Detecting Double Compressed JPEG Images

Babak Mahdian*, Stanislav Saic

Institute of Information Theory andAutomation of the ASCR, Czech Republic
{mahdian,ssaic}@utia.cas.cz

Keywords: Image forensics, image forgery detection, authentication, double jpeg compression.

Abstract

Verifying the integrity of digital images and detecting the traces of tampering without using any protecting pre–extracted or pre–embedded information has an important role in image forensics and crime detection. When altering a JPEG image, typically it is loaded into a photo–editing software and after manipulations are carried out, the image is re–saved. This operation, typically, brings into the image specific artifacts. In this paper we focus on these artifacts and propose an automatic method capable of detecting them.

1 Introduction

Verifying the integrity of digital images and detecting the traces of tampering without using any protecting pre–extracted or pre–embedded information have become an important and hot research field of image processing [11, 12, 4, 9, 8, 2].

The digital information revolution and issues concerned with multimedia security have generated several approaches to tampering detection. Generally, these approaches could be divided into active and passive–blind approaches. The area of active methods simply can be divided into the data hiding approach and the digital signature approach.

By data hiding we refer to methods embedding secondary data into the image. The most popular group of this area belongs to digital watermarks. Digital watermarking assumes an inserting of a digital watermark at the source side (e.g., camera) and verifying the mark integrity at the detection side. A major drawback of watermarks is that the they must be inserted either at the time of recording the image, or later by a person authorized to do so. This limitation requires specially equipped cameras or subsequent processing of the original image.

In this work, we focus on detecting double compressed jpeg images. When altering a JPEG image, typically it is loaded into a photo–editing software (decompressed) and after manipulations are carried out, the image is re–saved (compressed again). The quantization matrix of the unaltered image is called as primary quantization matrix. The quantization matrix of the re–saved image is called as secondary quantization matrix. If the primary and secondary quantization matrix are not identical, then the re–saving (double compressing) operation can bring into the image specific changes. Detecting these changes plays a valuable role in identifying image forgeries. Detecting the traces of double compression also is helpful in other research fields such as steganography [4]. Here, double–compressed images can be produced by some steganographic algorithms.

It is important to note that detecting the traces of double compression does not necessarily imply the existence of malicious modifications in the image. Often images are re–compressed due to reduce the image storage size or transmission time. Furthermore, the image could undergo only simple image adjustment operations such as contrast enhancing.

2 Related Work

So far, a number of methods dealing with detecting of double JPEG compression have been proposed. A. C. Popescu and H. Farid [10] proposed a technique examining the Fourier transform of the histograms of the DCT coefficients. In [3], J. Lukáš and J. Fridrich presented a paper for estimation of primary quantization matrix from a double compressed JPEG image. The paper discusses three different approaches from which the method based on neural network classifiers is the most effective one. The other two methods are based on histogram matching. T. Pevný and J. Fridrich [4] proposed a method based on support vector machine classifiers with feature vectors similar to [3]. Feature vectors are formed by features formed by histograms of multiples of quantization steps. In [13], Z. Qu et al. formulated the shifted double JPEG compression as a noisy convolutive mixing model to identify whether a given JPEG image has been compressed twice with inconsistent block segmentation. D. Fu and Y. Q. Shi. [5] proposed a statistical model based on Benford's law for the probability distributions of the first digits of the quantized JPEG coefficients. W. Luo et al. [7] proposed a method for detecting cropping and re–compressed image blocks based on JPEG blocking artifact characteristics. J. He et al. [6] used the double quantization effect hidden among the DCT coefficients to automatically detect the doctored parts of images. C. Chen et el. [1] proposed a double jpeg detecting method based on transition probability matrix and support vector machines.

By detailed analysis and examination of the proposed methods, we can notice that most of them are not suitable for real–life conditions. Many of them work with the entire quantization matrix (or with an impartible subset of the matrix) and use a statistical–learning method for the classification part. This approach generates very accurate results for a finite set of quantization tables. This condition can only be satisfied in laboratory conditions. Real–life images come from uncontrolled conditions with different quantization matrices. It is not possible to train the proposed methods for all possible quantization matrices. Existing methods mostly tried to rely on JPEG standard quantization tables and train their classifiers using these matrices. But, when applying a statistical–learning based double JPEG detection method trained in such a way to images from non–laboratory conditions, the detection accuracy rapidly decreases.

Based on these reasons, we prefer to focus on a method capable of generalization to real–life conditions. The core of the method proposed in [10] satisfies this condition.

3 Basics of JPEG Compression

Typically, the image is first converted from RGB to YCbCr, consisting of one luminance component (Y), and two chrominance components (Cb and Cr). Mostly, the resolution of the chroma components are reduced, usually by a factor of two. Then, each component is split into adjacent blocks of 8×8 pixels. Blocks values are shifted from unsigned to signed integers. Each block of each of the Y, Cb, and Cr components undergoes a discrete cosine transform (DCT). Let $f(x, y)$ denotes a 8×8 block. Its DCT is:

$$F(u,v) = \frac{1}{4}C(u)C(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) cos\frac{(2x+1)u\pi}{16} cos\frac{(2y+1)v\pi}{16}, \quad (1)$$

where

$$\begin{aligned} &(u, v \in \{0 \cdots 7\}); \\ &C(u), C(v) = 1/\sqrt{2} \quad \text{for} \quad u, v = 0; \\ &C(u), C(v) = 1 \qquad \text{otherwise}. \end{aligned} \quad (2)$$

In the next step, all 64 $F(u, v)$ coefficients are quantized. Then, the resulting data for all blocks is entropy compressed typically using a variant of Huffman encoding.

The quantization step is performed in conjunction with a 64–element quantization matrix, $Q(u, v)$. Quantization is a many–to–one mapping. Thus it is a lossy operation. Quantization is defined as division of each DCT coefficient by its corresponding quantizer step size defined in the quantization matrix, followed by rounding to the nearest integer:

$$F^Q(u,v) = round(\frac{F(u,v)}{Q(u,v)}), \quad u, v \in \{0 \cdots 7\} \quad (3)$$

Generally, the JPEG quantization matrix is designed by taking the visual response to luminance variations into account, as a small variation in intensity is more visible in low spatial frequency regions high spatial frequency regions.

The JPEG decompression works in the opposite order: entropy decoding followed by dequantization step and inverse discrete cosine transform.

4 Double JPEG Quantization and its Effect on DCT coefficients

By double JPEG compression we understand the repeated compression of the image with different quantization matrices Q_α (primary quantization matrix) and Q_β (secondary quantization matrix). The DCT coefficient $F(u, v)$ is said to be double quantized if $Q_\alpha(u, v) \neq Q_\beta(u, v)$. The double quantization is given by:

$$F^{Q^\beta}(u,v) = round(\frac{F^{Q^\alpha}(u,v)Q^\alpha(u,v)}{Q^\beta(u,v)}) \quad (4)$$

To illustrate the effect of double quantization, consider a set of random values in the range of $\langle -50, 50\rangle$ drawn from a normal zero–mean distribution (see Figure 1(a)). Figure 1 (b) shows the distribution after being quantized with quantization step $Q^\alpha = 3$. Figure 1(c) shows the same distribution after being double quantized with quantization steps $Q^\alpha = 3$ and $Q^\beta = 3$. In other words, Figure 1 (c) was generated by quantization of the distribution by quantization step $Q^\alpha = 3$. Then obtained values were de–quantized using $Q^\alpha = 3$ (so, now each value of the distribution is a multiple of the 3). In the end, values were quantized again using the quantization step $Q^\alpha = 2$. Apparently, the distribution of the doubly quantized values contains periodic empty bins. This is caused by the fact that in the second quantization values of the distribution are re–distributed into more bins than in the first quantization.

Generally, the double quantization process brings detectable artifacts like periodic zeros and double peaks. The double quantization effect has been studied in [10, 3, 6]. Therefore, for a more detailed analysis of double quantization effect, we refer you to any of these publications.

5 Detecting Double Quantization Effect

Last section briefly described the effect of double quantization. In this section, our main aim will be the detection of the traces of double quantization in JPEG images. To achieve this goal, we use the fact that the histograms of DCT coefficients of a double compressed image contain specific periodic artifacts detectable in the frequency space.

In the double JPEG detection method proposed in [10], authors compute the zero–mean histograms of DCT coefficients corresponding to low frequencies. For each of these histograms the magnitudes of their Fourier transforms are obtained. If DCT coefficients corresponding to DCT frequency (u, v) are double quantized, the corresponding histogram and Fourier transform has a specific behavior.

See Figure 2 for some examples of the method's output. Figures 2 (b) and (c) show the typical Fourier transforms of the zero–mean histograms of DCT coefficients corresponding to frequencies $(0, 0)$ and $(1, 1)$. Here the method was applied to a single compressed version of Figure 2 (a). Figure 2 (a) was saved by quality factor 85 (using standard JPEG luminance and chrominance quantization tables). Figures 2 (d) and (e) show method's outputs for DCT frequencies $(0, 0)$ and $(1, 1)$ of a double compressed version of Figure 2 (a). Here the image

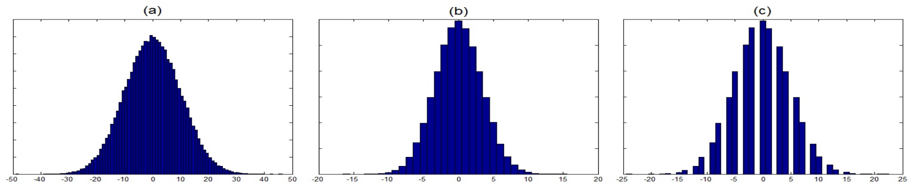

Figure 1. Shown are: (a) the histogram of a non–quantized random values drawn from a zero–mean normal distribution; (b) histogram of the quantized (a) with quantization step $Q^{\alpha} = 3$; (c) histogram of the double–quantized (a) with quantization steps $Q^{\alpha} = 3$ and $Q^{\beta} = 2$.

was saved by quality factor 85 followed by quality factor 75. Figures 2 (f) and (g) show the same for the double compressed version of Figure 2 (a) with primary quality factor 85 followed by secondary quality factor 80. Note that double compression artifacts generate a specific behavior in the method's output.

Specifically, if the image is double compressed, typically the output of the method applied to the DC component contains a specific clear peak (for example, see Figure 2 (c)). Otherwise, there is no strong peak in the spectrum (Figure 2 (b)). When the method is applied to a singe-quantized AC component, the spectrum has a decaying trend (Figure 2 (e)). Otherwise, in some parts, the spectrum has a local ascending trend (Figure 2 (g)).

To determine the presence of double compression artifacts, authors proposed a threshold–based quantitative measure. The magnitudes of FFT of DCT histograms are approximated by a two–parameter generalized Laplace model. The model parameters are estimated through non-linear minimization. The difference of the approximation and the actual FFT magnitudes are used for the threshold–based quantitative measure. If any of the FFT magnitudes is found as double quantized, then the image is classified as a double JPEG compressed image.

As pointed out in [10], the patterns introduced by double JPEG compression depend on particular compression quality parameters. So, using the method's output (peak positions), it is also possible to estimate the compression qualities that have been used in the primary compression process.

Despite the significant advantages of the method presented in [10], our experiments discovered that there are some drawbacks in the method. For example, applying the method to natural images with "non–perfect histograms" (histograms which have not a well decaying trend or histograms which being not perfectly approximated by a Gaussian or Laplacian) causes false positives (single compressed imaged classified as double compressed). For an example, see Figure 3 (a). We also noticed that the number of false positives is rapidly increasing when the method is applied to ün-naturalïmages (for example, scanned paper forms). Furthermore, we found out that application of a machine learning–method such as SVM improves the method's results.

Because of the mentioned reasons, the application of [10] to our test image set resulted in a big number of false–positives and missing a solid number of double compressed images. These reasons lead us to develop a more powerful method.

Similar to [10], our method computes the magnitudes of FFT of the histograms of the DCT coefficients corresponding low frequencies. Specifically, the following DCT frequencies are employed: $(0, 0)$, $(1, 0)$, $(2, 0)$, $(3, 0)$, $(0, 1)$, $(1, 1)$, $(2, 1)$, $(0, 2)$, $(1, 2)$ and $(0, 3)$. Because of the problem with insufficient statistics for high–frequency DCT coefficients (high frequency DCT coefficients are often quantized to zeros), other frequencies are not considered. Only the first half of the spectrum is considered. We denote the result of this part by $|H_1| \cdots |H_{10}|$, where $|H_1|$ corresponds to DC component and $|H_i|, i = 2 \cdots 10$, correspond to AC components. Furthermore, $|H_i|$ are normalized to have a unit length.

To reduce the number of false positives, before computing the FFT, the margin parts of the histograms are eliminated and not employed for further analysis. The reason is that images consisting of a poor number of colors have a non-typical behavior in the margin parts of the histograms leading to false positives.

Furthermore, we found out that many of false positives produced by [10] can be eliminated by considering only the magnitudes of DCT coefficients corresponding to AC components. Please, see Figure 3 (b). This modification significantly improves our output.

Only the luminance channel is employed to detect the double JPEG compression artifacts. The reason is that the two chrominance channels of a JPEG compressed image are typically down–sampled by a factor of two or four, and quantized using larger quantization steps. Thus, the histograms obtaining from these components contain only little information valuable for detecting the presence of double compression.

As mentioned previously, typically, $|H_2| \cdots |H_{10}|$ have a decaying trend. To be able to effectively compare and analyze different histograms, this trend should be removed. To achieve this, [10] computes the difference of the actual $|H_i|$ and its approximation based on a relatively complex two–parameter generalized Laplace model and a least squares optimization for parameters estimation. We replaced this step by a computationally much simpler and implementationally much faster way. We employed a simple local minimum subtraction operation resulting in removing the decaying trend and preserving local peaks.

First, $|H_i|, i = 2 \cdots 10$, are de–noised using an averaging filter. Then, from each frequency f of $|H_i|(f), i = 2 \cdots 10$, the minimum value of its neighbor frequencies is subtracted. Only

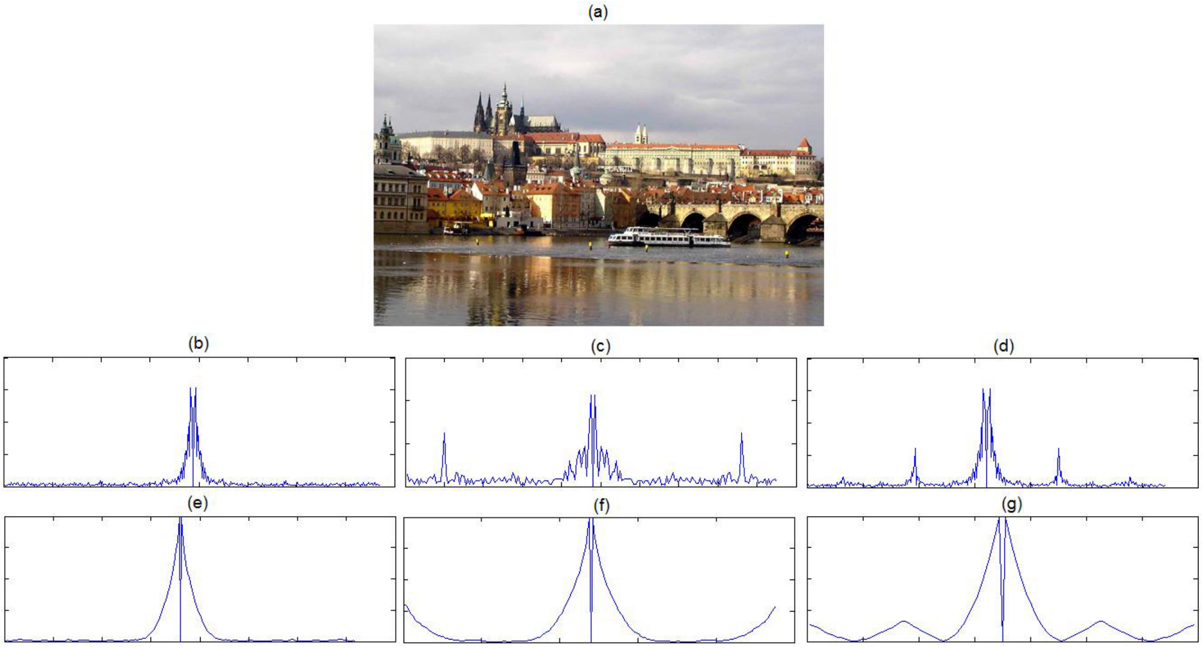

Figure 2. In (a) the test image is shown. (b) and (e) show the magnitudes of Fourier transform of the zero–mean histograms of DCT coefficients corresponding to frequencies $(0, 0)$ and $(1, 1)$ obtained from a single compressed version of (a). Here the image was saved by quality factor 85. (c) and (f) show the same for the double compressed version of (a). Here the image was saved by quality factor 85 followed by quality factor 75. (d) and (g) show the same for double compressed version of (a) with quality factor 85 followed by quality factor 80.

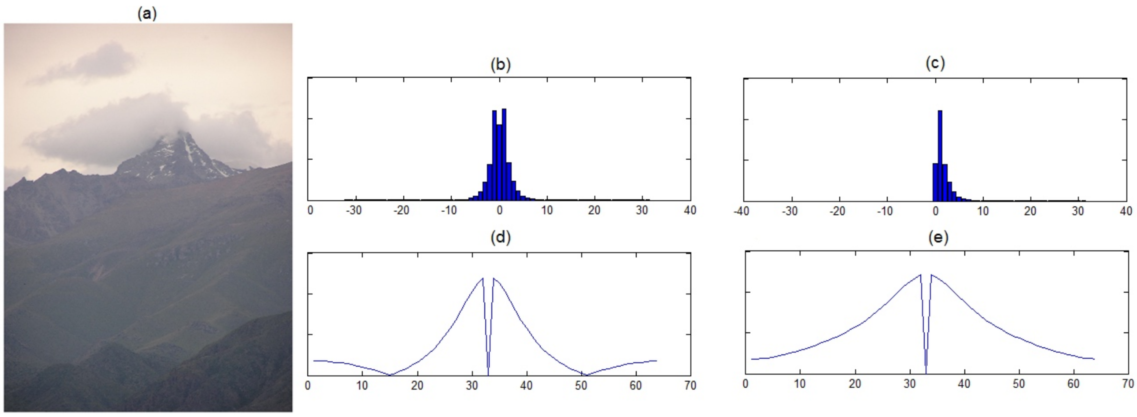

Figure 3. Shown are: (a) the tested single compressed images; (b),(d) outputs of the method described in [10] resulting in a false positive (the spectrum has not the typical decaying trend); (c),(e) output of the method described in this paper. In both cases the method was applied to DCT coefficients corresponding to frequency $(1, 0)$.

the neighbor frequencies in direction to the DC component are considered. More formally,

$$|\widetilde{H}_i|(f) = |H_i|(f) - M_i(f), \tag{5}$$

where $M_i(f)$ is the minimum value of $\{|H_i|(f) \cdots |H_i|(f - n)\}$, where $n \in \mathcal{N}_0$ denotes the length of the minimum filter.

Since the histograms of DCT coefficients undergone quantization with a quantization step $Q_1(u, v)$ differ from histograms of DCT coefficients undergone quantization with a step $Q_2(u, v)$ (where $Q_1(u, v) \neq Q_2(u, v)$), the size of the minimum filter n has a different value for different quantization steps. The value n is determined in the training process regarding to the desired detection accuracy and false positives rates.

Before going on, it is important to note that not all combinations of $Q_\alpha(u, v)$ and $Q_\beta(u, v)$ brings into the DCT histograms double quantization artifacts. If $\frac{Q_\beta(u,v)}{Q_\alpha(u,v)}$ is an integer value, the specific double quantization artifacts are not introduced into the histograms of DCT coefficients corresponding to frequency (u, v).

6 Classification

Last section resulted in $|\widetilde{H}_i|$, where $i = 1 \cdots 10$. The quantization step $Q(u, v)$ corresponding to $|\widetilde{H}_i|$ can be determined directly from the quantization table in the JPEG file. We use this fact and construct one separate classifier for each quantization step of interest distinguishing between two classes: single compressed and double compressed $|\widetilde{H}|$. When classifying $|\widetilde{H}_i|$, the corresponding classifier is used (the value of $Q(u, v)$ determines the classifier).

Let us assume that we want to build the classifier for a quantization step q, where $q \in \mathcal{N}$. Let us assume that P_q contains normalized positions of peaks in $|\widetilde{H}|$ corresponding to double quantization. Please note that P_q can easily be generated by computing $|\widetilde{H}|$ of a random signal having a uniform distribution and being double quantized with step q and primary step q_α, where $q_\alpha = 1 \cdots n$, $q \neq q_\alpha$.

The feature vector, v_i corresponding to $|\widetilde{H}_i|$, is constructed by taking the values of $|\widetilde{H}_i|$ in peak positions.

Our training set consists of 2000 uncompressed images (different kinds of images with narrow, wide, typical, untypical intensity histograms). Half of the images is employed for the training purposes and the second half for testing purposes.

As aforementioned, the classifier will be used for testing DCT coefficients quantized with the quantization step $Q_\beta(u, v) = q$. In other word, in order to train classifier for quantization step q, we need both single quantized DCT coefficients (with quantization step q) and DCT coefficients double quantized with the secondary quantization step q. To obtain single quantized coefficients, 1000 uncompressed images where compressed using the quantization step q. To obtain double–compressed feature vectors, non-compressed images were first JPEG compressed using the quantization step q_α and the re–compressed using q. Only q_α which brings detectable peaks into $|\widetilde{H}|$ were employed. Only DCT coefficients corresponding to DC component and AC component $(1, 0)$ were used for the training purposes.

Our classifiers are soft–margin support vector machines (SVM) with the the Gaussian kernel $k(x, y) = exp(-\gamma||x - y||^2)$. The false positive rate was controlled to be 1 percent. In our experiments we trained classifiers for quantization steps $1 \cdots 25$.

To test the method, we compressed 1000 images resulting in single JPEG compressed images (using quality factor Q_α and JPEG standard quantization matrix). Then each single compressed image was re–compressed using a quality factor Q_β, resulting in a double JPEG compressed image. Detection accuracies are reported in Table 1.

As is apparent from Table 1, the detection accuracies are higher for $Q_\beta > Q_\alpha$. When $Q_\beta < Q_\alpha$, generally, the DCT coefficient histograms have a shorter support. Furthermore, the introduced periodic properties have a larger period (due to the fact that $Q_\beta(u, v) > Q_\alpha(u, v)$). In some cases (for example, $Q_\alpha = 95$ and $Q_\beta = 70$), the detection accuracy is almost zero. This is because the fact that $\frac{Q_\beta(u,v)}{Q_\alpha(u,v)}$ is an integer value for employed DCT frequencies.

Generally, the content of the image also has an important impact on detecting the traces of double JPEG. Images containing heavy textures or images containing large uniform regions have different properties in their histograms of DCT coefficients comparing to natural images. Unfortunately, we do not have available the test set used in [10] allowing a direct comparison.

Though, it is beyond the scope of this paper, we briefly mention that the generated feature vectors also are very helpful in estimating the primary quantization matrix. One approach to estimate the primary matrix can be using of a SVM one–against–one strategy. In another approach, we used a k nearest neighbors search method (in a kd-tree space) with very good results.

7 Discussion

Detecting double JPEG compressed images plays an important role in image forensics and crime detection. In this paper we proposed a detection method based on histograms of DCT coefficients and SVM. When comparing our method to [10], almost–always when the image is double compressed and contains detectable artifacts, then both methods work well and detect the double compression. Nontheless, the method proposed in this paper produces a significantly less number of false positives.

Acknowledgments

This work has been supported by the Czech Science Foundation under the project No. GACR 102/08/0470.

References

[1] Chunhua Chen, Yun Q. Shi, and Wei Su. A machine learning based scheme for double jpeg compression detection. In *ICPR*, pages 1–4, 2008.

[2] T. Filler, J. Fridrich, and M. Goljan. Using sensor pattern noise for camera model identification. In *Proceedings IEEE, International Conference on Image Processing, ICIP 2008*, pages 1296–1299, San Diego, CA, October 12–15, 2008.

[3] J. Fridrich and J. Lukas. Estimation of primary quantization matrix in double compressed jpeg images. In *Proceedings of DFRWS*, volume 2, Cleveland, OH, USA, August 2003.

[4] J. Fridrich and T. Pevny. Detection of double–compression for applications in steganography. *IEEE Transactions on Information Security and Forensics*, 3(2):247–258, June 2008.

[5] Dongdong Fu, Yun Q. Shi, and Wei Su. A generalized benford's law for jpeg coefficients and its applications in

Table 1. Detection accuracy [%] as a function of different JPEG compression factors.

$Q_\beta \backslash Q_\alpha$	50	55	60	65	70	75	80	85	90	95
50	–	56	97	90	77	20	86	26	14	1
55	88	–	68	98	96	82	96	47	47	1
60	99	93	–	98	98	93	69	97	69	27
65	100	100	96	–	93	98	62	92	35	35
70	100	100	100	86	–	98	95	71	44	1
75	100	100	100	100	98	–	95	95	81	2
80	100	100	100	100	100	100	–	95	93	1
85	100	100	100	100	100	100	100	–	96	95
90	100	100	100	100	100	100	100	100	–	99
95	100	100	100	100	100	100	100	100	100	–

image forensics. In *SPIE Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents*, San Jose, CA, USA, January 2007.

[6] Junfeng He, Zhouchen Lin, Lifeng Wang, and Xiaoou Tang. Detecting doctored jpeg images via dct coefficient analysis. In *ECCV (3)*, pages 423–435, 2006.

[7] Weiqi Luo, Zhenhua Qu, Jiwu Huang, and Guoping Qiu. A novel method for detecting cropped and recompressed image block. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 217–220, Honolulu, HI, USA, April 2007.

[8] B. Mahdian and S. Saic. Blind authentication using periodic properties of interpolation. *IEEE Transactions on Information Forensics and Security*, 3(3):529–538, September 2008.

[9] B. Mahdian and S. Saic. Using noise inconsistencies for blind image authentication. *Image and Vision Computing*, 27(10):1497–1503, 2009.

[10] A. C. Popescu. *Statistical Tools for Digital Image Forensics*. PhD thesis, Ph.D. dissertation. Department of Computer Science, Dartmouth College, Hanover, NH, 2005.

[11] A.C. Popescu and H. Farid. Exposing digital forgeries by detecting traces of re-sampling. *IEEE Transactions on Signal Processing*, 53(2):758–767, 2005.

[12] A.C. Popescu and H. Farid. Exposing digital forgeries in color filter array interpolated images. *IEEE Transactions on Signal Processing*, 53(10):3948–3959, 2005.

[13] Zhenhua Qu, Weiqi Luo, and Jiwu Huang. A convolutive mixing model for shifted double jpeg compression with application to passive image authentication. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4244–1483, Las Vegas, USA, April 2008.