Detection of shifted double JPEG compression by an adaptive DCT coefficient model

Shi-Lin Wang[1*], Alan Wee-Chung Liew[2], Sheng-Hong Li[3], Yu-Jin Zhang[3] and Jian-Hua Li[1]

Abstract

In many JPEG image splicing forgeries, the tampered image patch has been JPEG-compressed twice with different block alignments. Such phenomenon in JPEG image forgeries is called the shifted double JPEG (SDJPEG) compression effect. Detection of SDJPEG-compressed patches could help in detecting and locating the tampered region. However, the current SDJPEG detection methods do not provide satisfactory results especially when the tampered region is small. In this paper, we propose a new SDJPEG detection method based on an adaptive discrete cosine transform (DCT) coefficient model. DCT coefficient distributions for SDJPEG and non-SDJPEG patches have been analyzed and a discriminative feature has been proposed to perform the two-class classification. An adaptive approach is employed to select the most discriminative DCT modes for SDJPEG detection. The experimental results show that the proposed approach can achieve much better results compared with some existing approaches in SDJPEG patch detection especially when the patch size is small.

Keywords: Digital image forensics; Shifted double JPEG compression; JPEG coefficient analysis; Image splicing detection

1 Introduction

With the rapid development of image processing tools, manipulating a digital image without leaving obvious visual traces is becoming easier and easier. The detection of malicious tampering and the verification of the credibility of the original digital image have become important research topics.

Some researchers have proposed digital watermarking as an image/video content authentication technique [1-3]. However, these kinds of 'active authentication' methods have not been widely used because most images on the Internet are not required to be watermarked. Moreover, there is also the challenge of how to guide against hostile attacks in watermark-based approaches. Recently, a variety of 'passive authentication' methods [4-7] have been proposed which perform image content authentication by detecting certain cues produced during creation and modification of the image, such as double compression, light abnormality, re-sampling, and photo response non-uniformity noise (PRNU). Compared with the active

approach, the passive or blind authentication approach does not require additional watermarks or signatures and could have broader applications in image forensics.

JPEG is the most widely used image format. Authentication or detection of forgeries on JPEG images plays an important role in image forensics. Since most tampered JPEG image undergoes at least two JPEG compressions, many JPEG image authentication methods are based on detection of double JPEG compression. As the block discrete cosine transform (BDCT) is the key operation in JPEG compression, the distribution of the BDCT coefficients usually contains important information which could indicate the compression history. Hence, most passive image tampering detection approaches for JPEG images are based on BDCT coefficient analysis. Lukas and Fridrich [8] tried to identify double JPEG compression by detecting the double-peak effect in DCT coefficient histogram. Fu et al. [9] observed that the distribution of the first digit of DCT coefficients after JPEG compression followed the generalized Benford's law and stated that double JPEG compression could be detected because it would cause violations of the first digit law. In [10], Pevny and Fridrich introduced a machine learning approach for detection of double JPEG compression. A set of features from the

* Correspondence: wsl@sjtu.edu.cn
[1]School of Information Security Engineering, Shanghai Jiaotong University, No. 800, Dong Chuan Rd, Shanghai 200240, China
Full list of author information is available at the end of the article

histograms of several low-frequency DCT coefficients was extracted, and support vector machine (SVM) was adopted as the classifier. Recently, Farid [11] observed that re-compression would introduce additional local minima in the difference between the image and its JPEG-compressed counterpart, and these local minima, referred to as the JPEG ghosts, could be used to detect double compression. However, these methods are only effective when the block structure of the first and second JPEG compressions are aligned with each other.

In many JPEG image tampering situations, a foreign JPEG-compressed patch is inserted into an authentic image and the resultant image is re-compressed to form the new image. The tampered region has double JPEG compressions, but the block structures of the two compressions in the tampered region are usually not aligned with each other. Such case is referred to as shifted double JPEG compression (SDJPEG) [12] or non-aligned double JPEG compression (NA-JPEG) [13,14]. For SDJPEG, the double JPEG compression detection methods discussed above cannot achieve satisfactory results. Luo et al. [15] tried to detect SDJPEG by analyzing the blocking artifact characteristics matrix (BACM). They observed that BACM is symmetric for single JPEG compression but the symmetry is destroyed after SDJPEG. However, BACM is highly related to the image content and the detection performance would decrease if the testing images are very different from those in the training set. In order to solve the above problem, Chen and Hsu [16] extended the idea of BACM and proposed a feature which is less related to the image content by introducing the inter-block correl-ation. However, since the statistical features in both [15] and [16] require large amount of data to obtain high dis-criminative power, their methods do not work well for small SDJPEG patches.

Recently, Bianchi and Piva [13] tried to detect SDJPEG effects by examining the integer periodicity in the DCT coefficient histogram when the BDCT is computed ac-cording to the first JPEG compression. In addition, they also proposed a statistical model that characterizes the artifacts due to SDJPEG [14]. They have observed that the shifted JPEG compression will introduce a Gaussian noise to each DCT coefficients and approximated the vari-ance of the noise by the quantization step of the shifted compression. Inspired by [14], we propose a highly effect-ive SDJPEG image tampering detection method in this paper. The major contributions of our work lie in the fol-lowing: (i) we perform a rigorous theoretic analysis of the DCT coefficient variations caused by SDJPEG and derive from it a rigorous statistical model for the DCT coeffi-cients of SDJPEG patches, which provides a more accurate estimation of the quantization noise introduced by the shifted JPEG quantization compared with that in [13,14]; (ii) based on the analysis, we propose an effective discrim-inative feature to detect SDJPEG patches; and (iii) we propose an adaptive DCT component selection method to select the most discriminative DCT components. Our al-gorithm not only detects image forgeries but also locates the tampered regions accurately.

This paper is organized as follows. Section 2 gives a description of the 'crop-and-paste' image tampering problem and introduces current approaches. Section 3 gives an in-depth analysis on the DCT coefficient varia-tions caused by shifted JPEG compression and describes the DCT coefficient histogram model for SDJPEG patches. In Section 4, a new discriminative feature is in-troduced to detect SDJPEG patches. An adaptive DCT mode selection method and a tampering detection algorithm are also given. Section 5 presents the experi-mental results comparing our approach with several state-of-the-art techniques. Finally, Section 6 draws the conclusion.

2 Crop-and-paste image tampering detection

Figure 1 illustrates a typical crop-and-paste image tamper-ing scenario. Given an original image shown in Figure 1a, an image region of arbitrary shape highlighted in Figure 1b was cropped from a foreign image and was pasted onto the original image to construct the tampered image of Figure 1c (with tampered region highlighted). The tam-pering detection problem can be stated as follows: given an image, detect whether it has been tampered through crop-and-paste and if so, locate the region where the crop-and-paste occurred. Since any arbitrary shape region can be divided into a concatenation of small squared patches, we consider a squared image patch composed of a number of 8 × 8 blocks as the fundamen-tal unit in this work.
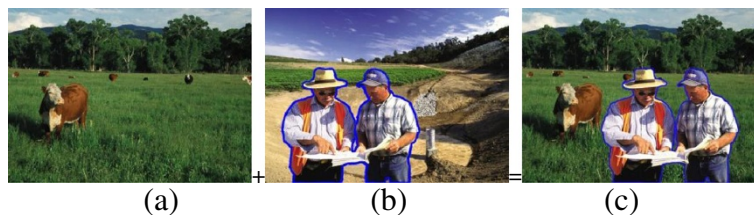

Figure 1 A typical crop-and-paste image tampering scenario. (a) Original image. **(b)** Foreign image being cropped. **(c)** Final spliced image.

To facilitate subsequent discussions, we first give some definitions. An image patch with square shape and containing a number of 8×8 blocks can be classified into one of five categories as shown in Figure 2 (from left to right, top to bottom):

(i) Uncompressed: These patches are raw images and have never been JPEG compressed.
(ii) Aligned single JPEG compressed (ASJPEG in short): When the uncompressed image patches undergo a JPEG compression with the block structure aligned to the image patch, the output image patches are called ASJPEG patches. They are usually referred to as single JPEG image patches in the literature.
(iii) Aligned double JPEG compressed (ADJPEG in short): When the ASJPEG patches undergo another JPEG compression and the block structures of the two JPEG compressions are aligned with each other, the output image patches are called ADJPEG patches. They are usually referred to as double JPEG image patches in the literature.
(iv) Shifted double JPEG compressed (SDJPEG in short): When the ASJPEG patches undergo another JPEG compression and the block structures of the two JPEG compressions are different, the output image patches are called SDJPEG patches.
(v) Shifted single JPEG compressed (SSJPEG in short): Different from SDJPEG patches, when the image

patches before the shifted JPEG compression (with the block structure in dashed line) have never been compressed with the block structure starting from the top-left corner, the output image patches are called SSJPEG patches.

There are two general approaches for crop-and-paste tampering detection in JPEG images: aligned double JPEG detection and shifted double JPEG detection. Both approaches involve analyzing an image patch within a window aligned with the JPEG block structure of the final image, i.e., denoted hereafter as aligned patch, to detect any sign of tampering. Their differences and requirements are briefly summarized in Table 1. In most crop-and-paste tampering detection applications, both kinds of approaches are applied together to provide a more robust detection result.

The underlying rationale of the aligned double JPEG detection approach is as follows. If the original image has been JPEG compressed before and undergoes another JPEG compression after crop-and-paste tampering, image patches in the unmodified region in the final JPEG image (such as aligned patch I in Figure 3a) would have been compressed twice with the same block structure and exhibit the aligned double JPEG effect, while image patches in the tampered region (such as aligned patch II in Figure 3a) will not show such effect. Algorithms in this approach look for untampered patch, i.e.,
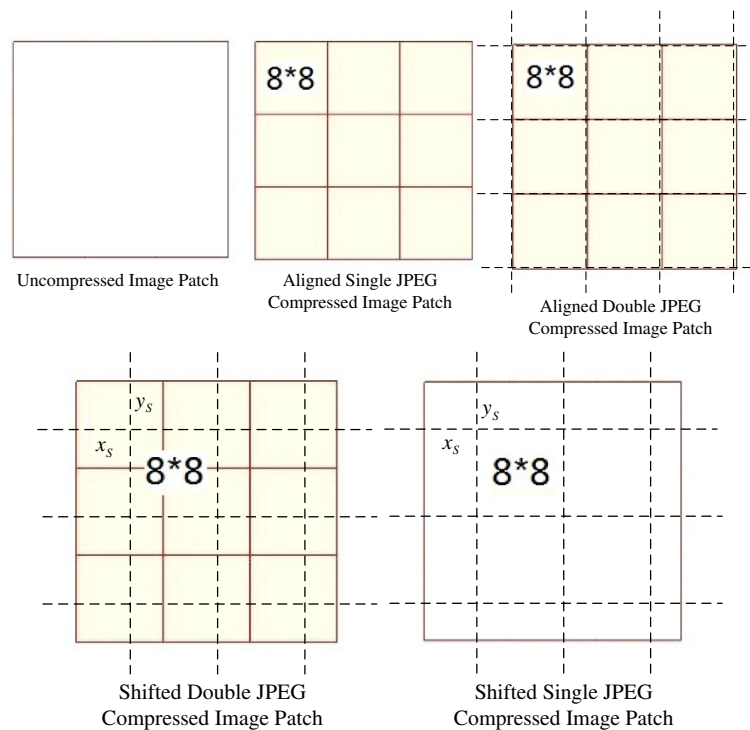

Figure 2 Five kinds of image patches.

**Table 1 Requirements of the crop-and-paste tampering detection methods**

	Aligned double JPEG detection approach	Shifted double JPEG detection approach
Original image	JPEG compressed	JPEG compressed or raw
Foreign image being inserted	JPEG compressed or raw	JPEG compressed
Final image	JPEG compressed	JPEG compressed
Decision basis	Presence of ADJPEG effect indicates un-tampered patch	Presence of SDJPEG effect indicates tampered patch
Additional limitation	The quantization matrices of original and final image cannot be the same	The block structure of the inserted patch cannot align to that of the final image

they detect the presence of aligned double JPEG effect, to establish the authenticity of the image patch. If such effect is absent, the patch is assumed to be tampered. It has been shown [8-11,17] that if the quality factors of the original and final image are the same, the detection performance would not be satisfactory. Recently, Huang et al. [18] proposed an aligned double JPEG detection algorithm that applied to the case where both the original and final images have the same quantization matrix. However, their approach cannot provide accurate detection results for small image patch. Finally, the aligned double JPEG detection approach cannot detect the tampered region if the original image is uncompressed.

In contrast, the shifted double JPEG detection-based approach [13-16] looks for tampered patch by detecting cues of tampering in an image patch. The basic idea is that if the inserted image region has been JPEG compressed

earlier, the second JPEG compression will leave some cues of shifted compression in the tampered region in the final image (such as the additional blocking artifacts [15,16]). For the aligned patches fully or partially located in the tampered region in the final image (such as aligned patch II in Figure 3a), such cues would exist, while for the aligned patches located in the unmodified region in the final image (such as aligned patch I in Figure 3a), such cues would be absent. Hence, the tampered region can be located by searching for all the aligned patches where the cues left by shifted compression exist. However, such cues vary greatly with image content and the quality factor of the inserted patch, and it is difficult to find a robust and discriminative feature for all situations. Moreover, this approach needs a large image patch to obtain robust detection results and is less effective for small patches.

In this paper, we propose a new shifted double JPEG detection method which also examines the characteristics of the inserted patch. Similar to [13,14], our algorithm considers image patches that are not aligned with the block structure of the final image (such as non-aligned patches I and II in Figure 3b). When analyzing the non-aligned patches in the final image, SDJPEG patches (such as non-aligned patch II in Figure 3b where the JPEG-compressed inserted patch undergoes a shifted JPEG compression, i.e., during JPEG compression of the final image) are located in the tampered region, while SSJPEG patches (such as non-aligned patch I in Figure 3b) are located in the untampered region. Note that the JPEG compression of the final image is aligned for the aligned patches in Figure 3a and shifted for non-aligned patches Figure 3b. In our algorithm, an exhaustive search on 63 possible

(a)   (b)

Figure 3 Aligned (a) and shifted (b) image patches in the tampered image.
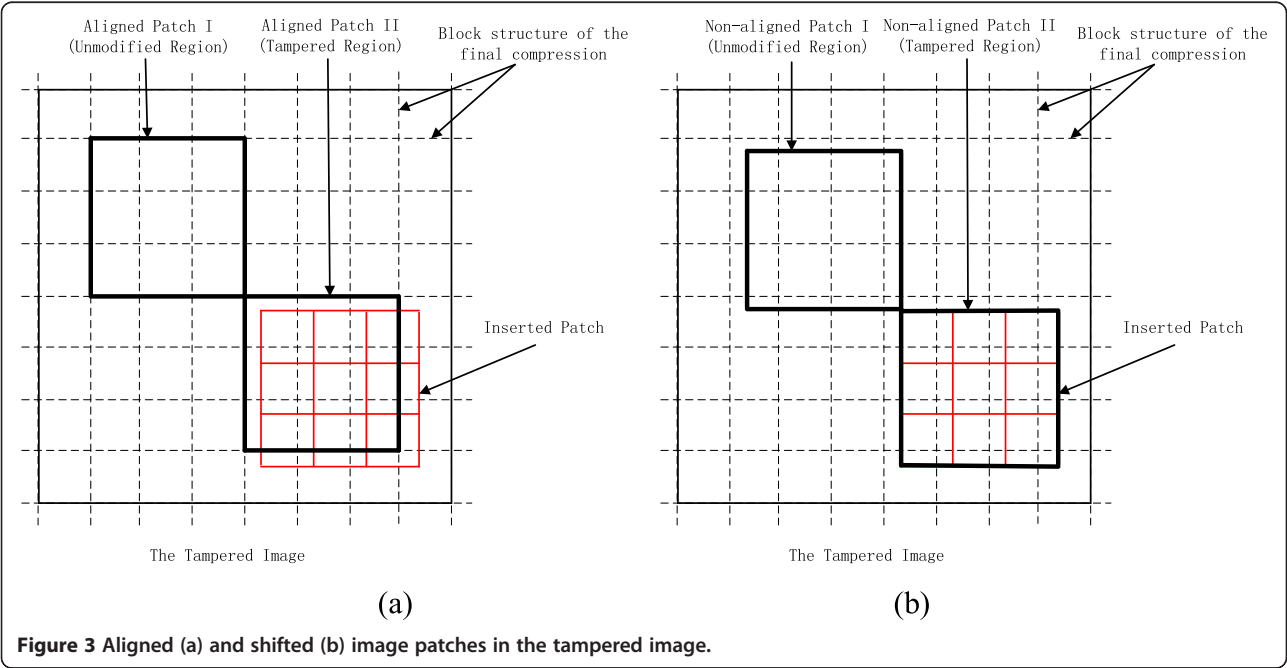
locations of the non-aligned image patches is performed to detect tampering. We will show in the experiment section that the increase in computational cost is worthwhile and our algorithm can achieve much better detection performance compared with the existing methods.

3 DCT coefficient analysis for SDJPEG patches

Since SDJPEG patches are generated by shifted JPEG compression on ASJPEG patches, we examine how the shifted JPEG compression affects the DCT coefficients. We first describe the effect of shifted JPEG compression on the DCT coefficients in Section 3.1. Then we derive a specific DCT coefficient model for SDJPEG patches in Section 3.2. The notation used hereafter is summarized in Table 2.

3.1 DCT coefficient variations caused by shifted JPEG compression

Before analyzing the effects on DCT coefficients caused by shifted JPEG compression, a simpler case, i.e., the effects cause by aligned JPEG compression, is discussed. Given an 8×8 image block A, its DCT coefficients can be represented by $D^{O}(A) = \left[D^{O}_{m,n}(A)\right]$ $1 \le m, n \le 8$. If block A undergoes an aligned JPEG compression (as shown in Figure 4a) with the quality factor QF, the resulting DCT coefficients of block A, denoted by $D^{AJ}(A) = \left[D^{AJ}_{m,n}(A)\right]$ $1 \le m, n \le 8$, will be a multiple of the quantization step. Hence, aligned JPEG compression will induce a zero-mean quantization error (denoted by $E^{AJ}_{m,n}(A)$) for all the DCT coefficients, i.e.,

$$D^{AJ}_{m,n}(A) = D^{O}_{m,n}(A) + E^{AJ}_{m,n}(A) \tag{1}$$

The discussions above can be extended to the case of shifted JPEG compression. If block A undergoes a shifted JPEG compression with a coordinate shift of (x_S, y_S) and quality factor QF (as shown in Figure 4b), the DCT coefficients of block A become $D^{SJ}(A) = \left[D^{SJ}_{m,n}(A)\right]$ $1 \le m, n \le 8$. To derive the expression for $D^{SJ}(A)$, we consider the DCT coefficients of the neighboring 8×8 aligned blocks whose block structure coincides with the block structure of the shifted JPEG compression on A. As shown in Figure 4c,d, block A is surrounded by four aligned blocks B_1, B_2, B_3, and B_4. According to [19], the DCT coefficients of block A and B_i $(i = 1,2,3,4)$ are related by (details are given in Appendix)

$$\begin{aligned} A &= A_1 + A_2 + A_3 + A_4 \Rightarrow D(A) \\ &= D(A_1) + D(A_2) + D(A_3) + D(A_4) \end{aligned} \tag{2}$$

$$\begin{aligned} A_i &= H_{i1} B_i H_{i2} \Rightarrow D(A) \\ &= \sum_{i=1}^{4} D(H_{i1}) D(B_i) D(H_{i2}) \end{aligned} \tag{3}$$

where H_{i1} and H_{i2} $(i = 1,2,3,4)$ are the row and column translation matrices to translate a specific block from B_i to A_i, as shown in Figure 4. Note that $D(H_{i1})$ and $D(H_{i2})$ $(i = 1,2,3,4)$ are only related to the coordinate shifts (x_S, y_S).

Let $D^{O}(B_i) = \left[D^{O}_{m,n}(B_i)\right]$ $1 \le m, n \le 8$ be the original DCT coefficients of block B_i $i = 1,2,3,4$ and thus before compression $D^{O}(A) = \sum_{i=1}^{4} D(H_{i1}) D^{O}(B_i) D(H_{i2})$ according to (3). $D^{O}(B_i)$ becomes $D^{AJ}(B_i)$, $i = 1,2,3,4$ after a

Table 2 Notation

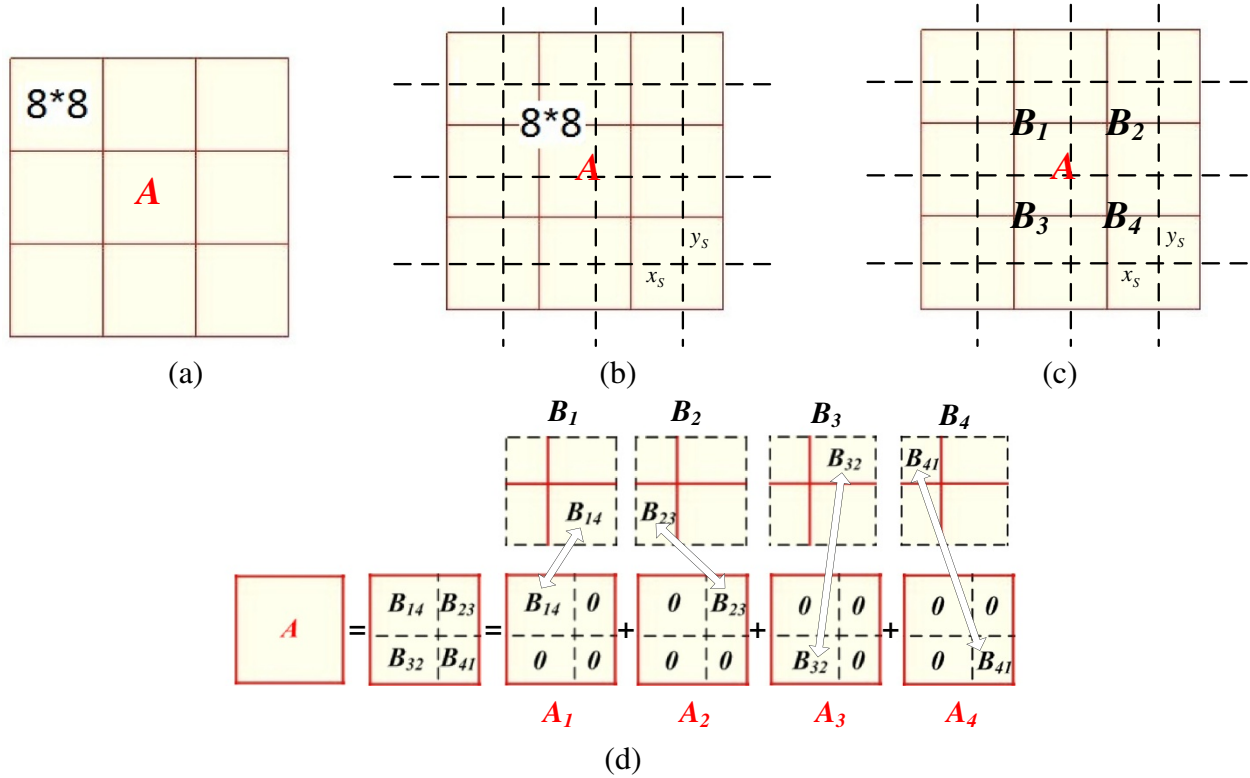|	Description
$D, D^{O}, D^{AJ}, D^{SJ}, D^{SD}$	DCT coefficients of an 8×8 image block, where the superscripts O, AJ, SJ, and SD stand for the original block before JPEG compression, the block after aligned JPEG compression, the block after shifted JPEG compression, and the block after shifted double JPEG compression respectively. Note that no superscript stands for the general case
$D^{O}_{m,n}, D^{AJ}_{m,n}, D^{SJ}_{m,n}, D^{SD}_{m,n}$	The (m,n)th DCT coefficient of an 8×8 image block, the subscript (m,n) stands for the (m,n)th DCT mode
$E^{AJ}_{m,n}, E^{SJ}_{m,n}, E^{SJR}_{m,n}$	Quantization error of the (m,n)th DCT coefficient caused by aligned JPEG compression, shifted JPEG compression, and shifted JPEG compression with spatial rounding, respectively
$\sigma^{SJ}_{m,n}, \sigma^{SJR}_{m,n}$	Standard deviation of the quantization error of the (m,n)th DCT coefficient caused by shifted JPEG compression and shifted JPEG compression with spatial rounding, respectively
$h_{m,n}(f), h^{AJ}_{m,n}(f), h^{SS}_{m,n}(f), h^{SD}_{m,n}(f)$	Histograms of the (m,n)th DCT coefficient of the image patch, where the superscripts AJ, SS, and SD stand for the image patch after aligned JPEG compression, the SSJPEG patch and the SDJPEG patch, respectively. Note that no superscript stands for the general case
(x_S, y_S)	Coordinate shift of the shifted JPEG compression
QF_1, QF_2	Quality factor of the primary JPEG compression and the final JPEG compression, respectively
$q_{m,n}(QF)$	Quantization step of the (m,n)th DCT mode with the quality factor QF.
$G(\mu, \sigma)$	Gaussian distribution with mean μ and standard deviation σ

Figure 4 DCT coefficient variations caused by shifted JPEG compression. Block A with **(a)** aligned JPEG compression **(b)** shifted JPEG compression. **(c)** Block A and its four adjacent blocks B_1 to B_4, whose block structures coincide with the block structure of shifted compression shown in dotted lines. **(d)** Relationship between block A and its four adjacent blocks B_1 to B_4.

JPEG compression is performed on these aligned blocks. Based on the analysis about aligned JPEG compression in (1), $D^{AJ}(B_i) = D^O(B_i) + E^{AJ}(B_i)$, $i = 1,2,3,4$. The aligned JPEG compression on B_i is equivalent to a shifted JPEG compression on A. Hence, after compression, the DCT coefficients of block A would change to

$$
\begin{aligned}
D^{SJ}(A) &= \sum_{i=1}^{4} D(H_{i1}) D^{AJ}(B_i) D(H_{i2}) \\
&= \sum_{i=1}^{4} D(H_{i1}) \big(D^O(B_i) + E^{AJ}(B_i) \big) D(H_{i2}) \\
&= D^O(A) + \sum_{i=1}^{4} D(H_{i1}) E^{AJ}(B_i) D(H_{i2}) \\
&= D^O(A) + E^{SJ}(A)
\end{aligned}
$$

$$(4)$$

where $E^{SJ}(A) = \Big[E^{SJ}_{m,n}(A) \Big] = \sum_{i=1}^{4} D(H_{i1}) E^{AJ}(B_i) D(H_{i2})$ denotes the shifted quantization error caused by shifted JPEG compression. For any DCT mode (m,n), the shifted quantization error $E^{SJ}_{m,n}(A)$ can be expressed by a linear combination of $4 \times 64 = 256$ zero-mean random variables,

i.e., $E^{SJ}_{m,n}(A) = \sum_{i=1}^{4} \sum_{u=1}^{8} \sum_{v=1}^{8} c^i_{m,n}(u,v) E^{AJ}_{u,v}(B_i)$ and $c^i_{m,n}(u,v)$ is the weighting parameter determined by $D(H_{i1})$ and $D(H_{i2})$ ($i = 1,2,3,4$). According to the Central Limit Theorem (CLT), $E^{SJ}_{m,n}(A)$ follows a zero-mean Gaussian distribution denoted by $G\big(0, \sigma^{SJ}_{m,n}\big)$ with standard deviation $\sigma^{SJ}_{m,n}$. $\sigma^{SJ}_{m,n}$ can be calculated with knowledge of the standard deviations of these 256 random variables ($E^{AJ}_{u,v}(B_i)$ $1 \le i \le 4, 1 \le u, v \le 8$) and the weighting coefficients $c^i_{m,n}(u,v)$ $1 \le i \le 4$, $1 \le u, v \le 8$.

In order to estimate the quantization error caused by aligned JPEG compression, we divide all the DCT coefficients into two parts, i.e., the DC and AC components. For the DC component, since the quantization step is relatively small and the DC coefficients have a large dynamic range, the quantization error caused by aligned JPEG compression approximately follows the uniform distribution of $\left[-\frac{q_{1,1}(QF)}{2}, \frac{q_{1,1}(QF)}{2} \right)$, where $q_{1,1}(QF)$ is the quantization step for the DC coefficients with the quality factor QF. For the AC components, according to [20], the AC components for a natural image approximately follow a Laplacian distribution. If we fit the coefficients

of AC components in the image patch to a Laplacian model, the standard deviations of $E^{AJ}_{u,v}(B_i)$ can be directly calculated with the quality factor QF. In addition, from (3), the weighting coefficients, $c^i_{m,n}(u,v)$, are determined by the coordinate shift (x_S, y_S). Then the theoretical value of $\sigma^{SJ}_{m,n}$ can be calculated when QF and (x_S, y_S) are known.

To summarize, the shifted JPEG compression will induce a zero-mean Gaussian-distributed quantization error for all the DCT coefficients, and for different DCT coefficients, the standard deviations of the shifted quantization error are different and can be calculated when the quality factor and the coordinate shift of the compression are known.

3.2 DCT coefficient analysis on SDJPEG patches

As discuss earlier, SDJPEG patches are constructed by performing shifted JPEG compression on ASJPEG patches. Similar to the ASJPEG patches, the DCT coefficients of SDJPEG patches also have certain specific distributions.

Given a gray scale image patch *IMG* consisting of a number of 8×8 blocks aligned JPEG compressed with the quality factor QF_1, let $D^{AJ}_{m,n}(k)$ denote the (m,n)-*th* DCT coefficient of the kth block in IMG. Since it has been JPEG compressed with quality factor QF_1, $D^{AJ}_{m,n}(k)$ would be a multiple of the quantization step of the (m,n)th DCT mode (denoted by $q_{m,n}(QF_1)$). Hence, considering all the (m,n)th DCT coefficients in IMG, the normalized histogram $h^{AJ}_{m,n}(f)$ of the (m,n)th DCT coefficients is given by

$$h^{AJ}_{m,n}(f) = \sum_{i=-N}^{N} w_i \delta\left(f - i \times q_{m,n}(QF_1)\right) \quad (5)$$

where w_i is the normalized frequency of the (m,n)th DCT coefficients having a value of $i \times q_{m,n}(QF_1)$ and $Nq_{m,n}(QF_1)$ is the maximum absolute value of the (m,n)th DCT coefficient in IMG.

When the ASJPEG-compressed image patch IMG is transformed back to the spatial domain, two kinds of errors would be introduced. One is the truncation error. Since the luminance value of the gray scale image ranges from 0 to 255, any gray level greater than 255 or less than 0 will be truncated to 255 or 0, respectively. However, as discussed in [21], such kind of error seldom appears in natural images (about 1% of the image blocks have truncation errors), and any block with pixels having luminance value of 0 or 255 is discarded for further analysis. The other is the rounding error, i.e., a rounding process will be carried out after IDCT to ensure that the luminance value in the spatial domain is an integer. Such rounding error in the spatial domain will lead to bias (denoted by $E^{rounding}_{m,n}$) in all the DCT coefficients.

Assuming that the rounding error in the spatial domain follows a zero-mean uniform distribution with range [−0.5, 0.5) and considering that the DCT transform is unitary, $E^{rounding}_{m,n}$ is Gaussian distributed with zero mean and variance 1/12 for all $1 \le m$, $n \le 8$ according to the CLT [14].

After DCT-to-spatial transformation, the image patch undergoes a shifted JPEG compression with quality factor QF_2 and the coordinate shifts (x_S, y_S). According to the analysis in Section 3.1, the shifted JPEG compression will introduce a zero-mean Gaussian-distributed error term $E^{SJ}_{m,n}(k)$. The (m,n)th DCT coefficient of the kth block in the final SDJPEG patch, denoted by $D^{SD}_{m,n}(k)$, is given by

$$\begin{aligned} D^{SD}_{m,n}(k) &= D^{AJ}_{m,n}(k) + E^{rounding}_{m,n} + E^{SJ}_{m,n}(k) \quad (6) \\ &= D^{AJ}_{m,n}(k) + E^{SJR}_{m,n}(k) \end{aligned}$$

where $E^{SJR}_{m,n}(k)$ is the overall error of the (m,n)th DCT coefficient during the construction of the SDJPEG patch from the ASJPEG patch. Since $E^{rounding}_{m,n}$ and $E^{SJ}_{m,n}(k)$ are both Gaussian distributed and independent of each other, $E^{SJR}_{m,n}(k)$ also follows the Gaussian distribution with zero-mean and standard deviation of $\sigma^{SJR}_{m,n} = \sqrt{\left(\sigma^{SJ}_{m,n}\right)^2 + 1/12}$. The histogram of the (m,n)th DCT coefficients after shifted double JPEG compression, denoted by $h^{SD}_{m,n}(f)$, is then given by

$$\begin{aligned} h^{SD}_{m,n}(f) &= h^{AJ}_{m,n}(f) \otimes G\left(0, \sigma^{SJR}_{m,n}\right) \\ &= \sum_{i=-N}^{N} w_i G\left(i \times q_{m,n}(QF_1), \sigma^{SJR}_{m,n}\right) \quad 1 \le m, n \le 8 \end{aligned}$$

$$(7)$$

Figure 5 illustrates the DCT coefficient histograms for the SDJPEG and SSJPEG patches from the uncompressed 512×512 image 'Lena.bmp' (Figure 5 (c1)). From the figure, it can be observed that (i) the coefficient variations caused by rounding errors and shifted JPEG compression follow Gaussian distributions and thus $h^{SD}_{m,n}(f)$ can be approximated by (7); (ii) the AC DCT coefficient distribution of the SSJPEG patch (shown in Figure 5 (c2 and c3)) follows the Laplacian distribution, which is similar to that of uncompressed images [22], and is very different from that of the SDJPEG patches shown in Figure 5 (a2 and b2). Based on the above analysis, it can be concluded that the SDJPEG patches can be differentiated from SSJPEG patches by analyzing the DCT coefficient distributions. In the following section, details of the SDJPEG detection method based on the proposed DCT coefficient model in (7) will be elaborated.
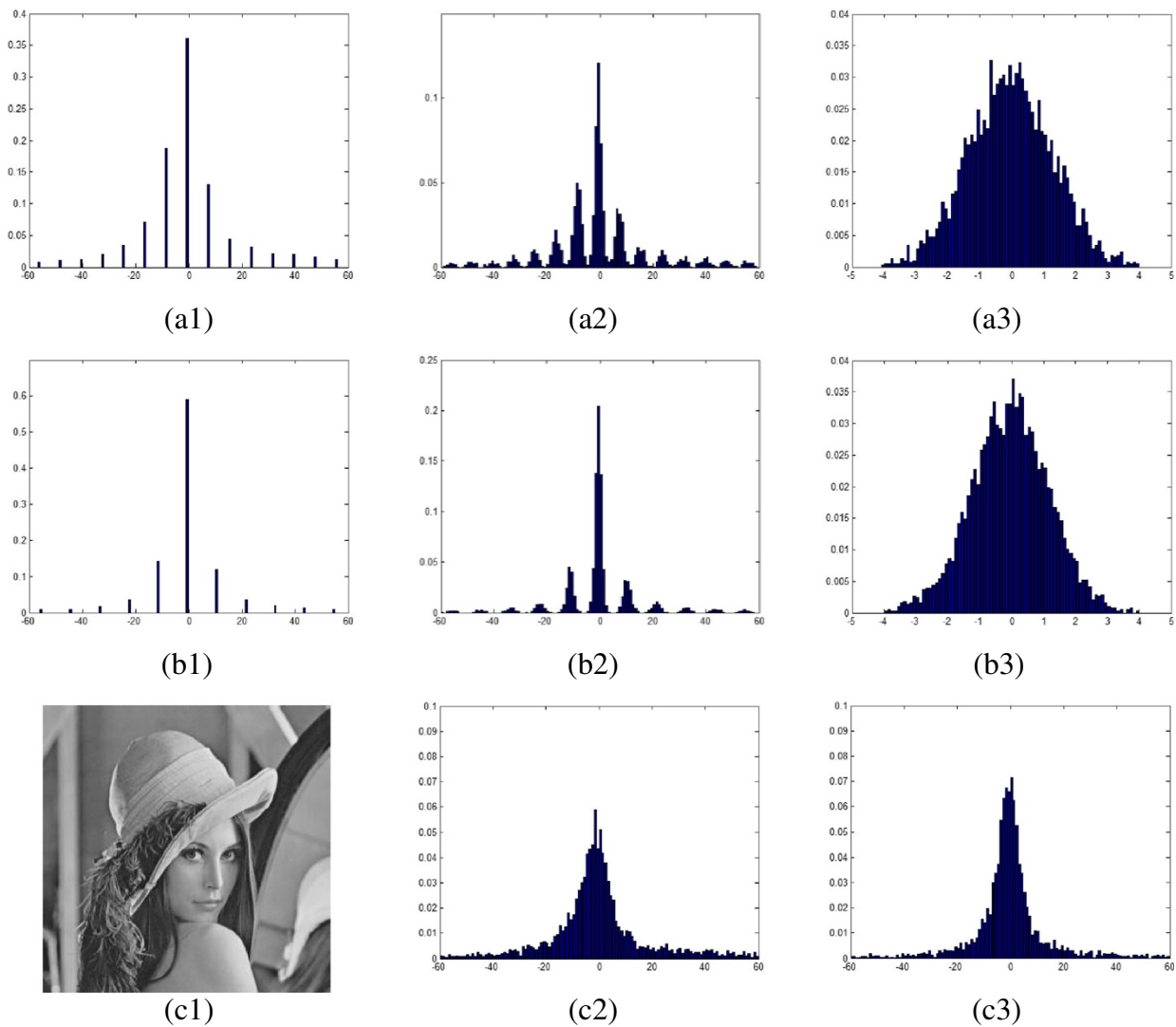
Figure 5 DCT coefficient histograms for SDJPEG and SSJPEG patches from uncompressed 512 × 512 image. (a1 and b1) $h_{1,3}^{AJ}(f)$ and $h_{3,1}^{AJ}$ (f) with $QF_1 = 60$. **(a2 and b2)** $h_{1,3}^{SD}(f)$ and $h_{3,1}^{SD}(f)$ with $QF_2 = 90$ and $(x_S, y_S) = (4,4)$. **(a3 and b3)** Normalized distribution of the shifted double quantization error $E_{1,3}^{SJR}$ and $E_{3,1}^{SJR}$. **(c1)** The original image patch. **(c2 and c3)** The (1,3)th and (3,1)th DCT coefficient histograms of the SSJPEG-compressed image with QF = 90 and $(x_S, y_S) = (4,4)$, respectively.

4 Detection of image patch with SDJPEG compression

The analysis in Section 3 shows that the DCT coefficient distribution of SDJPEG patches follows a weighted summation of Gaussian components with the same standard deviation (as shown in (7) and Figure 5). However, in order to detect SDJPEG patches based on (7), two questions have to be addressed: (i) How to obtain discriminative features that capture the differences in the (m,n)th DCT coefficient distributions between SDJPEG and SSJPEG patches? (ii) How to select DCT modes which provide high discriminative power since the differences in the DCT

coefficient distributions of SDJPEG and SSJPEG patches would vary for different DCT mode? In the following, we will address these questions.

4.1 Discriminative feature extraction based on the (m,n)th DCT coefficients

For a specific DCT mode, say (m,n)th, the following detection algorithm is carried out by determining whether the histogram $h_{m,n}(f)$ is similar to that of $h_{m,n}^{SD}(f)$.

Given a specific quantization step $q_{m,n}$, we project the histogram $h_{m,n}(f)$ onto the interval $\left(-\frac{q_{m,n}}{2}, \frac{q_{m,n}}{2}\right)$, and the

sum of the histogram function within the interval is defined as

$$sh_{m,n}(f) = \sum_{i=-N}^{N} h_{m,n}\left(f + i \times q_{m,n}\right), \quad -\frac{q_{m,n}}{2} \le f < \frac{q_{m,n}}{2} \tag{8}$$

where $Nq_{m,n}$ is the maximum absolute value of the (m,n)th DCT coefficient.

According to (7), for SDJPEG-compressed image patches, $h_{m,n}^{SD}(f)$ follows a weighted summation of Gaussian components with a standard deviation of $\sigma_{m,n}^{SJR}$. Hence, $sh_{m,n}^{SD}(f)$ would follow a specific distribution determined by $\sigma_{m,n}^{SJR}$ and $q_{m,n}$. Based on the different ratios of $q_{m,n}/\sigma_{m,n}^{SJR}$, $sh_{m,n}^{SD}(f)$ can be obtained as follows. It should be noted that in our discussions, the Gaussian distribution, $G(0, \sigma)$, is assumed to be bounded in $[-3\sigma, 3\sigma]$, and any outliers are omitted.

If $\frac{q_{m,n}}{2} \ge 3\sigma_{m,n}^{SJR}$, i.e., all the Gaussian components in $h_{m,n}^{SD}(f)$ are isolated, $sh_{m,n}^{SD}(f)$ would follow a Gaussian distribution of $G\left(0, \sigma_{m,n}^{SJR}\right)$ (as shown in Figure 6a), i.e.,

$$sh_{m,n}^{SD}(f) = \sum_{i=-N}^{N} w_i G\left(0, \sigma_{m,n}^{SJR}\right) = G\left(0, \sigma_{m,n}^{SJR}\right) \tag{9}$$
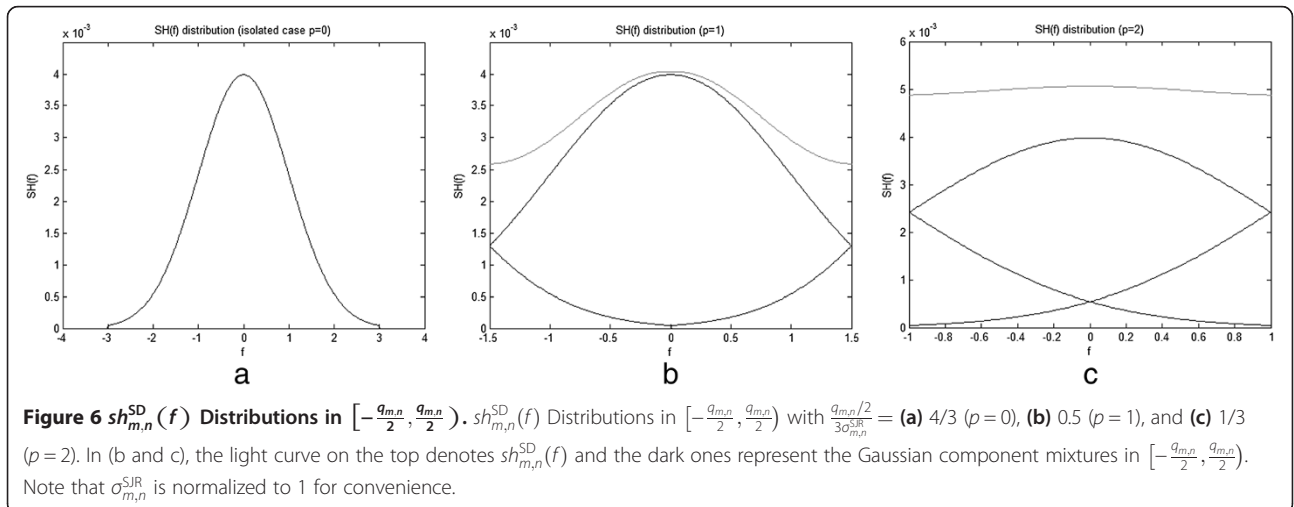
$$-\frac{q_{m,n}}{2} \le f < \frac{q_{m,n}}{2}$$

If $p \times \frac{q_{m,n}}{2} < 3\sigma_{m,n}^{SJR} \le (p+1) \times \frac{q_{m,n}}{2} \quad p = 1, 2, \ldots$, some Gaussian components would overlap with each other and $sh_{m,n}^{SD}(f)$ would follow a distribution of a mixture of $(p+1)$ Gaussian distributions (as shown in Figure 6b,c), i.e.,
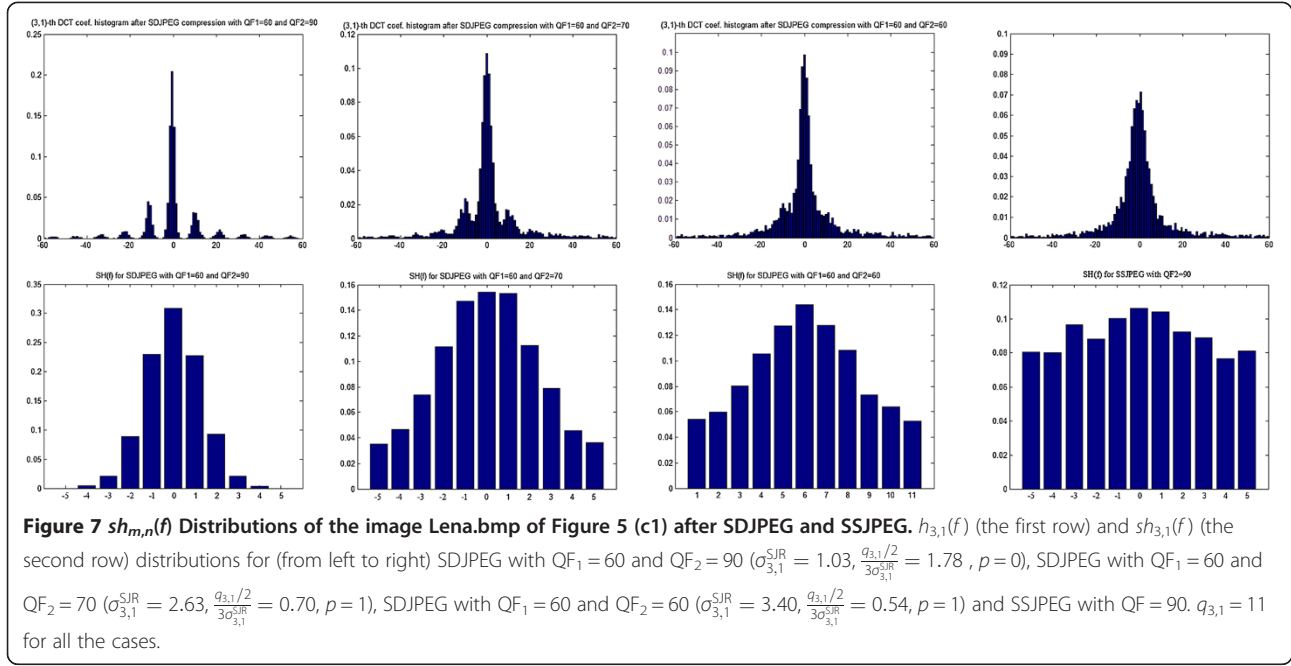
$$sh_{m,n}^{SD}(f) = \begin{cases} \sum_{i=-N}^{N} w_i \left[\sum_{j=0}^{p} G\left(f + j \times q_{m,n}, \sigma_{m,n}^{SJR}\right) \right] \\ = \sum_{j=0}^{p} G\left(f + j \times q_{m,n}, \sigma_{m,n}^{SJR}\right) \quad -q_{m,n}/2 \le f \le 0 \\ \sum_{i=-N}^{N} w_i \left[\sum_{j=0}^{p} G\left(f - j \times q_{m,n}, \sigma_{m,n}^{SJR}\right) \right] \\ = \sum_{j=0}^{p} G\left(f - j \times q_{m,n}, \sigma_{m,n}^{SJR}\right) \quad 0 \le f \le q_{m,n}/2 \end{cases} \tag{10}$$

where $p + 1 = \text{ceiling}\left(\frac{3\sigma_{m,n}^{SJR}}{q_{m,n}/2}\right)$.

From (9) and (10) and Figure 6, it is observed that for the isolated or slight overlapping cases ($p = 0$ or $p = 1$), $sh_{m,n}^{SD}(f)$ has a distinctive peak at small $|f|$. The peak at small $|f|$ becomes more prominent with larger $q_{m,n}/\sigma_{m,n}^{SJR}$. Such phenomenon does not occur for SSJPEG patches. Figure 7 illustrates $sh_{m,n}(f)$ distributions of the image Lena.bmp of Figure 5 (c1) after SDJPEG and SSJPEG with various parameter settings.

From Figure 7, it is observed that the energy of $sh_{m,n}(f)$ for SDJPEG image patches is concentrated on the center region whereas the energy of $sh_{m,n}(f)$ for SSJPEG patches is almost evenly distributed. It should be noted that when the size of the image patch is small, e.g., 64×64, the total number of DCT coefficients is small (64 coefficients in total), and it is difficult to estimate the actual distribution of the patch accurately and robustly using such limited data. In order to solve the above problem of inadequate data, the 1-D feature similar to that in [23] is adopted to differentiate between SDJPEG and SSJPEG image patches as follows:

Figure 6 $sh_{m,n}^{SD}(f)$ **Distributions in** $\left[-\frac{q_{m,n}}{2}, \frac{q_{m,n}}{2}\right)$. $sh_{m,n}^{SD}(f)$ Distributions in $\left[-\frac{q_{m,n}}{2}, \frac{q_{m,n}}{2}\right)$ with $\frac{q_{m,n}/2}{3\sigma_{m,n}^{SJR}} =$ **(a)** 4/3 ($p = 0$), **(b)** 0.5 ($p = 1$), and **(c)** 1/3 ($p = 2$). In (b and c), the light curve on the top denotes $sh_{m,n}^{SD}(f)$ and the dark ones represent the Gaussian component mixtures in $\left[-\frac{q_{m,n}}{2}, \frac{q_{m,n}}{2}\right)$. Note that $\sigma_{m,n}^{SJR}$ is normalized to 1 for convenience.

Figure 7 $sh_{m,n}(f)$ Distributions of the image Lena.bmp of Figure 5 (c1) after SDJPEG and SSJPEG. $h_{3,1}(f)$ (the first row) and $sh_{3,1}(f)$ (the second row) distributions for (from left to right) SDJPEG with $QF_1 = 60$ and $QF_2 = 90$ ($\sigma_{3,1}^{SJR} = 1.03$, $\frac{q_{3,1}/2}{3\sigma_{3,1}^{SJR}} = 1.78$, $p = 0$), SDJPEG with $QF_1 = 60$ and $QF_2 = 70$ ($\sigma_{3,1}^{SJR} = 2.63$, $\frac{q_{3,1}/2}{3\sigma_{3,1}^{SJR}} = 0.70$, $p = 1$), SDJPEG with $QF_1 = 60$ and $QF_2 = 60$ ($\sigma_{3,1}^{SJR} = 3.40$, $\frac{q_{3,1}/2}{3\sigma_{3,1}^{SJR}} = 0.54$, $p = 1$) and SSJPEG with $QF = 90$. $q_{3,1} = 11$ for all the cases.

$$s_{m,n} = \int_{R_2} sh_{m,n}(f)df \bigg/ \int_{R_1} sh_{m,n}(f)df \tag{11}$$

where $R_1 = \left(-\frac{q_{m,n}}{6}, \frac{q_{m,n}}{6}\right)$ representing the central region and $R_2 = \left(-\frac{q_{m,n}}{2}, -\frac{q_{m,n}}{3}\right) \cup \left(\frac{q_{m,n}}{3}, \frac{q_{m,n}}{2}\right)$ representing the peripheral region.

For SDJPEG patches, the reference value of the feature, denoted by $s_{m,n}^{SD}$, can be derived from $sh_{m,n}^{SD}(f)$ and is determined by $\sigma_{m,n}^{SJR}$ and $q_{m,n}$. The reference value of SSJPEG patch, denoted by $s_{m,n}^{SS}$, is obtained as follows. Since the SSJPEG patch has never been compressed with the block structure starting from the top-left corner of the image patch, the DCT coefficients of SSJPEG can be assumed distributed approximately as the original uncompressed image patch [8,14]. Moreover, $h_{m,n}^{SS}(f)$ is equivalent to the distribution of the quantization error of the (m,n)th DCT component with the quantization step $q_{m,n}$. Hence, $h_{m,n}^{SS}(f)$ can be approximated using the aligned JPEG quantization error estimation approach introduced in Section 3.1. Then $s_{m,n}^{SS}$ can be derived from (8) and (11). Note that for the low-frequency DCT modes, the quantization step $q_{m,n}$ is relatively smaller compared with the dynamic range of the DCT coefficients and thus $sh_{m,n}^{SS}(f)$ is almost evenly distributed and $s_{m,n}^{SS} \approx 1 > s_{m,n}^{SD}$.

The final discriminative feature indicating the likelihood of an image patch having been SDJPEG compressed is derived by normalizing the extracted feature with the reference values $s_{m,n}^{SD}$ and $s_{m,n}^{SS}$, i.e.,

$sn_{m,n} = \left(s_{m,n} - s_{m,n}^{SD}\right) \big/ \left(s_{m,n}^{SS} - s_{m,n}^{SD}\right)$. Note that $sn_{m,n}$ is small for SDJPEG patches and large for SSJPEG patches.

4.2 Discriminative feature extraction based on all the DCT coefficients

The analysis in Section 4.1 shows that $sn_{m,n}$ for the SDJPEG image patches is smaller than those for the SSJPEG patches. Such phenomenon becomes more prominent with the increase of $q_{m,n}/\sigma_{m,n}^{SJR}$, and thus the DCT components with larger $q_{m,n}/\sigma_{m,n}^{SJR}$ would be more discriminative in SDJPEG detection. Our analysis in Section 3 shows that for a specific image, $\sigma_{m,n}^{SJR}$ can be estimated theoretically when QF_2 and the shift coordinates (x_S, y_S) are known. Figure 8 gives some examples of $\sigma_{m,n}^{SJR}$ $1 \le m, n \le 8$ with different QF_2 and (x_S, y_S).

Figure 8 shows the following: (i) Shifted JPEG compression with lower quality (QF_2) will introduce larger spread to the DCT coefficients. (ii) $\sigma_{m,n}^{SJR}$ for different DCT modes varies from each other and are not proportional to their corresponding quantization step. For instance, in Figure 8 (b2), the quantization steps for (2,1) and (2,2) are the same while $\sigma_{2,1}^{SJR}$ and $\sigma_{2,2}^{SJR}$ are quite different. (iii) Even for the same DCT mode, the variations caused by shift JPEG compression do not remain constant with different coordinate shifts, e.g., for the (2,3)th DCT coefficient, $\sigma_{2,3}^{SJR}$ is quite different when (x_S, y_S) changes between (4,4) and (1,6). Hence, $\sigma_{m,n}^{SJR}$ can be obtained by a table-lookup with the knowledge of QF_2 and (x_S, y_S). In order to have large value of $q_{m,n}/\sigma_{m,n}^{SJR}$, smaller
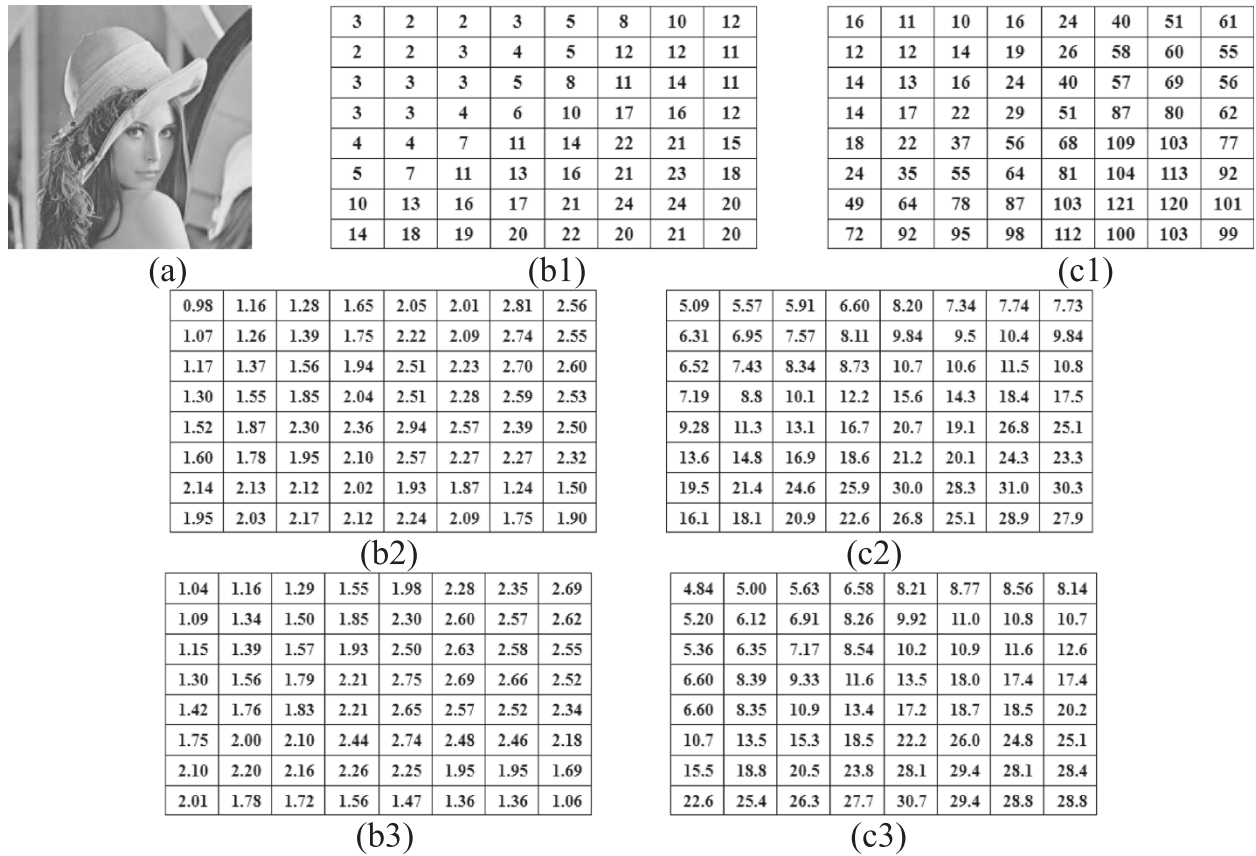
Figure 8 Examples of $\sigma_{m,n}^{\mathrm{SJR}}$ $1 \le m, n \le 8$, with different QF$_2$ and $(x_\mathrm{S}, y_\mathrm{S})$. **(a)** An example image Lena. **(b1)** The quantization matrix for QF$_2$ = 90. **(c1)** The quantization matrix for QF$_2$ = 50. **(b2, b3)** Estimated $\sigma_{m,n}^{\mathrm{SJR}}$ with QF$_2$ = 90, $(x_\mathrm{S}, y_\mathrm{S})$ = (4,4) and (1,6), respectively. **(c2, c3)** Estimated $\sigma_{m,n}^{\mathrm{SJR}}$ with QF$_2$ = 50, $(x_\mathrm{S}, y_\mathrm{S})$ = (4,4) and (1,6), respectively.

values of $\sigma_{m,n}^{\mathrm{SJR}}$ or larger values of $q_{m,n}$ are preferred. For any value of the quality factor, $Q = [q_{m,n}] = \lambda Q_{\mathrm{default}}$, where Q_{default} is the default quantization table defined by the Independent JPEG Group (IJG) and λ is a constant determined by the quality factor. Hence, we have $\frac{q_{m,n}}{\sigma_{m,n}^{\mathrm{SJR}}} = \lambda \times \frac{q_{\mathrm{default}}(m,n)}{\sigma_{m,n}^{\mathrm{SJR}}} = \lambda \times \mathrm{dis}_{m,n}$ $1 \le m, n \le 8$. It should be noted that in practice, only QF$_2$ is known. We will show how QF$_1$ can be estimated later.

The discriminative feature extraction considering all the DCT coefficients runs as follows:

1. Input the prior information: the image patch, the quality factor of the final compression QF$_2$, and the coordinate shift $(x_\mathrm{S}, y_\mathrm{S})$ (which is given by an exhaustive enumeration on 63 possible coordinate shifts, see Section 4.3).
2. According to the image patch, QF$_2$ and $(x_\mathrm{S}, y_\mathrm{S})$, estimate $\sigma^{\mathrm{SJR}} = \left[\sigma_{m,n}^{\mathrm{SJR}} \right]$ which is introduced by the shifted double JPEG compression. Calculate the

discriminative table DIS = [dis(m,n)] for low frequencies (where $m + n < 8$)

3. Sort all the DCT modes according to their discriminative value in descending order and the first N_c ($N_c = 3$ in our experiment) components are selected to construct the candidate set, i.e., $\{(m_1, n_1), (m_2, n_2), \ldots, (m_{N_c}, n_{N_c})\}$. Estimate the quantization step $q_{m_i,n_i}(\mathrm{QF}_1)$ (for the unknown QF$_1$) of these DCT modes by analyzing their histograms. It should be noted that if the coefficients of a DCT mode in the candidate set has too many zero values (>80% of the total number of coefficients), the quantization step cannot be estimated accurately. Hence, considering the quantization noise caused by shifted compression, for any (m_i, n_i) whose coefficients are concentrated in the range of

$$\left[-3 \times \sigma_{m_i,n_i}^{\mathrm{SJR}}, 3 \times \sigma_{m_i,n_i}^{\mathrm{SJR}} \right], \text{ i.e., } \int_{f=-3\times\sigma_{m_i,n_i}^{\mathrm{SJR}}}^{3\times\sigma_{m_i,n_i}^{\mathrm{SJR}}} h_{m_i,n_i}^{\mathrm{SD}}(f) \ge 80\%,$$

the mode is discarded and replaced with the mode with the next largest discriminative value outside the candidate set.

4. For the ith mode (m_i,n_i) in the candidate set, extract the normalized discriminative feature for the (m_i,n_i)th mode, i.e., sn_{m_i,n_i}, using the approach described in Section 4.1 with the estimated $\sigma^{\mathrm{SJR}}_{m_i,n_i}$ and $q_{m_i,n_i}(\mathrm{QF}_1)$. The final discriminative feature for the image patch, denoted by sn_{all}, is obtained by averaging over sn_{m_i,n_i} $1 \le i \le N_c$. In step 3, the quantization step $q_{m,n}(\mathrm{QF}_1)$ can be obtained by an exhaustive search among the reference SDJPEG's $sh^{\mathrm{SD}}_{m,n}(f)$ where $sh_{m,n}(f)$ is the most similar to. However, this approach is quite computationally expensive. Since $h^{\mathrm{SD}}_{m,n}(f)$ exhibits periodic-like pattern with a period of $q_{m,n}(\mathrm{QF}_1)$ to reduce the complexity, the approach similar to that in [17] is adopted, i.e., the fast Fourier transform is applied to the histogram $h^{\mathrm{SD}}_{m,n}(f)$, and the peak of the spectrum with the DC removed is extracted to estimate the quantization step of the (m,n)th DCT mode $q_{m,n}(\mathrm{QF}_1)$. With the quantization step of the (m,n)th DCT mode $q_{m,n}(\mathrm{QF}_1)$ estimated, the quality factor QF_1 can be estimated by comparing $q_{m,n}(\mathrm{QF}_1)$ with the default quantization table Q_{default}. In order to improve the robustness, all $q_{m,n}(\mathrm{QF}_1)$ values in the candidate set are estimated. The median value of the predicted QF_1 is taken as the quality factor of the first compression, and all $q_{m,n}(\mathrm{QF}_1)$ values in the candidate set are refined using the estimated quality factor.

4.3 Crop-and-paste image tampering detection by detecting the SDJPEG effects

In order to detect crop-and-paste image tampering, the JPEG image is divided into a series of $B \times B$ subimages. Each $B \times B$ subimage is examined to detect whether it contains any image patch having SDJPEG effects, which runs as follows:

(i) For a specific coordinate shift (x_S, y_S) $(0 \le x_S, y_S \le 7$ and $(x_S, y_S) \ne (0, 0))$, crop an image patch IMG_{x_S,y_S} from the subimage with the size of $(B - 8) \times (B - 8)$ and starting from (x_S, y_S).

(ii) With (x_S, y_S) and QF_2, extract the discriminative feature sn_{all} for IMG_{x_S,y_S}. Then the SDJPEG effect map (SEM) for (x_S, y_S) is set to sn_{all}, i.e., SEM $(x_S, y_S) = sn_{\mathrm{all}}$.

(iii) Repeat steps (i) and (ii) for all the 63 possible coordinate shifts to obtain the SEM for the $B \times B$ subimage.

(iv) Compare SEM with those of the positive (containing SDJPEG patches) and negative (not containing any SDJPEG patch) samples in the training database to detect whether the subimage has been tampered with. The Fisher's linear discriminant analysis (LDA) [24] is adopted as the classifier in our approach.

(v) Loop through steps (i) to (iv) for all $B \times B$ subimages in the image to detect the suspicious regions that might have been tampered with.

5 Experiments and discussion

In this section, we first investigate the effectiveness of several key issues for the proposed approach, i.e., the quantization error estimation for shifted JPEG compression, the primary quality factor estimation, and the proposed discriminative feature for SDJPEG detection. Then, we analyze the SDJPEG detection performance for image patches of various sizes and compared the detection performance with four state-of-the-art SDJPEG detection algorithms, i.e., Luo et al.'s [15] (Luo's in short), Chen and Hsu's [16] (Chen's in short), Bianchi and Piva's [13] (Bianchi I's in short), and Bianchi and Piva's [14] (Bianchi II's in short). Finally, we compare the image tampering detection performance of all five algorithms for two example images. The images in our experiments come from two widely used image databases, i.e., the UCID [25] and NRCS [26] image datasets, and all the original images are uncompressed.
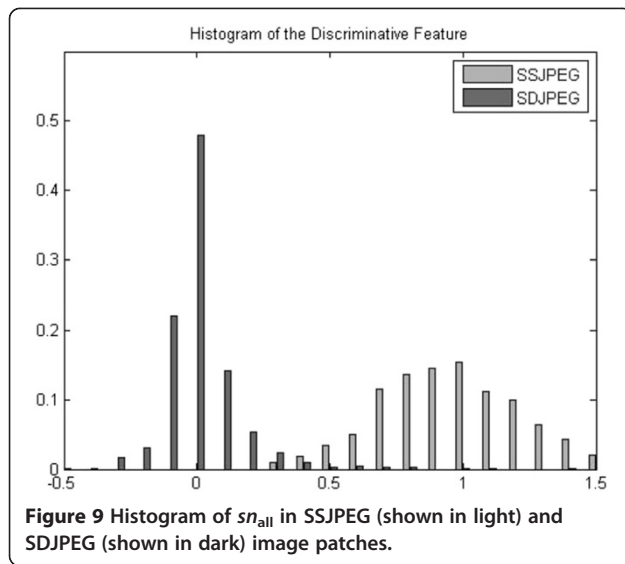
5.1 Effectiveness evaluation of the proposed approach
5.1.1 Quantization error estimation for shifted JPEG compression

In order to evaluate the effectiveness of the proposed approach to estimate $\sigma^{\mathrm{SJR}} = \left[\sigma^{\mathrm{SJR}}_{m,n}\right]$, 10,000 image patches with the size 256×256 are randomly collected from the

Table 3 Average estimation accuracy a1/a2/a3 of QF$_1$ with various patch sizes ((a1) 64 × 64; (a2) 128 × 128; (a3) 256 × 256), QF$_1$ and QF$_2$

QF$_1$	QF$_2$				
	50	60	70	80	90
50	0.62/0.95/0.99	0.78/0.97/0.97	0.90/1.00/1.00	0.95/1.00/1.00	0.96/1.00/1.00
60	0.48/0.54/0.59	0.62/0.82/0.95	0.75/0.97/0.98	0.92/1.00/1.00	0.93/1.00/1.00
70	0.13/0.14/0.13	0.36/0.52/0.61	0.65/0.85/0.92	0.79/0.99/0.98	0.92/1.00/1.00
80	0.13/0.12/0.10	0.14/0.13/0.12	0.44/0.53/0.62	0.68/0.85/0.94	0.82/0.98/0.96
90	0.08/0.08/0.09	0.12/0.14/0.13	0.14/0.13/0.14	0.44/0.55/0.59	0.69/0.82/0.89

Figure 9 Histogram of sn_{all} in SSJPEG (shown in light) and SDJPEG (shown in dark) image patches.

$$\eta_{\mathrm{SJR}} = \sum_{m=1}^{8} \sum_{n=1}^{8} \frac{\left|\sigma_{m,n}^{\mathrm{SJR}} - \sigma_{m,n}^{\mathrm{ACT-SJR}}\right|}{\left|\sigma_{m,n}^{\mathrm{ACT-SJR}}\right|}/64 \times 100\% \qquad (12)$$

The average relative estimation error η_{SJR} for the above image patches using the proposed approach is 10.61%, which is much less than that using the rough approximation in [13,14] (214.94%). Hence, the proposed DCT coefficient model can better describe the effects caused by shifted JPEG compression.

5.1.2 Primary quality factor estimation

To show the effectiveness of the proposed quantization step estimation approach, the following experiments have been carried out. For a specific patch size, QF_1 and QF_2, 500 SDJPEG patches are generated with random selection of (x_S, y_S) $((x_S, y_S) \neq (0, 0))$. The average estimation accuracy is given in Table 3.

From the table, it is observed that the estimation performance is improved with the increase of the patch size. It is because for larger size of the image patch, more data are collected to construct the histogram of the DCT coefficients and the period estimation approach will be more robust. On the other hand, the estimation performance is also improved with the increase of $QF_2 - QF_1$. Such phenomenon can be explained as follows. From (7) and Figure 5 (a2 and b2), for SDJPEG image patches, the period in the histogram (i.e., the primary quantization step $q_{m,n}(QF_1)$) is large for low quality factor QF_1 and the standard deviation of the quantization noise with rounding error, i.e., $\sigma_{m,n}^{\mathrm{SJR}}$, is small for high quality factor QF_2. Hence when $QF_2 - QF_1$ is large, i.e., $QF_2 - QF_1 \geq 10$, the Gaussian impulses in the histogram are non-overlapping or slightly overlapping and thus the period can be estimated accurately. However, when $QF_2 - QF_1$ is small, i.e.,

Table 4 Average classification accuracy, i.e., 1 − (FAR + FRR)/2, in percent with various values of QF_1 and QF_2 for patch size of 64 × 64

QF_1	QF_2				
	50	60	70	80	90
50	51.86/50.40/50.96	51.55/55.15/53.51	54.89/70.08/62.07	56.32/78.95/70.02	62.29/82.12/76.71
	50.90/**58.28**	51.56/**71.32**	54.29/**86.77**	56.58/**91.65**	63.01/**96.66**
60	51.31/50.26/49.34	50.57/50.87/52.10	51.40/55.24/56.21	53.92/70.50/62.51	59.02/78.39/71.51
	51.95/**54.05**	50.34/**56.96**	51.75/**69.80**	54.31/**90.71**	60.89/**92.17**
70	50.62/50.44/50.70	50.33/50.55/51.09	51.30/51.13/51.07	52.71/59.49/60.02	56.33/70.25/65.25
	50.80/**51.10**	49.60/**52.04**	50.45/**59.19**	52.63/**71.76**	57.22/**92.12**
80	50.42/**50.92**/50.47	49.62/50.63/**50.98**	51.62/50.42/50.09	50.49/51.91/51.77	54.34/58.41/59.39
	49.79/50.83	49.77/50.44	49.66/**53.49**	50.33/**56.28**	53.90/**75.50**
90	50.44/50.27/49.26	49.86/49.41/49.20	49.89/49.91/49.45	49.39/50.37/49.62	50.02/50.14/50.48
	49.13/**51.70**	**50.64**/50.39	49.87/**50.45**	50.48/**51.68**	50.26/**58.96**

Average classification accuracy, i.e., 1 − (FAR + FRR)/2, in percent, with various values of QF_1 and QF_2 for patch size 64 × 64. The accuracies a1/a2/a3/a4/a5 in the table are obtained by Luo's (a1), Bianchi I's (a2), Bianchi II's (a3), Chen's (a4), and the proposed method (a5), respectively. The best accuracies among the five methods investigated are boldfaced. Note that an accuracy of 50 is equivalent to random guess.

uncompressed images in the databases [25,26]. Then half of the image patches are used to generate the SSJPEG patches by shifted JPEG compression with the quality factor QF_2 randomly picked from 50 to 90 and random selection of (x_S, y_S) $((x_S, y_S) \neq (0, 0))$. The other half of the image patches are used to generate the SDJPEG by aligned JPEG compression with quality factor QF_1 randomly picked from 50 to 90 and shifted JPEG compression with quality factor QF_2 randomly picked from 50 to 90 and random selection of (x_S, y_S) $((x_S, y_S) \neq (0, 0))$. For each image patch, the actual standard deviation of the quantization error caused by shifted JPEG compression (denoted by $\sigma^{\mathrm{ACT-SJR}} = \left[\sigma_{m,n}^{\mathrm{ACT-SJR}}\right]$) is recorded as the ground truth, and the average relative estimation error is adopted to evaluate the estimation performance, i.e.,

Table 5 Average classification accuracy for patch size of 128×128

QF_1	QF_2				
	50	**60**	**70**	**80**	**90**
50	51.81/52.95/53.82	52.62/56.83/55.13	58.25/72.74/67.85	64.26/81.69/77.62	74.29/89.93/86.55
	51.88/**70.35**	52.30/**85.51**	59.59/**96.14**	65.94/**97.35**	75.15/**99.29**
60	50.91/51.96/51.45	51.33/52.02/51.37	54.12/61.65/59.72/	58.48/74.19/70.39	69.45/81.78/79.76
	51.29/**59.59**	51.48/**74.70**	55.08/**82.15**	58.44/**97.26**	70.78/**98.76**
70	50.00/50.58/**51.15**	51.26/50.05/50.53	52.64/53.55/53.97	53.76/64.37/64.07	62.79/76.49/78.42
	50.12/50.23	50.99/**58.90**	51.57/**72.26**	54.28/**95.48**	62.53/**98.57**
80	50.10/50.24/50.00	**51.48**/50.17/51.36	50.11/50.30/50.44	50.99/52.79/52.42	57.24/65.44/68.36
	50.28/49.76	50.67/49.96	50.05/**59.82**	51.00/**71.59**	58.44/**91.08**
90	50.25/49.36/49.58	**50.26**/49.65/49.02	49.47/49.19/50.25	49.86/52.09/52.67	52.46/53.31/53.25
	50.30/50.26	49.53/49.55	50.20/**50.93**	49.39/**57.92**	52.33/**72.54**

The accuracies a1/a2/a3/a4/a5 in the table are obtained by Luo's (a1), Bianchi I's (a2), Bianchi II's (a3), Chen's (a4), and the proposed method (a5), respectively. The best accuracies among the five methods investigated are boldfaced. Note that an accuracy of 50 is equivalent to random guess.

$QF_2 - QF_1 \leq -20$, the Gaussian impulses in the histogram are highly overlapping and thus the periodic pattern almost disappears in the histogram.

5.1.3 Effectiveness of the proposed discriminative feature

In order to investigate the effectiveness of the proposed discriminative feature sn_{all}, the distributions of sn_{all} for SSJPEG and SDJPEG patches with the size 256×256 are analyzed. For SSJPEG patches, the quality factor QF_2 is set to 70. For SDJPEG patches, the second compression quality factor QF_2 is also set to 70 and the primary compression quality factor QF_1 is randomly picked from 50 to 70. The coordinate shift (x_S, y_S) is also selected randomly in the range of $0 \leq x_S, y_S \leq 7$, $(x_S, y_S) \neq (0, 0)$. The histograms of sn_{all} in SSJPEG and SDJPEG patches are given in Figure 9. It is observed that sn_{all} is usually small

for SDJPEG patches while it is large for SSJPEG ones, which demonstrates that sn_{all} is effective in differentiating SDJPEG and SSJPEG patches.

5.2 SDJPEG patch detection

To assess the performance of SDJPEG effects detection, we prepare the dataset as follows:

i. Image patches of three sizes, i.e., 64×64, 128×128 and 256×256 are used.
ii. An overall 25 $\{QF_1, QF_2\}$ pairs are investigated with $QF_1, QF_2 = \{50, 60, 70, 80, 90\}$.
iii. For a specific $\{QF_1, QF_2\}$ pair, 10,000 image patches are randomly extracted from the uncompressed image database. The 'positive' samples are constructed by performing shifted JPEG

Table 6 Average classification accuracy for patch size 256×256

QF_1	QF_2				
	50	**60**	**70**	**80**	**90**
50	53.24/58.51/59.75	56.39/66.82/64.12	65.63/78.59/71.01	75.82/89.20/86.18	84.39/94.19/93.59
	52.84/**72.78**	57.54/**86.17**	67.88/**95.66**	77.05/**98.54**	86.72/**99.81**
60	50.91/53.52/52.40	55.02/60.36/57.57	58.23/65.21/63.58	67.78/78.18/71.37	80.62/89.98/88.14
	50.08/**60.94**	54.81/**81.80**	60.39/**84.75**	69.72/**98.00**	81.64/**99.78**
70	50.81/**51.82**/49.92	50.68/53.94/52.15	52.14/57.31/57.99	59.04/68.23/69.21	74.18/84.12/82.61
	49.97/50.13	50.00/**59.53**	53.63/**75.09**	61.30/**97.45**	75.50/**98.82**
80	49.62/**51.80**/51.42	49.39/**50.88**/49.50	50.04/50.48/50.31	53.55/58.32/59.16	64.82/72.41/71.58
	50.53/50.47	49.38/50.17	50.71/**60.92**	52.89/**77.11**	64.57/**96.87**
90	49.99/50.38/**50.42**	50.72/50.56/49.39	50.02/50.24/49.78	50.03/52.93/51.13	54.39/54.27/53.67
	50.26/49.34	**50.99**/50.27	49.67/**51.16**	49.66/**60.97**	54.85/**75.09**

The accuracies a1/a2/a3/a4/a5 in the table are obtained by Luo's (a1), Bianchi I's (a2), Bianchi II's (a3), Chen's (a4), and the proposed method (a5), respectively. The best accuracies among the five methods investigated are boldfaced. Note that an accuracy of 50 is equivalent to random guess.

Figure 10 Tampering detection results using our approach compared with two recent state-of-the-art approaches. (a) Another example of JPEG image forgery. **(b)** Ground truth of **(a)**, the tampered region are bounded by the highlighted contour. **(c1 to c3)** Tampered region detection by Bianchi I's method with $\{QF_1,QF_2\} = $ (c1) {50,90}, (c2) {50,70}, (c3) {50,50}, respectively. **(d1 to d3)** Tampered region detection by Bianchi II's method with $\{QF_1,QF_2\} = $ (d1) {50,90}, (d2) {50,70}, (d3) {50,50}, respectively. **(e1 to e3)** Tampering region detection by the proposed method with $\{QF_1,QF_2\} = $ (e1) {50,90}, (e2) {50,70}, (e3) {50,50}, respectively.

compression with quality factor of QF_1 and random coordinate shifts (x_S, y_S) $(0 \le x_S, y_S \le 7$ and $(x_S, y_S) \ne (0, 0))$ and then saving the image patches in JPEG format with QF_2. The 'negative' samples are constructed by directly saving the same uncompressed image patches in JPEG with QF_2.

Since in practice only QF_2 is available and QF_1 is unknown to an algorithm, the following are the settings for the different algorithms evaluated:

1) For Luo's [15] and Chen's [16] approaches, the SVM is adopted as the classifier. In the training

procedure, since QF_1 is unknown for each QF_2, a specific SVM is trained with the features extracted from half of the positive samples obtained from all possible selections of $QF_1 = \{50,60,70,80,90\}$ and the corresponding negative samples (there are overall 25,000 positive and 25,000 negative samples). In the testing procedure, the test samples are classified by their corresponding classifier with the knowledge of QF_2.

2) For Bianchi I's [14] approach, with the knowledge of QF_2, QF_1 is estimated by an EM algorithm. Then the tampered region likelihood is calculated for each 8×8 block in the image patch, and the investigated image patch is classified as a SDJPEG patch if half of the 8×8 blocks in the patch are with the likelihood greater than 1. For Bianchi II's [13] approach, with the knowledge of QF_2, QF_1 is achieved by an exhaustive search.

3) For the proposed approach, QF_1 is estimated by the method introduced in Section 4.2. For each QF_2, SEM features for 630 positive samples (ten samples for each possible (x_S, y_S) with QF_1 randomly selected) and 630 negative samples are adopted to train the classifier, and the rest (about 48,740 samples) are adopted for testing.

The detection performance obtained by our algorithm compared with the four existing methods is given in Tables 4, 5 and 6. For all the methods investigated, the classification performance is affected by two key parameters: the patch size and the difference in quality factor between the first and second JPEG compressions, i.e., $QF_2 - QF_1$. From the table, the following observations can be made.

First, the recognition accuracy increases substantially with the increase of patch size. As the features used in all five methods are based on statistical models, more data leads to better estimation of the models. Among all five approaches, our algorithm always outperforms the others especially when the patch size is small (64×64 for instance) since it requires less data to estimate a discriminative model. Since only one or several small image patches are modified in most image tampering detection tasks, our algorithm is more appropriate in such applications.

Second, the detection performance is improved with the increase of $QF_2 - QF_1$. It is because the lower quality factor of the first JPEG compression will leave more traces of the compression history and the higher quality factor of the second JPEG compression will introduce less distortion to the image, which makes the traces left by the first compression easier to be detected. The proposed approach can achieve reasonable accuracies (above 80%) when $QF_2 - QF_1 \geq 10$ and high accuracies (above 90%) when $QF_2 - QF_1 \geq 20$ for 128×128 and 256×256 patches. However, when $QF_2 - QF_1 \leq -20$, the detection

performance is very poor for all approaches, i.e., the classification accuracies are close to that of random guess (50%). The poor performance of the proposed approach is to be expected because when $QF_2 < QF_1$, $q_{m,n}/\sigma_{m,n}^{\text{SJR}}$ will be very small for all the DCT modes and $s_{m,n}^{\text{SD}}$ will be close to $s_{m,n}^{\text{SS}}$. In addition, as shown in subsection 5.1.2, with small values of $q_{m,n}/\sigma_{m,n}^{\text{SJR}}$, all the Gaussian components of $h_{m,n}^{\text{SD}}(f)$ in (10) will highly overlap with each other and the estimation of QF_1 will not be accurate especially when the patch size is small.

5.3 Crop-and-paste image tampering detection

Figure 10 illustrates the tampering detection results using our approach compared with two recent state-of-the-art approaches [13,14]. The forgery image is made by cropping the image patch (enclosed by the highlighted boundary as shown in Figure 10 (b)) from another JPEG image with quality factor of QF_1, pasting it to the original image and saving as the forgery JPEG file with quality factor of QF_2. Three different settings of $\{QF_1,QF_2\}$, i.e., $\{50,50\}$, $\{50,70\}$, and $\{50,90\}$, are investigated, and the detected tampered region is highlighted in the figures. Note that for all the three approaches, the corresponding parameters are adjusted to ensure that the false alarm rate (FAR) is less than 10%. Moreover, the detection results undergo a 3×3 median filtering process to remove isolate detection errors. It is observed from the figures that the proposed algorithm can accurately detect and locate tampered region when $\{QF_1,QF_2\} = \{50,70\}$ and $\{50,90\}$. It also accurately points out some suspicious regions when $\{QF_1, QF_2\} = \{50,50\}$. Compared with Bianchi and Piva's approaches [13,14], the proposed approach provides more accurate detection results due to (i) better estimation of the quantization error caused by shifted JPEG compression and (ii) more discriminative information investigated by the adaptive DCT mode selection approach in Section 4.1 compared with that only DC mode is investigated in [13].

6 Conclusions

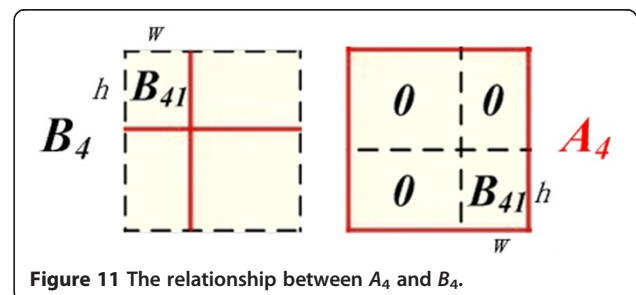In this paper, a new JPEG image tampering detection algorithm based on SDJPEG detection is presented. The

Figure 11 The relationship between A_4 and B_4.

DCT coefficient distribution for the SDJPEG patches has been expressed by a statistical model, and a discriminative feature has been proposed which can effectively differentiate between SDJPEG and SSJPEG patches. By an adaptive DCT mode selection scheme, several highly discriminative DCT modes are selected. We used several thousand SDJPEG and SSJPEG patches extracted from the UCID and NRCS image database to evaluate the performance of the proposed algorithm along with several existing algorithms. From the experimental results, the proposed algorithm can achieve much better results compared with the existing algorithms, especially when the patch size is small. We also performed experiments to detect and locate the tampered region of two test images, where the proposed algorithm always outperforms the other algorithms investigated. As a result, the proposed algorithm has provided a new and effective solution for SDJPEG detection and JPEG image forensics.

Appendix
Derivation of DCT coefficient from adjacent blocks

From Figure 4d, it is observed that block A can be decomposed into four non-overlapped region, i.e., A_1, A_2, A_3, and A_4, and $A = A_1 + A_2 + A_3 + A_4$. In order to derive $A_i = H_{i1}B_iH_{i2}$, we just demonstrate the situation when $i = 4$ and the same analysis can be applied to $i = 1,2,3$ (Figure 11).

For the bottom right corner, A_4 has the following relationship to B_4: $A_4 = H_{41}B_4H_{42}$, where $H_{41} = \begin{bmatrix} 0 & 0 \\ I_h & 0 \end{bmatrix}$ and $H_{42} = \begin{bmatrix} 0 & I_w \\ 0 & 0 \end{bmatrix}$. I_w and I_h are the identity matrices with size $h \times h$ and $w \times w$, respectively; h and w are the number of rows and columns extracted. Since all unitary orthogonal transforms such as the DCT are distributive to matrix multiplication [19], we have $D(A_4) = D(H_{41})D(B_4)D(H_{42})$

$$\Rightarrow D(A) = \sum_{i=1}^{4} D(A_i) = \sum_{i=1}^{4} D(H_{i1})D(B_i)D(_{Hi2}).$$

Author details
[1]School of Information Security Engineering, Shanghai Jiaotong University, No. 800, Dong Chuan Rd, Shanghai 200240, China. [2]School of Information and Communication Technology, Gold Coast Campus, Griffith University, Queensland QLD4222, Australia. [3]Department of Electric and Electronic Engineering, Shanghai Jiaotong University, Shanghai 200240, China.

References
1. X Wang, J Xue, Z Zheng, Z Liu, N Li, Image forensic signature for content authenticity analysis. J. Vis. Commun. Image. Repres. 23(5), 782–797 (2012)
2. A Swaminathan, Y Mao, M Wu, Robust and secure image hashing. IEEE. Trans. Info. Forensics. Sec. 1(2), 215–230 (2006)
3. A Phadikar, SP Maity, M Mandal, Novel wavelet-based QIM data hiding technique for tamper detection and correction of digital images. J. Vis. Commun. Image. Represent. 23(3), 454–466 (2012)
4. H Farid, A survey of image forgery detection. IEEE. Signal. Process. Mag. 2(26), 16–25 (2009)
5. AC Popescu, Statistical tools for digital image forensics. Ph.D. Thesis (Department of Computer Science, Dartmouth College, Hanover, NH, 2005)
6. AC Popescu, H Farid, Exposing digital forgeries in color filter array interpolated images. IEEE. Trans. Signal. Proc. 53(10), 3948–3959 (2005)
7. L Chen, W Lu, J Ni, W Sun, J Huang, Region duplication detection based on Harris corner points and step sector statistics. J. Vis. Commun. Image. Represent. In Press
8. J Lukas, J Fridrich, Estimation of primary quantization matrix in double compressed JPEG images, in Proceedings of Digital Forensic Research Workshop Cleveland, 2003), pp. 5–8
9. D Fu, YQ Shi, Q Su, A Generalized Benford's law for JPEG coefficients and its applications in image forensics, in Proceedings of the SPIE Electronic Imaging, Security and Watermarking of Multimedia Contents IX, vol. 650, 5th edn. (Springer, San Jose, CA, 2007), pp. 1L1–1L11
10. T Pevny, J Fridrich, Detection of double-compression in JPEG images for applications in steganography. IEEE. Trans. Info. Forensics. Sec. 3(2), 247–258 (2008)
11. H Farid, Exposing digital forgeries from JPEG ghosts. IEEE. Trans. Info. Forensics. Sec. 4(1), 154–160 (2009)
12. SM Ye, QB Sun, EC Chang, Detecting digital image forgeries by measuring inconsistencies of blocking artifact, in Proceedings of IEEE International Conference on Multimedia and Expo (China, Beijing, 2007). pp. 12–15
13. T Bianchi, A Piva, Detection of nonaligned double JPEG compression based on integer periodicity maps. IEEE. Trans. Info. Forensics. Sec. 7, 2 (2012)
14. T Bianchi, A Piva, Image forgery localization via block-grained analysis of JPEG artifacts. IEEE. Trans. Info. Forensics. Sec. 7, 3 (2012)
15. W Luo, Z Qu, J Huang, G Qiu, A novel method for detecting cropped and recompressed image block, in Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP'07), vol. 6, 2nd edn. (IEEE, Piscataway, Honolulu, Hawaii, pp. 217–220
16. YL Chen, CT Hsu, Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection. IEEE. Trans. Info. Forensics. Sec. 6(2), 396–406 (2011)
17. Z Lin, J He, X Tang, CK Tang, Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis. Pattern. Recogn. 42(11), 2492–2501 (2009)
18. FJ Huang, JW Huang, YQ Shi, Detecting double JPEG compression with the same quantization matrix. IEEE. Trans. Info. Forensics. Sec. 5(4), 848–856 (2010)
19. SF Chang, DG Messerschimitt, Manipulation and compositing of MC-DCT compressed video. IEEE. J. Select. Areas. Commun. 13(1), 1–11 (1995)
20. R Reininger, J Gibson, Distributions of the two-dimensional DCT coefficients for images. IEEE. Trans. Commun. 31(6), 835–839 (1983)
21. Z Fan, RL de Queiroz, Identification of bitmap compression history: JPEG detection and quantizer estimation. IEEE. Trans. Image. Process. 12(2), 230–235 (2003)
22. AC Popescu, H Farid, Statistical tools for digital forensics. Lect. Notes. Comput. Sci. 3200, 395–407 (2005)
23. W Luo, J Huang, G Qiu, JPEG error analysis and its application to digital image forensics. IEEE. Trans. Info. Forensics. Sec. 5(3), 480–491 (2010)
24. RO Duda, PE Hart, DG Stork, Pattern Classification, 2nd edn. (Wiley-Interscience, USA, 2000)
25. G Schaefer, M Stich, UCID: an uncompressed color image database, in Proceedings of the SPIE Storage and Retrieval Methods and Applications for Multimedia, vol. 530, 7th edn. (Springer, San Jose, CA, 2003), pp. 472–480
26. NRCS Photo Gallery, 2005. http://photogallery.nrcs.usda.gov/res/sites/photogallery/