International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.11, (2016), pp.341-352
http://dx.doi.org/10.14257/ijsip.2016.9.11.31

Blind Separation of Tampered Image Based on JPEG Double Quantization Effect

Duan Xintao[1,2], Peng Tao[1], Huang Jingjing[3], Li feifei[1] and Wang Jingjuan[1]

[1]Institute College of Computer and Information Engineering, Henan Normal University, Xinxiang 45300)
[2]Henan Province Engineering Technology Research Center for Computing Intelligence & Data Mining, Xinxiang 453007
[3]Departmen of Electronic Engineering , Henan Economic and Trade Vocational College, Zhengzhou 450000)
duanxintao@126.com

Abstract

The double quantization effect of JPEG provides important clue for detecting image tampering. Whenever an original JPEG image has undergone a localized tampering and saved in JPEG again, the DCT coefficients of the areas without tampering will be compressed for twice while the tampered areas only suffered once. The Alternating Current (AC) coefficient distribution accord with a Laplace probability density distribution described with parameter λ *. This paper proposed a new double compression probability model of JPEG image to describe the change of DCT coefficients' statistical properties after the double compression. According to Bayes' theorem, using the posterior probability, the model can also show the eigenvalues of the double and single compressed block. We assign a dynamic adaptive threshold for the eigenvalues with the Particle Swarm Optimization Algorithm. Then the tampered region is detected and separated automatically by using the threshold. The experimental results show that the method can detect and separate the tamped area effectively and it outperforms other algorithms in terms of the detection result especially when the second compression factor is smaller than the first one. Compared with other traditional methods, the proposed approach could effectively separate the tampered regions from the tampered image without respect to the location, size and number of tampered images.*

Keywords: *blind separation, double quantization effect of JPEG, image tampering, Laplace distribution, dynamic adaptive threshold, Particle Swarm Optimization*

1. Introduction

JPEG (Joint Photographic Experts Group) is the mainstream compression standard in the current, which has a higher compression ratio and widely used in multimedia and networking programs. With the development of image editing software, for such an image, it is easy to change the real content. And it's difficult to discern whether the image has been tampered only rely on the human eyes. In this case, the authenticity of the image has become a problem we are concerned. Therefore, the study of JPEG images forensics has very important significance. In general, the digital image forensics can be divided into active forensics and passive forensics (blind image forensics). Active forensics technology [1] is to embed fragile digital image watermarking in advance, to obtain evidence by extracting watermark and signature. In contrast, passive forensics technology does not require any prior knowledge of the image. We can do forensics research only rely on the image itself. Thus, forensics research technology has a higher value, but the

ISSN: 2005-4254 IJSIP

evidence collection difficulty is much greater than active forensics. Aiming at this kind of image blind forensics, the research of domestic and foreign scholars have developed all sorts of blind forensics algorithm. In [2], Zhigang Fan and Ricardo Queiroz have proposed a method to detect whether an image has experienced the JPEG compression. Yu L.Chen and Chiou T.Hsu [3] have proposed a model by quantization noise to detect whether a JPEG image has undergo a JPEG compression or double compression. In [4-5], the author put forward by estimating the quantization matrix of the first time for the JPEG images experience double compression JPEG image. In the literature [6-8], the authors through estimate the original quantization matrix to the JPEG images which have been undergone a second JPEG compression, and this method successful detection and separate the tampered regions from the tampered image. Tomas Pevny and Jessica Fridrich [9] proposed a method based on knowledge of the vector product with the low-frequency DCT coefficients histogram trained to detect whether the image has undergo a double JPEG compression.

However, most of the existing requirements of forensics algorithms are mostly uncompressed image to be detected or higher compression factor image and JPEG image tamper detection algorithm can be widely applied in the still relatively small. For these reasons, we are more focused on the case examined under realistic conditions. The Alternating Current (AC) coefficient distribution can accord with a Laplace probability density distribution described with a parameter λ. In this paper, this paper proposed a new double compression probability model of JPEG image to describe the change of DCT coefficients' statistical properties before and after the double compression. According to Bayes criterion, using the posterior probability, the model also can show the eigenvalues of the double and single compressed block. We assign a threshold for the eigenvalues. Then the tampered region is automatically detected and extracted by using the threshold to classify the eigenvalues. The experimental results show that the method can detect and locate the tamped area effectively and it outperforms other algorithms in terms of the detection result especially when the second compression factor is smaller than the first one.

2. JPEG Compression Principle

In the standard JPEG compression scheme [10-11], consists of the following basic steps, Figure 1 shows the JPEG compression and decompression process flow diagram:

(1)The preprocessing stage, a color image (RGB) first mapped to a YCbCr color space, including a luminance channel (Y) and two chroma channels (Cr, Cb). The two chroma channels are usually obtained by the two sampling of the luminance channel factor.

(2)Discrete cosine transform, each channel of the pixel is divided into a sub image block without overlapping. These pixel values from unsigned integer into a signed integer (from [0 255] into [-128 127]). For each image sub block of two-dimensional discrete cosine transform (DCT). A pixel value of an 8×8 image block represented by $f(x,y)$. The DCT transform can be expressed as:

$$F(u,v)=\frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16},(u,v\in\{0\cdots7\} \tag{1}$$

$$C(u),C(v)=\begin{cases}1/\sqrt{2} & u,v=0\\ 1 & otherwise\end{cases}$$

(3)Quantification, $F^{Q}(u,v)=round(\frac{F(u,v)}{Q(u,v)}),\quad u,v\in\{0...7\}$, $F(u,v)$ is the DCT coefficient of the sub image $f(x,y)$. $Q(u,v)$ is the quantization matrix. The process of quantification is losssy. Again in the form of a zigzag of quantization matrix for scanning reordering, usually matrix quantization is one-to-one correspondence with the image quality factor, quality factor is an integer from 1 to 100, the higher the JPEG image quality factor, the image is clear.

(4)Code, the DC (Direct Current) component is coded by differential pulse modulation coding, and the coding of the AC (Alternating Current) component is encoded by the entropy coder.

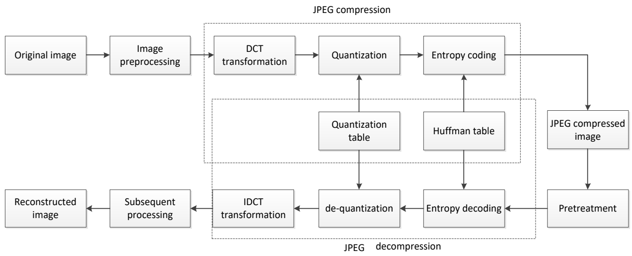


Figure 1. JPEG Compression and Decompression Process

3. The Mathematical Model of JPEG Tampered Image

Tampering image (Figure 2 (d)) refers to a part of the image is changed, it should include the area (background area) is not be tampered and tampering area.

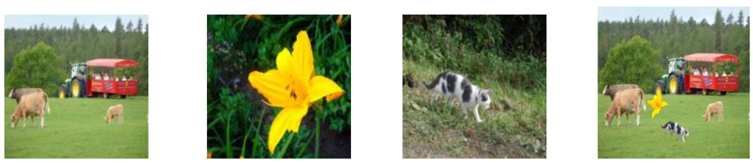
A JPEG format image is partially replaced with other images, and then saved as JPEG format, such as shown in Figure 2, Figure (a) as a JPEG format background image p, (b)(c)for tampering with the source image, Figure (d) for tampering with the synthetic image p, the tampered images Y can described as:

$$Y = A\odot P + A_{T1}\odot P_{T1} + ... + A_{Tm}\odot P_{Tm} + ... + A_{Tn}\odot P_{Tn} \tag{2}$$

where $\odot$ indicate hadamard multiplication. A is the full 1 matrix. $P_{Ti}(i=1,2,...,n)$ is the source image.

$$A_{T1}=\begin{cases}1 & (i,j)\in U_{T1}\\ \vdots \\ 0 & (i,j)\in U_{Ti}\\ \vdots \\ 0 & (i,j)\in U_{Tn}\end{cases},...,A_{Ti}=\begin{cases}0 & (i,j)\in U_{T1}\\ \vdots \\ 1 & (i,j)\in U_{Ti}\\ \vdots \\ 0 & (i,j)\in U_{Tn}\end{cases},...,A_{Tn}=\begin{cases}0 & (i,j)\in U_{T1}\\ \vdots \\ 0 & (i,j)\in U_{Ti}\\ \vdots \\ 1 & (i,j)\in U_{Tn}\end{cases}$$

Among them $U_{T1},U_{T2},\cdots U_{Tn}$ is the activation interval. $U_p = U-[A_{T1}+A_{T2}+\cdots+A_{Tn}]$, $U_P\cap U_{T1}\cap U_{T2}\cap\cdots\cap U_{Ti}\cap\cdots\cap U_{Tn}=\varnothing$, $U_P\cup U_{T1}\cup U_{T2}\cup\cdots\cup U_{Ti}\cup\cdots\cup U_{Tn}=U$. The purpose of this paper is to exactly separate the tampered regions from the tampered image Y.



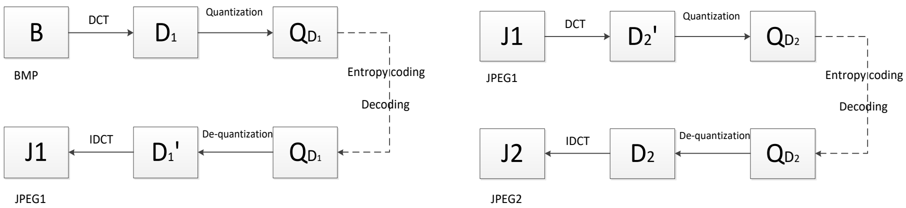
(a) (b) (c) (d)

((a) Original image P_1;(b),(c) are the tampered source image P_2;(d) is the tampered image Y)

Figure 2. Synthesis Tampered Image of JPEG

4. The Double Quantization Effect in JPEG Compression

When the image undergo the first time JPEG compression, we need a quantization matrix Q^{T_1} ,and do the quantization to the DCT coefficients. Then, we can obtain the first quantized DCT coefficients. And for the second times JPEG compression, the first quantized DCT coefficients are multiplied by the first quantization matrix Q^{T_1}, then use the second quantization matrix Q^{T_2} for quantization operation. Figure 3 shows the process of JPEG image compression.

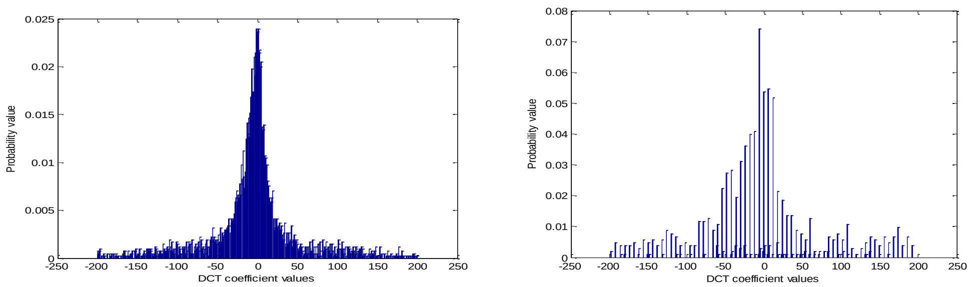


(a) (b)

((a) single JPEG compression；(b) double JPEG compression)

Figure 3. Single JPEG Compression and Double JPEG Compression

In order to observe the histogram feature, with a resolution 256×256 of the Lena grayscale image as an example to illustrate. The image is first divided into 8×8 DCT blocks. We extract all 8×8 image blocks DCT coefficients at the (1, 2) position. Figure 4 (a), shows the DCT coefficients of the probability distribution histogram without the quantization operation at (1, 2) position. Figure 4 (b), shows the DCT coefficients of the probability distribution histogram after quantization step $Q^1 = 5$ at (1, 2) position. From Figure 4 (a), we can see, when the image through a discrete cosine transform, the Alternating Coefficient (AC) of DCT coefficients in the same position with the 8×8 image blocks showing the approximate Laplace distribution. When the DCT coefficients undergo a quantization operation. It's coefficient will be showing the periodic artifacts in the probability distribution histogram, as Figure 4 (b), shows.



(a) (b)

(a) the DCT coefficients of the probability distribution histogram without the quantization operation

(b) the DCT coefficients of the probability distribution histogram after once quantization

Figure 4. The DCT Coefficients of the Probability Distribution Histogram of JPEG Images

We assume that the first compression process the unquantized DCT coefficients value is u_1, the quantized coefficients value is $u_1^{'}$ .The first quantization step is Q^1, and the second quantization step is Q^2. The second quantized DCT coefficients is u_2. According to the JPEG compression and decompression process we can get:

$$u_2 = \left[\left[{u_1}/{Q^1} \right] {Q^1}/{Q^2} \right] \quad (3)$$

According to the nature of rounding：

$$Q^1(\left\lceil \frac{Q^2}{Q^1}(u_2 - \frac{1}{2}) \right\rceil - \frac{1}{2}) \le u_1 < Q^1(\left\lfloor \frac{Q^2}{Q^1}(u_2 + \frac{1}{2}) \right\rfloor + \frac{1}{2}) \quad (4)$$

Where $\lfloor \bullet \rfloor$ denote the floor function, $\lfloor \bullet \rfloor$ is the ceiling function.

We can set up

$$L(u_2) = Q^1(\left\lceil \frac{Q^2}{Q^1}(u_2 - \frac{1}{2}) \right\rceil - \frac{1}{2})$$

$$R(u_2) = Q^1(\left\lfloor \frac{Q^2}{Q^1}(u_2 + \frac{1}{2}) \right\rfloor + \frac{1}{2}) \quad (5)$$

So we can use the interval length of the DCT coefficient histogram to describe the variation relationship of the DCT coefficient between the first and the second JPEG compression, it can be expressed as Formula (6):

$$\begin{aligned} n(u_2) &= R(u_2) - L(u_2) \\ &= Q^1\left(\left\lfloor \frac{Q^2}{Q^1}(u_2 + \frac{1}{2}) \right\rfloor - \left\lceil \frac{Q^2}{Q^1}(u_2 - \frac{1}{2}) \right\rceil + 1 \right) \end{aligned} \quad (6)$$

According to Equation (6) we can see that, $n(u_2)$ is the length of the interval u_1 .Note that $n(u_2)$ is a periodic function, and it's cyclical nature contributing to the DCT coefficients of the probability distribution histogram. And it's period is $P = \frac{Q^1}{\gcd(Q^1, Q^2)}$, where $\gcd(Q^1, Q^2)$ is the greatest common divisor of Q^1 and Q^2 .

5. The Detection Algorithm Based on the Double Quantization Effect on DCT Coefficients

5.1. Double Quantization in Tampered JPEG Images

When an original image has been tampered and then saved as JPEG format. The DCT coefficients of the areas without tampering will be compressed for twice while the tampered areas only suffered once. Below we shall discuss by three kinds of situation to discuss what happened on the situation in different regions of compression of the JPEG synthetic tempering:

(a) The original image of the tamper with the regional image is not a JPEG format (for example, an BMP, GIF format image or other lossless formats), The original image itself is not experienced JPEG compression, when the final image is saved as tampering JPEG format. When the final tampered image is saved as JPEG format, the tampered regions experienced only one time JPEG compression, naturally not exhibit the double compression effect.

(b) The tampered region with the background region have a low probability match of the DCT grid. In the actual operation of the image tampering, the tamper often focus on a specific area of the image. When the tampered region comes from a JPEG image, we assume that the start position coordinates of the background area (x_1, y_1) , the tampered area starting position (x_2, y_2) . Then the probability of $(|x_2 - x_1|\%8, |y_2 - y_1|\%8) = (0,0)$ is

only $1/64$.
(c) In other words, the probability of the tampered regions showed double compression effect is very small.

In order to make the tampered image looks more realistic, the tamper often carry out fuzzy retouching, feathering, smoothing to the edge of the tampered region. At this point these blocks will not include the complete 8×8 image blocks. Therefore, the tampered area can be considered experienced only one time JPEG compression.

5.2. Extraction the Eigenvalues of the Tampered Block Based on Bayesian

From the analysis in Section 5.1 we can know, the tampered regions experienced only one time JPEG compression, does not have the double quantization effect; the background region experienced two times JPEG compression, with the double quantization effect. If we can determine the probability distribution model of each image pixel when it undergoes the first JPEG compression and the second JPEG compression. We can use Bayesian methods to estimate the probability to be detected each pixel in the image has been tampered.[12] pointed out that the image after DCT transform, the AC coefficient distribution accord with a Laplace probability density distribution described with parameter λ .Thus, for a non-quantized DCT coefficient block, it's AC component coefficient (u_1) distribution accord with a Laplace probability density distribution described with parameter λ :

$$p(u_1)=\frac{\lambda}{2}\exp(-\lambda|u_1|) \tag{7}$$

Where $\lambda=\sqrt{2}/\sigma$, σ is the standard deviation of the image. By the Formula (6) we can see, the area has not been tampered experience twice JPEG compression , the original DCT coefficient $[L(u_2),R(u_2)]$ of u_1 it will be mapped to the same value u_2 .And therefore, the probability of the image has not been tampered block DCT coefficient' value u_2 can be described by the Formula (8):

$$\begin{aligned} p(u_2|H_1) &= \int_{L(u_2)}^{R(u_2)} p(u_1)du_1 \\ &= F(R(u_2))-F(L(u_2)) \end{aligned} \tag{8}$$

Where H_1 represents the normal pixel of the image, it's DCT coefficient distribution meet the double quantization mapping relationship. According to the characteristics of the absolute value function, if the function of a Laplace distribution is divided into two symmetrical situation. Then, it is easy to do integral calculation for the function:

$$\begin{aligned} & p(u_2|H_1)=F(R(u_2))-F(L(u_2)) \\ & =\frac{1}{2}\big[1+\text{sgn}(R(u_2)).(1-\exp(-\lambda|R(u_2)|))\big]-\frac{1}{2}\big[1+\text{sgn}(L(u_2)).(1-\exp(-\lambda|L(u_2)|))\big] \end{aligned} \tag{9}$$

Among them, $F(x)$ is the cumulative distribution function, so, $F(x)=\int_{-\infty}^{x} f(u)du=\frac{1}{2}\big[1+\text{sgn}(x)\cdot(1-\exp(-\lambda|x|))\big]$.

Through the analysis of Section 4 we can see, the tampered areas once undergo the quantization operation can be viewed as a quantization operation twice with the same quantization step $Q^2=Q^1$. Therefore, the probability of the DCT coefficients u_2 of the tampered block can be represented by the Formula (10) is expressed as:

$$p(D_2|H_2)=F(R'(D_2)-F(L'(D_2)) \tag{10}$$

Among them, $R'(u_2) = Q^2(\left\lfloor u_2 + \frac{1}{2} \right\rfloor + \frac{1}{2})$, $L'(u_2) = Q^2(\left\lceil u_2 - \frac{1}{2} \right\rceil - \frac{1}{2})$ Where H_2 represents the tampered pixel of the image. Its DCT coefficient distribution does not meet the double quantization mapping relationship (or we can be considered to meet the specific circumstances : $Q^2 = Q^1$).

According to Bayes' theorem, the probability of the image pixels as normal pixels can be described as:

$$p(H_1|u_2) = \frac{p(u_2|H_1) \times p(H_1)}{p(u_2|H_1) \times p(H_1) + p(u_2|H_2) \times p(H_2)} \quad (11)$$

Where $p(H_1)$ and $p(H_2)$ is the priori probability of the normal pixel and the tampered pixel of the image. In our experiments, for convenience, is taken to be equal probability distribution, $p(H_1) = p(H_2) = \frac{1}{2}$.Therefore,

$$p(H_1|u_2) = \frac{p(u_2|H_1)}{p(u_2|H_1) + p(u_2|H_2)} \quad (12)$$

Equation (8) is the posterior probability of a single pixel of a normal pixel. Because JPEG compression is based on 8×8 pixel block operations, with 64 frequency values(a DC coefficient and 63 AC coefficients). In our experiments, we put 63 AC component of the posterior probability of each 8×8 pixel block of value added, then we can obtained the posterior probability values T of each 8×8 pixel block, to obtain a block after posterior probability density map, which is 1/64 the size of the original image.

$$T = \sum_{j=1}^{N} p_j \quad (13)$$

Where $N = 63$, p_i is the value of 63 AC component of the posterior probability of each 8×8 pixel block.

T is the eigenvalues of each 8×8 image block. We believe that by the above algorithm, if the block is to be detected tampering block, its DCT coefficient distribution meet the double quantization mapping relationship and the posterior probability value is relatively large, and are gathered in a central region, performance in the posterior probability map is a white area. While the normal block has no such phenomenon, showing nearly black. So that we can locate the tampered area from the posterior probability map.

In the experiment, by using Matlab JPEG Toolbox [13] can be directly obtained the second quantization matrix of the tampered images, for the first time quantization matrix there are many documents are given corresponding solving method, in this experiment we use the improved algorithm in [14] to perform strike, will not repeat them here.

5.3. Adaptive Multi-Threshold Set by Particle Swarm Optimization (PSO) Algorithm Automatically Extracts the Tampered Regions

By the above algorithm we get the posterior probability map of the tampered JPEG image. But how to separate the tampered regions from the tampered image without respect to the location, size and number of tampered images is still a very difficult problem. In the experiment, we assign an adaptive multi- threshold for the eigenvalues with the Particle Swarm Optimization Algorithm to separate the tampered region automatically. Here are the basic steps of the algorithm:

(1) Initialization: Set population size N, the dimension of each particle is D, Each particle value Represent the pixel values within the interval [0,255]. Then, the initial position and velocity of each particle groups were set by random.

(2) Select the Formula (14) $F(t)$ as a adaptation function of the Particle Swarm Optimization (PSO) Algorithm, and calculate the fitness value of each particle in the particle swarm. Then according to the fitness value, select the current best position P_i and the global best position P_g of each particle.

$$F(t)=1+2[p_0(t)\ln\sigma_0(t)+p_1(t)\ln\sigma_1(t)]-2[p_0(t)\ln p_0(t)+p_1(t)\ln p_1(t)] \quad (14)$$

Where $p_0(t)=\sum_{i=0}^{t}h(i)$, $p_1(t)=\sum_{i=i+1}^{L-1}h(i)$, $\mu_0(t)=\frac{\sum_{i=0}^{t}h(i)*i}{p_0(t)}$ $\mu_1(t)=\frac{\sum_{i=i+1}^{L-1}h(i)*i}{p_1(t)}$, $\sigma^2{}_0=\frac{\sum_{i=0}^{t}[i-\mu_0(t)]^2h(i)}{p_0(t)}$, $\sigma^2{}_1=\frac{\sum_{i=t+1}^{L-1}[i-\mu_1(t)]^2h(i)}{p_1(t)}$

the best threshold is $T_{best}=Arg\min_{t\in(0,1,2,\ldots,L-1)}F(t)$.

(3) Continue to the next step of the iteration. According to the Formula (1), evolution of each particle V_i and X_i, calculate their adaptation value. For each particle, compare the adaptive value with the current best position P_i, if the former is better, we take it as the best current position. Then, for each particle, compare the adaptive value with the global best position P_g, if better, we take it as the best global position.

$$V_i=\omega V_i+c_1*rand()*(P_i-X_i)+c_2*rand()*(P_g-X_i) \quad (15)$$

$$X_i=X_i+V_i \quad (16)$$

In Formula (15), where $rand()$ expressed as a uniformly distributed random numbers in (0,1), c_1, c_2 representing the best position to fly its own direction and global best position learning factor, in our experiments $c_1=c_2=2$. ω represent the inertia weight. Usually beginning with 1 and decreased with the increasing number of iterations to meet the search needs.

(4) If P_g constitute the global optimal solution F,is less than the minimum permissible error or the number of iterations exceeds the preset number of termination, then the algorithm ends, the answer is consistent with the final result. Otherwise, go to Step (3) to continue the iteration.

(5) We obtained QF_2, corresponds to the value of the QF_2 as the segmentation threshold automatically extracts the tampered regions.

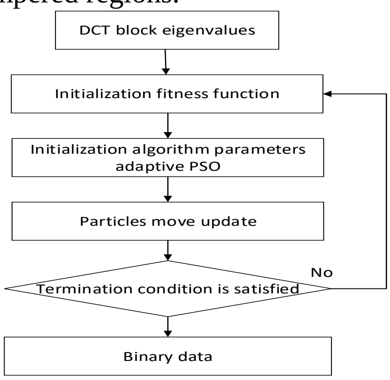


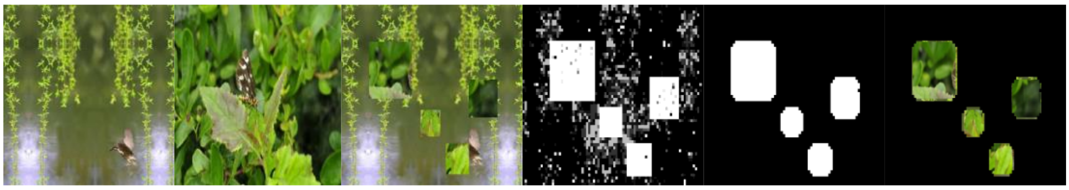
Figure 5. Optimization Algorithm Flowchart

6. Experimental Results and Analysis

In order to verify the accuracy of the proposed algorithm, in our experiment, we select 100 images form the lossless formats images (eg:TIFF, BMP formats and the resolution of the image is 1024×1024) from different cameras in our real-life. Experimental platform using Matlab R2014a, and do the tampering operation with Photoshop CS6.0. Starting from this, we refer to these images into JPEG format and saved with quality factor of QF_1 (use the imwrite function in the Matlab to complete). Then we use the mathematical model of JPEG tampered image to do the tampering operation, Finally, we resaved the tampered JPEG images with a quality factor QF_2 , in the experiment, $QF_1 = \{60,65,70,75,80,85,90,95\}$ $QF_2 = \{60,65,70,75,80,85,90,95\}$. Therefore, for each image we can get 30 groups $< QF_1, QF_2 >$ values, we do not consider the case of $QF_1 = QF_2$ .Because according to Section 2.4 analysis we can know that this situation does not meet the double quantization effects. After our algorithm, for each size of 1024×1024 tampering map, we calculate the posterior probability of each DCT block has been tampered, and finally we get a 128×128 posterior probability map, we can successfully locate the tampered areas.

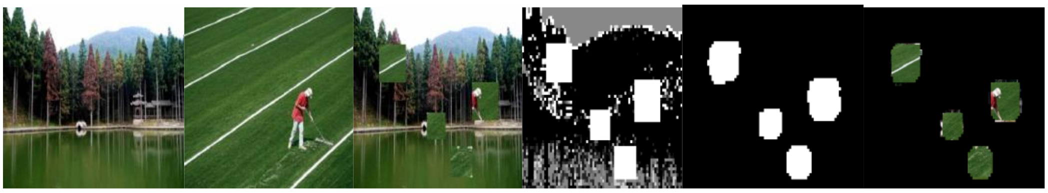
We conducted two types of comparative experiments to verify the accuracy and robustness of the algorithm in this paper.

(1) The second quality factor is bigger than the first quality factor (the situation of $QF_1 < QF_2$), and the location, size and number of the tampered regions of the tampered images are unknown. And the Figure(6), shows the blind separation results, Figure (a), is the background image ($QF_1 = 65$), Figure (b) is the tampered source image, Figure (c), is the synthetic tamper image($QF_2 = 85$),Figure (d), is the posterior probability density maps of the tampered image, Figure (e), is the binary image through an adaptive multi-threshold for the eigenvalues with the Particle Swarm Optimization (PSO) Algorithm, and Figure (f), shows the tampered regions by our algorithm. It can be seen in the case of $QF_1 < QF_2$, we almost fully detect the tampered regions, and there is almost no error checking.



(a) (b) (c) (d) (e) (f)

Figure 6. The Test Results of in the Case of $QF_1 < QF_2$



(a) (b) (c) (d) (e) (f)

Figure 7. The Test Results of in the Case of $QF_1 > QF_2$

(2) The second quality factor is smaller than the first quality factor (the situation of $QF_1 > QF_2$), and the location, size and number of the tampered regions of the tampered images are unknown. And the Figure (7), shows the blind separation results, Figure (a), is the background image ($QF_1 = 85$), Figure (b) is the tampered source image, Figure (c), is the synthetic tamper image ($QF_2 = 70$), Figure (d), is the posterior probability density maps of the tampered image, Figure (e), is the binary image through an adaptive multi-threshold for the eigenvalues with the Particle Swarm Optimization (PSO) Algorithm. The white areas correspond to high probability of being tampered, and Figure (f) shows the tampered regions by our algorithm. In the case of $QF_1 > QF_2$, as can be seen from the posterior probability density map, it showed a greater deviation error that we get the posterior probability value with the actual result, but after our optimization algorithm, we still can automatically detect the fully tampered regions from the tampered image without respect to the location, size and number of tampered images.

To test the accuracy of this algorithm performance, we have established an objective evaluating criterions. The first evaluating criterion is correct separation rate(CSR),as shown in Equation (17), which is ratio of number of DCT blocks (8×8 in our test) tampered which has been detected and total number DCT blocks which has been tampered. The second evaluating criterion is false separation rate (FSR) which is ratio of number of DCT blocks that false detected and total area of the image has not been tampered block number, as defined in (18).

$$CSR = \frac{n_T - n_{TNT}}{n_T} \tag{17}$$

$$FSR = \frac{n_{NTT}}{n_I - n_T} \tag{18}$$

The parameters in the formula: n_{NTT} :the number of DCT blocks have been not tampered, but detected as tampered. n_{TNT} : the number of DCT blocks tampered, but not detected as tampered. n_I : the number of DCT blocks in the image. n_I : the number of DCT blocks have been tampered. The values for CRS and FSR achieved by this article are shown in Tables 1, and 2. The values for CRS and FSR achieved by the algorithm in [14] are shown in Tables 3, and 4 .The best results for each combination are highlighted in bold

Table 1. CSR Achieved by this Article

QF_1	QF_2							
	60	65	70	75	80	85	90	95
60		**0.86**	**1.00**	**0.99**	0.99	1.00	1.00	1.00
65	**0.84**		**0.87**	0.94	0.99	0.99	0.99	1.00
70	**0.85**	**0.88**		**0.90**	0.90	0.98	1.00	0.99
75	**0.81**	**0.80**	0.80		0.91	0.98	0.99	0.99
80	0.81	0.80	0.82	0.86		0.94	0.98	0.98
85	**0.90**	**0.95**	**0.86**	0.94	**0.91**		0.96	1.00
90	0.86	0.93	0.87	0.95	0.86	0.89		0.99
95	0.86	0.95	0.85	0.91	0.80	0.85	0.90	

Table 2. FSR Achieved by this Article

QF_1	QF_2							
	60	65	70	75	80	85	90	95
60		**0.01**	**0.00**	**0.00**	**0.00**	0.02	0.00	0.00
65	**0.13**		**0.00**	**0.02**	**0.00**	**0.02**	0.00	0.01
70	0.09	**0.09**		**0.10**	0.09	0.00	0.00	0.00
75	0.10	0.10	0.20		0.16	**0.02**	0.01	0.00
80	0.06	**0.08**	**0.01**	**0.02**		**0.05**	**0.02**	0.00
85	**0.06**	**0.01**	**0.11**	**0.09**	**0.09**		**0.01**	**0.01**
90	0.08	0.12	0.12	**0.08**	0.13	0.10		0.00
95	**0.06**	0.20	0.15	0.16	0.10	0.10	0.01	

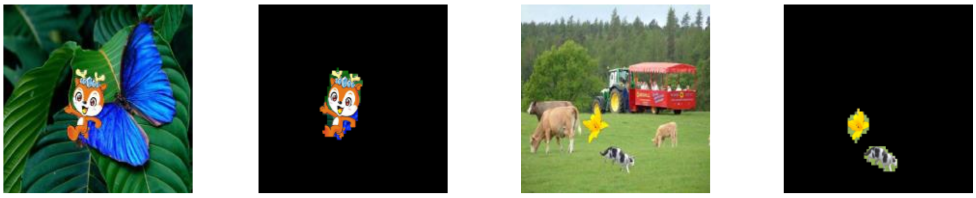
Table 3. CSR Achieved by the by the Algorithm in [14]

QF_1	QF_2							
	60	65	70	75	80	85	90	95
60		0.56	0.76	0.89	0.91	1.00	1.00	1.00
65	0.55		0.78	0.87	0.95	0.99	0.99	1.00
70	0.63	0.62		0.84	0.90	0.99	1.00	1.00
75	0.69	0.69	0.78		0.94	0.98	0.94	0.99
80	0.72	0.81	0.75	0.80		0.93	0.96	0.98
85	0.85	0.80	0.81	0.87	0.76		0.99	0.93
90	0.93	0.89	0.83	0.91	0.93	0.84		0.85
95	0.90	0.91	0.86	0.92	0.94	0.95	0.98	

Table 4. FSR Achieved by the by the Algorithm in [14]

QF_1	QF_2							
	60	65	70	75	80	85	90	95
60		0.21	0.16	0.12	0.10	0.09	0.08	0.00
65	0.23		0.19	0.22	0.18	0.18	0.09	0.01
70	0.12	0.30		0.13	0.19	0.20	0.06	0.06
75	0.10	0.20	0.25		0.26	0.19	0.11	0.10
80	0.16	0.26	0.31	0.26		0.23	0.12	0.06
85	0.31	0.19	0.28	0.23	0.19		0.12	0.11
90	0.10	0.10	0.24	0.18	0.13	0.14		0.08
95	0.20	0.26	0.23	0.16	0.10	0.13	0.13	

To illustrate the robustness of the algorithm, the algorithm has been also tested in real life. We use Photoshop CS6.0 to do the tampered manipulate. To make the tampered images look more realistic, we carry out fuzzy retouching, feathering, smoothing to the edge of the tampered region. Figure (8), shows the application to realistic forgeries. Figure (a), (c), are the tampered images. Figure (b), (d), are the blind separation results. Obviously, our algorithm has good blind separation effect to the tampered region and the region has not been tampered. For example, from Figure (a) (c), we can easily separate the forged area (a cartoon animals, a cat and a flower) from the tampered images.



(a) (b) (c) (d)

Figure 8. Application to Realistic Forgeries

7. Concluding Remarks

This paper proposed a new probability model to describe the change of DCT coefficients' statistical properties before and after the double compression in JPEG images. The AC coefficient distribution accord with a Laplace probability density distribution described with parameter λ .The experimental results show that the method can detect and locate the tamped area effectively and it outperforms other algorithms in terms of the detection result especially when the second compression factor is smaller than the first one. Compared with other traditional methods, the proposed approach could effectively separate the tampered regions from the tampered image without respect to the location, size and number of tampered images. But when the second compression factor is equal to the first one ($QF_1 = QF_2$), our Algorithms discussed here in our paper is no longer work. We will continue to study these issues in the future.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (No. U1204606), the Key Programs for Science and Technology Development of Henan Province (No.142102210565), the doctoral scientific research foundation of Henan Normal University.

References

[1] W. Lie, T. Lin and S. Cheng, "Dual protection of JPEG images based on informed embedding and two-stage watermark extraction techniques", Information Forensics & Security IEEE Transactions on, vol. 11, **(2006)**, pp. 330–341.
[2] Z. Fan and R. L. de Queiroz, "Identification of bitmap compression history: Jpeg detection and quantizer estimation", IEEE Transactions on Image Processing, vol. 12, **(2003)**, pp. 230–235.
[3] Y. L. Chen and C. T. Hsu, "Detecting doubly compressed images based on quantization noise model and image restoration", in IEEE Workshop on Multimedia Signal Processing, **(2009)**.
[4] Q. Liu, P. A. Cooper and L. Chen, "Detection of JPEG double compression and identification of smartphone image source and post-capture manipulation", Applied Intelligence, vol. 39, **(2013)**, pp. 705-726.
[5] Z. Lin, J. He and X. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis", Pattern Recognition, vol. 21, **(2009)**, pp. 2492-2501.
[6] J. Lukas and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images", Proceedings of Digital Forensic Research Workshop, Orlando, Florida, USA, **(2003)**, pp. 5-8.
[7] G. S. Lin, M. K. Chang and Y. L. Chen, "A passive-blind forgery detection scheme based on content-adaptive quantization table estimation", IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, **(2011)**, pp. 421-434.
[8] F. L. Shi, B. S. Kang and H. A. Li, "A new method for detecting JEPG doubly compression images by using estimated primary quantization step", 2012 IEEE International Conference on Systems and Informatics (ICSAI), Yantai, China, **(2012)**, pp. 1810-1814.
[9] J. Fridrich and T. Pevny, "Detection of double–compression for applications in steganography", IEEE Transactions on Information Security and Forensics, vol. 3, **(2008)**, pp. 247–258.
[10] G. Wallace, "The JPEG still picture compression standard", IEEE Transactions on Consumer Electronics, vol. 34, **(1991)**, pp. 30–44.
[11] "Digital compression and coding of continuous-tone still images", Part 1: Requirements and guidelines, ISO/IEC JTC1 Draft International Standard 10918-1, **(1991)**.
[12] R. Reininger and R. J. Gibson, "Distributions of the two-dimensional DCT coefficients for images", Communications IEEE Transactions on, vol. 31, **(1983)**, pp. 835-839.
[13] DDE Lab at Binghamton University. Matlab JPEG toolbox [2015-07-08].http://dde.binghanmton.edu/download/feature_extractors/.
[14] T. Bianchi, A. De Rosa and A. Piva, "Improved DCT coefficient analysis for forgery localization in JPEG images", Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on, **(2011)**, pp. 2444-2447.

Author

Duan Xintao, was born in Xinxiang China on April 1972. He received the M.Sc. degree in Computer application technology from Shanghai Normal university in 2004, and D.Sc. degrees in communication and information system from Shanghai university in 2011. He is now Associate Professor at Henan normal university, China. His main research interests include sparse approximation, blind source separation and image processing.

