9. In both $15/16$ and $1/2$ scenarios, when QF_2 is sufficiently high we can effectively localize traces of tampering: for $QF_2 > 80$, the proposed algorithm detects more than 85% of the regions presenting NA-DJPG artifacts. However, the NA-DJPG features are in general less reliable than the A-DJPG ones. In the $1/16$ scenario, the performance of forgery localization is much lower than under the other two scenarios, allowing to localize some traces of double compression only when QF_2 is very high (> 90).

When comparing the two approaches, the simplified map appears more robust than the standard map for lower QF_2 values. As to the effects of the cumulation of different DCT coefficients, the best results are obtained by considering 3-6 coefficients with the simplified map: when considering a higher number of coefficients the AUC values decrease, suggesting that NA-DJPG artifacts can not be reliably detected at the higher frequencies.

In order to assess the effects of different QF_1 values, the AUC values obtained for different combinations of (QF_1, QF_2) in the $1/2$ scenario, using the first 6 DCT coefficients to compute the likelihood map, are reported in

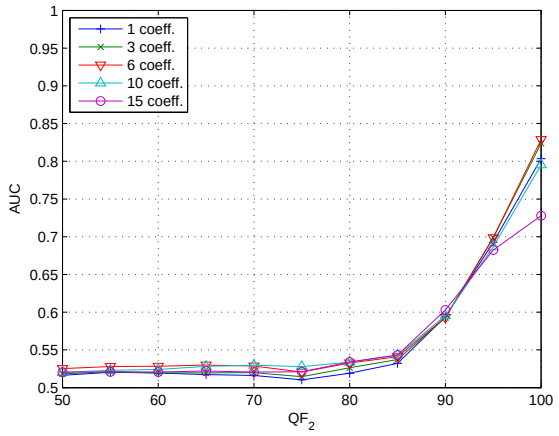


(a)

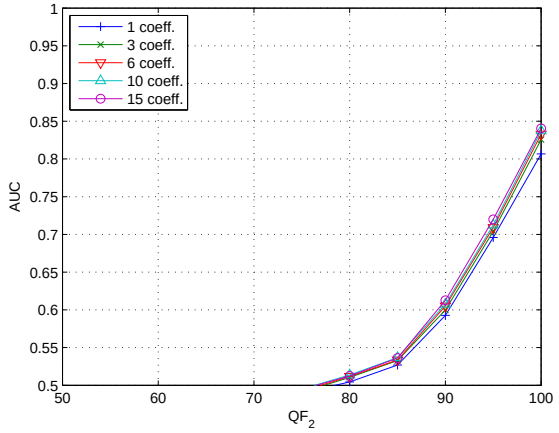


(b)

Fig. 8. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of NA-DJPG compression and $1/2$ scenario: (a) proposed algorithm with standard map; (b) proposed algorithm with simplified map.



(b)



(c)

Fig. 9. AUC achieved for different QF_2, averaged over all QF_1 values, using different numbers of DCT coefficients in the case of NA-DJPG compression and $1/16$ scenario: (a) proposed algorithm with standard map; (b) proposed algorithm with simplified map.

Tables VI-VII. For ease of reading, for each combination of (QF_1, QF_2) the highest AUC value between the two considered approaches is highlighted in bold. Also in this case, the simplified map achieves the best performance when QF_2 is similar to QF_1, whereas the standard map is slightly better when $QF_2 \gg QF_1$. Differently from the A-DJPG case, it is not possible to achieve AUC values significantly greater than 0.5 when $QF_2 < QF_1$. However, it suffices $QF_2 - QF_1 \geq 10$ to achieve an AUC value close to one, which means that forged areas can be localized with great accuracy.

The influence of the reliability of the estimated (r, c) pairs and Q_1 values on the above results can be assessed from Tables VIII and IX, where the percentage of erroneously estimated (r, c) pairs and Q_1 values for $1/2$ and $1/16$ scenarios is shown, respectively. From the above results, it is evident that detection performance is strongly influenced by the ability to estimate the correct grid shift. In Fig. 6, the percentages of erroneously estimated Q_1 values are averaged over all (QF_1, QF_2) such that $QF_2 > QF_1$, showing that the estimated Q_1 values are more reliable for lower DCT frequencies, except in the $1/16$ scenario, where the estimation is usually unreliable.

TABLE VI
AUC ACHIEVED BY THE PROPOSED ALGORITHM USING THE STANDARD MODEL IN THE CASE OF NA-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	0.533	0.706	0.910	**0.992**	**0.998**	**0.999**
60	0.497	0.536	0.790	0.972	0.995	**0.999**
70	**0.499**	**0.498**	0.547	0.899	0.989	0.998
80	**0.501**	**0.502**	**0.496**	0.526	0.935	0.989
90	0.498	**0.498**	**0.495**	0.494	0.493	0.954

TABLE VII
AUC ACHIEVED BY THE PROPOSED ALGORITHM USING THE SIMPLIFIED MODEL IN THE CASE OF NA-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	**0.542**	**0.729**	**0.924**	0.989	0.996	0.998
60	**0.497**	**0.546**	**0.808**	**0.975**	**0.995**	0.999
70	0.493	0.497	**0.557**	**0.920**	**0.993**	**0.999**
80	0.499	0.497	0.496	**0.537**	**0.955**	**0.990**
90	**0.499**	0.497	0.495	**0.496**	**0.500**	**0.972**

TABLE VIII
PERCENTAGE (%) OF ERRONEOUSLY ESTIMATED (r, c) PAIRS AND Q_1 VALUES (BETWEEN PARENTHESES) IN THE CASE OF NA-DJPG COMPRESSION AND $1/2$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	72(41)	39(38)	15(25)	0(18)	0(14)	0(1)
60	99(95)	70(61)	28(41)	3(16)	1(10)	0(0)
70	100(94)	99(96)	59(53)	8(17)	1(7)	0(0)
80	99(100)	98(95)	99(96)	66(53)	3(4)	0(0)
90	96(100)	97(100)	95(100)	98(95)	90(53)	2(3)
100	96(100)	98(100)	99(100)	97(100)	98(100)	98(100)

TABLE IX
PERCENTAGE (%) OF ERRONEOUSLY ESTIMATED (r, c) PAIRS AND Q_1 VALUES (BETWEEN PARENTHESES) IN THE CASE OF NA-DJPG COMPRESSION AND $1/16$ SCENARIO.

QF_1 \ QF_2	50	60	70	80	90	100
50	99(51)	98(79)	95(87)	93(93)	65(77)	19(24)
60	99(95)	99(69)	96(95)	95(96)	82(88)	32(35)
70	99(94)	99(96)	98(65)	96(94)	86(91)	43(46)
80	97(100)	98(95)	99(96)	95(65)	92(94)	61(62)
90	96(100)	95(100)	98(100)	99(94)	98(58)	89(64)
100	97(100)	99(100)	100(100)	96(100)	97(100)	99(100)

D. Examples

The algorithm has also been tested on a set of images representing realistic cases of forgery. In Figure 11 two examples of tampered image are shown: the likelihood maps clearly reveal that the pyramid on the left image exhibits NA-DJPG artifacts, whereas the license plate on the right image is the only area that does not exhibit A-DJPG artifacts, suggesting that both objects are likely forgeries. The maps also show some false alarms in image regions with either low intensity variance, like the sky in the image on the left, or saturated values, like the reflections on the hood of the car in the image on the right. However, these errors can be eliminated by a proper postprocessing on the map, before a decision is made.

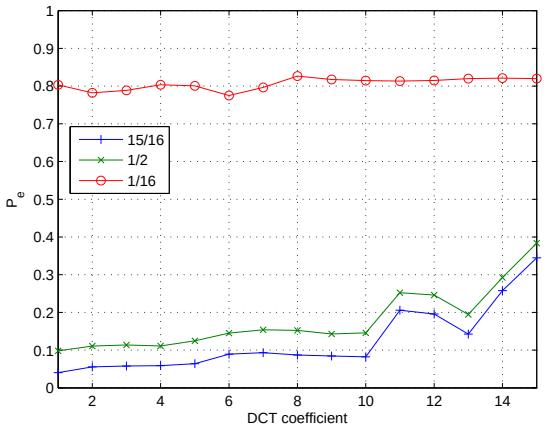


Fig. 10. Percentage of erroneously estimated Q_1 values for each DCT coefficient in the NA-DJPG case. The index on the x-axis corresponds to zig-zag ordering. Values are averaged over all (QF_1, QF_2) such that $QF_2 > QF_1$.

VI. CONCLUSIONS

In this paper, a new forensic algorithm to detect and localize a tampered area into a digital image in the presence of a JPEG recompression has been proposed. The method is based on the derivation of a unified statistical model characterizing the DCT coefficients when an aligned or a non-aligned double JPEG (A-DJPG or NA-DJPG) compression is applied; following an approach similar to the one proposed in [7] and in [8], the statistical model is used for the generation of a likelihood map that shows the probability of each 8×8 image block of being doubly compressed. The validity of the proposed system has been demonstrated by computing the ROC curves and the corresponding AUC values for the double compression detector based on properly thresholding the likelihood map. The effectiveness of the proposed method is also confirmed by tests carried on realistic tampered images.

The results show that the proposed method is able to correctly identify traces of A-DJPG compression unless $QF_2 = QF_1$ or $QF_2 \ll QF_1$, whereas it is able to correctly identify traces of NA-DJPG compression whenever $QF_2 > QF_1$ and there is a sufficient percentage of doubly compressed blocks. In our opinion, the limitations of the proposed system are due to the fact that it is very difficult to separate the distributions of singly compressed and doubly compressed DCT coefficients when $QF_2 < QF_1$. A possible way to overcome this problem could be that of jointly estimate recompression parameters on several DCT frequencies, however at the cost of a greater complexity.

Morever, the proposed approach, like previous methods for detecting double JPEG compression, is no more valid if certain image processing operations, like resizing, are applied between the two compressions. Future research will be devoted to methods that can cope with simple image editing operations. Nevertheless, we can think of particular scenarios in which the present tool is already practical. For example, when an image coming from a particular camera must be manipulated without altering the camera format, as it may happen in a legal case where images supposedly acquired by a given digital camera are examined as evidence.

Even if our results consider only simplified scenarios in which one part of the image is singly compressed, as discussed in Section II our algorithm can also be applied when both parts are doubly compressed but exhibit different kind of artifacts. Tests conducted on a dataset of images where the left half was A-DJPG compressed and the right half was NA-DJPG compressed, not shown here due to lack of space, reported for both detectors the same performance observed in the simplified scenarios. There are also cases in which the proposed approach is not expected to work, for example when both parts present a double JPEG compression with the same grid shift, and that require further investigation.

Finally, it is worth noting that the proposed forgery localization method can be applied also to similar features representing other traces left by common tampering operations, which will be the topic of future work.

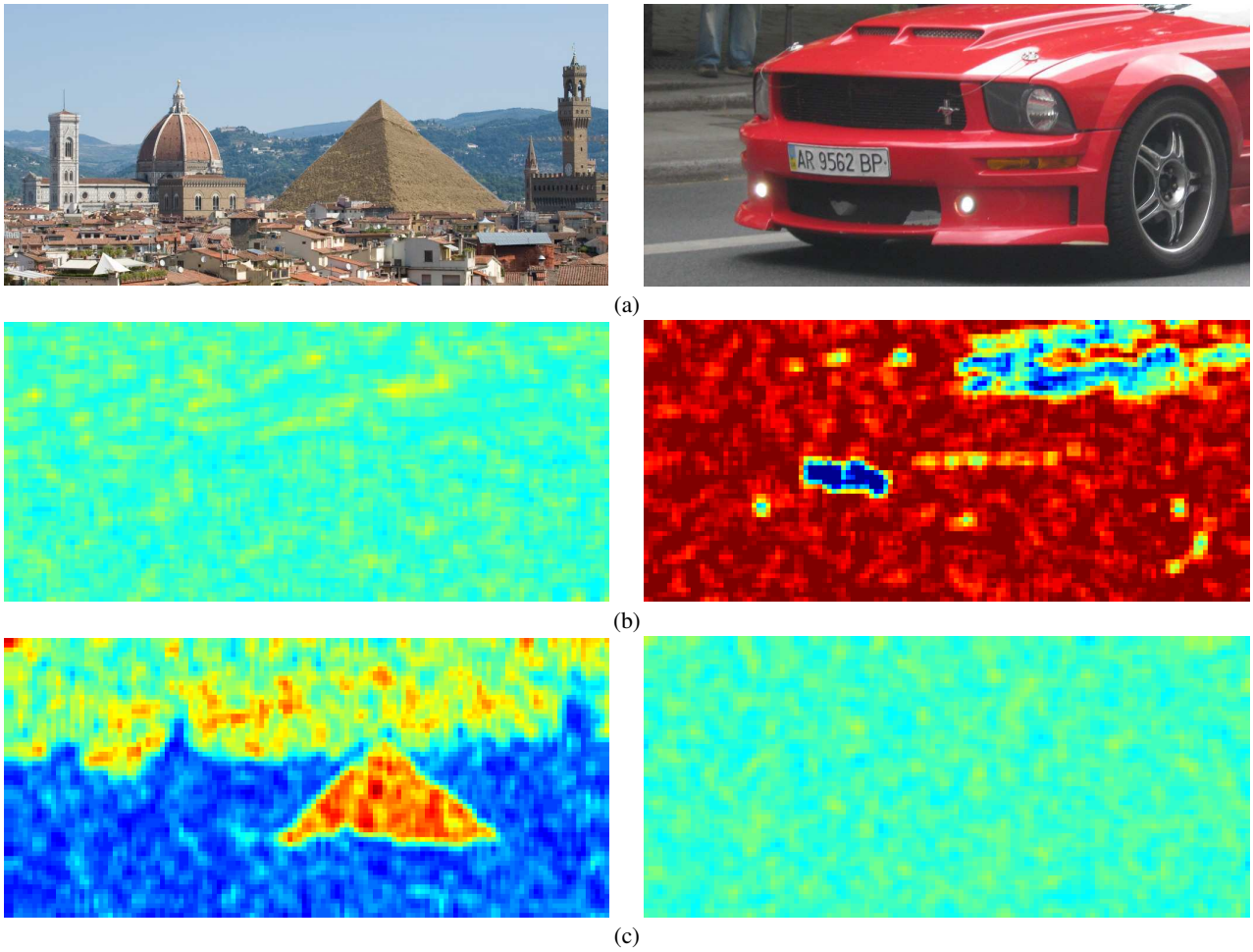


Fig. 11. *Application to realistic forgeries: (a) images under analysis; (b) likelihood maps obtained using the A-DJPG simplified model; (c) likelihood maps obtained using the NA-DJPG simplified model. Red/blue areas correspond to high/low probability of being doubly compressed. On the left side, the proposed algorithm shows that there is a high probability of the pyramid to be doubly compressed according to NA-DJPG model. On the right side, the proposed algorithm shows that the license plate has a high probability of being singly compressed, whereas the rest of the image has a high probability of being doubly compressed according to A-DJPG model. The quality settings are* $QF_1 = 60$, $QF_2 = 95$ *(left side),* $QF_1 = 90$, $QF_2 = 95$ *(right side).*

REFERENCES

[1] H. Farid, "A survey of image forgery detection," *IEEE Signal Processing Mag.*, vol. 2, no. 26, pp. 16–25, 2009.

[2] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *In 6th International Workshop on Information Hiding*. Springer-Verlag, Berlin-Heidelberg, 2004, pp. 128–147.

[3] D. Fu, Y. Q. Shi, and W. Su, "A generalized Benford's law for JPEG coefficients and its applications in image forensics," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX*, P. W. Wong and E. J. Delp, Eds., vol. 6505, San Jose, CA, USA, January 2007, pp. 1L1–1L11.

[4] B. Li, Y. Shi, and J. Huang, "Detecting doubly compressed JPEG images by using mode based first digit features," in *Proc. of IEEE 10th Workshop on Multimedia Signal Processing*, oct. 2008, pp. 730–735.

[5] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Trans. on Information Forensics and Security*, vol. 4, no. 1, pp. 154–160, Mar. 2009.

[6] X. Feng and G. Doërr, "JPEG recompression detection," in *Media Forensics and Security II*, ser. Proceedings of the SPIE, vol. 7541, Feb. 2010, pp. 75 410J–75 410J–12.

[7] Z. Lin, J. He, X. Tang, and C.-K. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognition*, vol. 42, no. 11, pp. 2492–2501, Nov. 2009.

[8] T. Bianchi, A. D. Rosa, and A. Piva, "Improved DCT coefficient analysis for forgery localization in JPEG images," in *Proc. of ICASSP 2011*, May 2011, pp. 2444–2447.

[9] W. Luo, Z. Qu, J. Huang, and G. Qui, "A novel method for detecting cropped and recompressed image block," in *Proc. of ICASSP 2007*, vol. 2, 2007, pp. II–217–II–220.

[10] M. Barni, A. Costanzo, and L. Sabatini, "Identification of cut & paste tampering by means of double-JPEG detection and image segmentation," in *Proc. of ISCAS 2010*, 2010, pp. 1687–1690.

[11] Y.-L. Chen and C.-T. Hsu, "Image tampering detection by blocking periodicity analysis in JPEG compressed images," in *Proc. of IEEE 10th Workshop on Multimedia Signal Processing*, Oct. 2008, pp. 803–808.

[12] T.Bianchi and A.Piva, "Detection of nonaligned double JPEG compression based on integer periodicity maps," *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 2, Apr. 2012.

[13] T. Pevný and J. Fridrich, "Detection of double-compression for applications in steganography," *IEEE Trans. on Information Forensics and Security*, vol. 3, no. 2, pp. 247–258, June 2008.

[14] Z. Fan and R. de Queiroz, "Identification of bitmap compression history: Jpeg detection and quantizer estimation," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 230 – 235, feb 2003.

[15] Y.-L. Chen and C.-T. Hsu, "Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 396–406, June 2011.

[16] Z. Qu, W. Luo, and J. Huang, "A convolutive mixing model for shifted double JPEG compression with application to passive image authentication," in *Proc. of ICASSP 2008*, 2008, pp. 1661–1664.

[17] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii –xxxiv, Feb 1992.

[18] J. Lukáš and J. Fridrich, "Estimation of primary quantization matrix

in double compressed JPEG images," in *Digital Forensic Research Workshop*, 2003.
[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B 39*, pp. 1–38, 1977.
[20] T.Bianchi and A.Piva, "Detection of non-aligned double JPEG compression with estimation of primary compression parameters," in *Proc. of ICIP 2011*, Sept. 2011, pp. 1969–1972.
[21] M. Stamm and K. Liu, "Anti-forensics of digital image compression," *Information Forensics and Security, IEEE Transactions on*, 2011.
[22] F. Muller, "Distribution shape of two-dimensional DCT coefficients of natural images," *Electronics Letters*, vol. 29, no. 22, pp. 1935 –1936, Oct. 1993.
[23] E. Jaynes, *Probability Theory: The Logic of Science*. Cambridge, UK: Cambridge University Press, 2003.

Tiziano Bianchi (S'03-M'05) was born in Prato, Italy, in 1976. He received the M.Sc. degree (Laurea) in electronic engineering and the Ph.D. degree in information and telecommunication engineering from the University of Florence, Italy, in 2001 and 2005, respectively.

Since March 2005, he is with the Department of Electronics and Telecommunications, University of Florence as a Research Assistant. His research interests have involved processing of SAR images, signal processing in communications, multicarrier modulation techniques, and ultra-wideband systems. Current research topics include multimedia security technologies and signal processing in the encrypted domain.

Alessandro Piva (M'04-SM'10) received the Ph.D. degree in computer science and telecommunications engineering from the University of Florence in 1999.

From 2002 to 2004, he was a Research Scientist at the National Inter-University Consortium for Telecommunications. He is at present Assistant Professor at the University of Florence, Firenze. Italy. His current research interests are the technologies for multimedia content security, and image processing techniques for the Cultural Heritage field. He is coauthor of more than 100 papers published in international journals and conference proceedings.

Dr. Piva holds three Italian patents and an international one regarding watermarking. He serves as Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, of the *EURASIP Journal on Information Security*, and of the *LNCS Transactions on Data Hiding and Multimedia Security*.