Soft Comput (2009) 13:393–400
DOI 10.1007/s00500-008-0327-7

FOCUS

PM1 steganography in JPEG images using genetic algorithm

Lifang Yu · Yao Zhao · Rongrong Ni · Zhenfeng Zhu

Published online: 3 June 2008

Abstract Plus minus 1 (PM1) is an improved method to least significant bits (LSB)-based steganography techniques, which not only foils typical attacks against LSB-based techniques, but also provides high capacity. But how to apply it to JPEG images does not appear in literatures. In this paper, PM1 steganography in JPEG images using genetic algorithm (GA) is proposed, in which the GA is used to optimize the performance, such as minimizing blockiness. Theoretical analysis to the histogram characteristics after steganography is discussed in details, which proves that PM1 used in JPEG images preserves the first-order statistical properties. Experiments show that the proposed method outperforms the other methods in terms of capacity and security.

1 Introduction

Along with watermarking, steganography is one of the two major branches of information hiding (Petitcolas et al. 1999; Pan et al. 2007), which is the art and science of undetectable communication. The secret message is hidden in a cover medium, without arousing suspicion that would be caused by sending an encrypted message. The resulting medium is called stego medium. In general, there are three requirements on a steganography algorithm: capacity, security and robustness. The former two attract more attention of researchers. However, capacity and security conflict with each other, that is, improvement on capacity will impair security, and vice versa.

JPEG images are extensively used in email transmission and the Internet. Many researchers have attempted to provide different solutions for steganography in JPEG images to balance the two conflicting requirements: capacity and security.

JSteg (D. Upham, JPEG-Jsteg-v4: http://www.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz), a well-known information hiding-tool, replaces the least significant bits (LSBs) of the quantized DCT coefficients (excluding DC members and AC members valued 0/1) with the secret message bits. It has high capacity, almost the same as the number of quantized non-zero AC coefficients. But it can be easily detected by χ^2 (chi-square) attack (Westfeld and Pfitzmann 2000) that uses the closed gap between pairs of values ($0\leftrightarrow1$, $2\leftrightarrow3$, $4\leftrightarrow5$, ..., $254\leftrightarrow255$) after information hiding.

On one hand, some researches focus on improving security while obtaining high capacity by searching for different mechanisms to modify the quantized DCT coefficients. The improved methods based on JSteg randomly scatter embedding, which have the same capacity as JSteg and survive χ^2 attack. But they can be detected by extended χ^2 attack (Provos 2001). F5 (Westfeld 2001) changes the quantized non-zero AC coefficients by decreasing their absolute value (decreasing when positive, and increasing when negative) rather than flipping its LSBs (decreasing when odd, and increasing when even). It can foil χ^2 attack and extended χ^2 attack, but its capacity is lower than JSteg because of matrix coding. Moreover, it has been shown that F5 still changes the histogram of the coefficients in a detectable way. By estimating the original histogram of the coefficients from the cropped and recompressed version of the stego image, differences of histogram between the estimated coefficients and the stego coefficients become evident (Fridrich et al. 2002a). Outguess (Provos 2001) also preserves the first-order statistical properties, which reserves about half of the coefficients to correct statistical deviations caused by flipping LSBs in the

L. Yu (✉) · Y. Zhao · R. Ni · Z. Zhu
Institute of Information Science, Beijing Jiaotong University, Beijing, China
e-mail: yzhao@bjtu.edu.cn

other half. It reduces capacity almost by half because of statistical compensation. Model-based steganography (Sallee 2003, 2005) aims at preserving the characteristics of histogram by making the histogram meet a given distribution (e.g., generalized cauchy distribution). It is detectable because the stego images' quantized DCT coefficients values match the distribution much better than cover images. It can also be detected by the difference of blockiness between a stego image and its estimated image reliably (Fridrich 2004).

On the other hand, modification to the quantization table was made to improve capacity. For example, JQTM (Chang et al. 2002) decreases quantization steps at middle frequency and results in an increase of the number of non-zero coefficients. It has higher capacity because of the additional non-zero AC coefficients. However, steganography system using this method can be easily detected by checking its quantization table even with very little message payload, so it is not secure at all (Fridrich et al. 2003).

The genetic algorithm (GA) has been used plentifully in information hiding discipline these years (Chu et al. 2008; Huang et al. 2007; Pan et al. 2004) and has been shown to be an effective technique for improving the performance of information hiding systems. For example, the GA is helpful to select proper zerotrees in a wavelet transform for a watermark system (Chu et al. 2008), select proper frequency coefficients to carry watermark (Huang et al. 2007), and overcome the VQ index assignment problem in VQ-based watermark systems to make them suitable for transmitting the watermarked image over noisy channels (Pan et al. 2004).

Our proposed method embeds secret messages in JPEG images based on plus minus 1 (PM1) and the GA, which can use all non-zero AC coefficients to get high capacity while preserve typical statistical characteristics, including characteristics of histogram and blockiness. In fact, PM1 preserves histogram characteristics, while the GA finds a proper solution for PM1 to decide whether to plus or minus one at each position that need a modification in the perspective of minimizing blockiness.

The rest of this paper is organized as follows. Section 2 introduces the general principle of the GA. Then, in Sect. 3, we present the proposed steganography method, including PM1 used in JPEG images, the embedding and extraction procedure, the GA's particular use in our method, and proofs for PM1 used in JPEG images preserving the characteristics of histogram. The experimental results are conducted in Sect. 4, and Sect. 5 gives some conclusions.

2 General principle of the GA

Genetic algorithm is a technique for optimization and search, which is based on the Darwinian principles of survival and reproduction (Goldberg 1989).

The GA processes populations of chromosomes (individuals), which replace one population with another successively. The chromosome in the GA is often held in binary encoding. Each chromosome represents a candidate solution in the searching space. The GA usually needs a fitness function to assign a score (fitness) to each chromosome in current population.

The GA starts with initializing a population of individuals by guess. The individuals evolve through iterations, called generations. In each generation, each individual is evaluated against the fitness function. Genetic operators are used for individuals in the population to generate a next generation of individuals. The process is continued until some form of criterion is met (e.g., a given fitness is met).

The simplest form of the GA uses three types of operators to control chromosomes' reproduction, which are stated as follows:

Selection. Select chromosomes in the population for reproduction. The fitter the chromosome is, the more likely it is selected. That is, fitter chromosomes have greater than average chance of promoting the information they contain within the next generation (Coley 1999).

Crossover. Choose pairs of chromosomes promoted by the selection operator. For each pair, randomly choose a single point and exchange the sequences before and after the point between the two chromosomes to create two off-springs, as showed in Fig. 1.

Mutation. Randomly change (flip) the value of single bits within a chromosome. It can be implemented in a way that randomly select one chromosome from the population and then arbitrarily change some of its bits, as showed in Fig. 2.

There are several schemes for the selection process: roulette wheel selection, scaling techniques, tournament, elitist models, and ranking methods (Michalewicz 1994). Because ranking methods allow for minimization and negativity, it is adopted in our method. Typically, crossover is given a rate ranges from 0.6 to 1.0 and a small mutation rate less than 0.1 is usually used (Goldberg 1989).

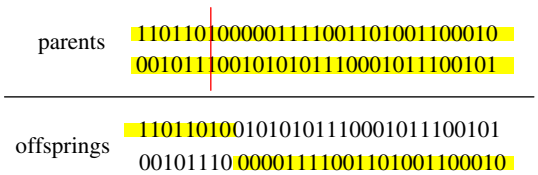

Fig. 1 Crossover

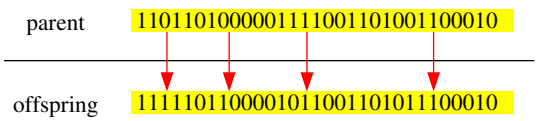

Fig. 2 Mutation

3 The proposed method

3.1 PM1 method and its use in JPEG images

Plus minus 1 (PM1) embedding is an improved method to LSB-based steganography techniques, which is easy to implement but difficult to detect (Soukal 2006). In fact, if the LSB of a given coefficient does not match the message bit to be embedded, LSB-based steganography techniques add one to the even coefficients or subtract one from the odd coefficients, while PM1 randomly increases or decreases by one to change the original value.

Although PM1 has been pointed out to be a possible secure way to implement a high capacity steganography (Fridrich et al. 2003), its application in JPEG has not been mentioned yet. Here we present concretely how to use PM1 properly in JPEG images to get high capacity while preserving high security.

Quantized DCT coefficients consist of three parts, stated as DC coefficients, zero AC coefficients and non-zero AC coefficients. First, DC coefficients represent the mean luminance within a block, so changes to them are more likely to result in perceptual artificial blockiness. Second, zero AC coefficients occur at middle and high frequency continuously, so modifications to them break the structure of continuous zeros and abrupt non-zero values give a hint of the existence of secret bits. Last but the most important, non-zero AC coefficients occur at low and middle frequency, and perturbations to them do not affect the visual quality as much as DC members. So non-zero AC coefficients are proper choices for carrying secret bits. Considering preservation of characteristics of histogram, PM1 should be used in JPEG in this way:

A negative even coefficient represents a steganographic one, a negative odd coefficient means a zero; while a positive even coefficient represents a steganographic zero, and a positive odd coefficient means a one.

During embedding, if the secret bit is the same as what its corresponding non-zero AC coefficient represents, the coefficient is unchanged; otherwise the coefficient is increased or decreased by one randomly. In case the coefficient is changed to zero, modify it to $+1$ or -1 according to the secret bit. In details, if the coefficient is 1 and random process says decreasing, it is changed to -1, and vice versa.

In brief, PM1 used in JPEG images is denoted as "J-PM1" in the rest of this paper.

3.2 The embedding procedure

The embedding procedure starts with decoding the cover JPEG image to quantized DCT coefficients, which are later shuffled by the key-based permutation. Secret message bits are compared with quantized permutated non-zero AC coefficients sequentially to decide the number of coefficients that need modifying. Then the GA was used to find the best plus/minus solution for modification and corresponding coefficients are modified. After the coefficients are inversely permutated, they are encoded in the Huffman encoder to achieve stego JPEG images. The use of the GA in J-PM1 increases the security of our proposed method.

Figure 3 shows the embedding procedure of the proposed method. The whole embedding procedure can be divided into six steps.

Step 1: Pre-processing. Apply an entropy decoder to decode the cover JPEG image to attain quantized DCT coefficients X. Then shuffle all coefficients using a key based permutation (straddling mechanism (Westfeld 2001)) to obtain quantized permutated DCT coefficients X_p. Due to permutation the secret bits are scattered all over the cover medium, and the embedding density can be same everywhere (Westfeld 2001).

Step 2: Combination. The length of the secret message L(represented using 16 bits) and the secret message itself are combined together to form a combined message.

Step 3: Optimization. Compare the combined message bit by bit with their corresponding non-zero AC coefficients to determine the number of coefficients that need modifying, which is the length of chromosomes, L_c, used later in the GA. Then, the GA

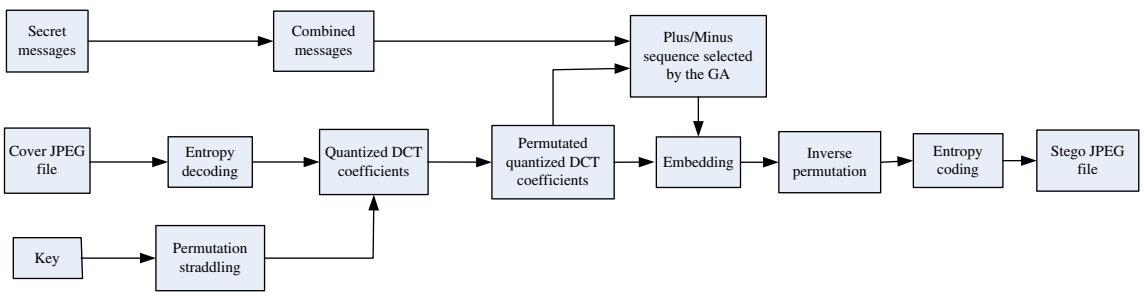

Fig. 3 The diagram of embedding procedure

algorithm is used to select the optimal plus/minus solution for each coefficient to be modified.

Step 4: Modification. Use the plus/minus solution from the third step to modify the corresponding non-zero AC coefficients, getting permutated version of stego quantized DCT coefficients X'_p.

Step 5: Post-processing. The stego quantized DCT coefficients in permutated version (X'_p) are inversely permuted to their original sequence version X', and are delivered to the Huffman encoder to achieve the stego JPEG image.

Step 6: The JPEG image and the key are transferred to the receiver.

3.3 The extraction procedure

The extraction procedure starts with decoding a stego JPEG image to quantized stego DCT coefficients. They are then shuffled by the key based permutation which is the same as that used in the embedding procedure. The length of the secret message is extracted from the first 16 coefficients of the permutated stego non-zero AC coefficients, and then secret message bits are extract successively.

Figure 4 shows the extraction procedure at the receiver side. The extraction procedure consists of two steps described as follows:

Step 1: After receiving the JPEG image and the key, the receiver uses an entropy decoder to recover the quantized stego coefficients X', which are shuffled later based on the received key to attain the permutated DCT coefficients X'_p.

Step 2: From the first 16 coefficients of X', the length of the secret message, L, is extracted. Then, subsequently extracts the secret message bits from the successive L non-zero AC coefficients.

3.4 Searching for the optimal plus/minus solution through the GA algorithm

Blockiness of a stego image and of its estimated image can be used to detect the very existence of secret message bits (Fridrich 2004). The ratio of blockiness between a stego image and its corresponding estimated image (ROB) decides the probability of successful detection. The higher ROB is, the higher the probability of detection is, and vice versa.

Blockiness of an image is defined as follows (Fridrich et al. 2002b):

$$B = \sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} |g_{8i,j} - g_{8i+1,j}| + \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \sum_{i=1}^{M} |g_{i,8j} - g_{i,8j+1}| \quad (1)$$

where $g_{i,j}$ are pixels' values in a $M \times N$ grayscale image and $\lfloor x \rfloor$ denotes the integer part of x. Then ROB is defined as follows (Zhang and Wang 2005):

$$ROB = \frac{\sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} |S_{8i,j} - S_{8i+1,j}| + \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \sum_{i=1}^{M} |S_{i,8j} - S_{i,8j+1}|}{\sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} |E_{8i,j} - E_{8i+1,j}| + \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \sum_{i=1}^{M} |E_{i,8j} - E_{i,8j+1}|} \quad (2)$$

where S means the stego images, and E means the corresponding estimated images.

In the rest of this section, we show how to adopt the GA algorithm to search for the optimal plus/minus solution for PM1 used in JPEG images. In this study, a chromosome of L_c-dimension is described by a plus/minus solution, which is defined as follows:

$$P = p_1 p_2 \cdots p_{L_c}, (p_i \in \{0, 1\}, 1 \leq i \leq L_c),$$

when p_i is 0, it means to change the coefficient by decreasing it by one, and when 1, increasing it by one.

The reciprocal of ROB is regarded as the fitness function, and the reciprocal of ROB value of current stego image is the chromosome's fitness. So the fitness can be stated as follows:

$$fitness = \frac{\sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} |E_{8i,j} - E_{8i+1,j}| + \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \sum_{i=1}^{M} |E_{i,8j} - E_{i,8j+1}|}{\sum_{i=1}^{\lfloor (M-1)/8 \rfloor} \sum_{j=1}^{N} |S_{8i,j} - S_{8i+1,j}| + \sum_{j=1}^{\lfloor (N-1)/8 \rfloor} \sum_{i=1}^{M} |S_{i,8j} - S_{i,8j+1}|}$$

the GA's goal is to find the chromosome in a certain generation with the best (maximum) fitness, which is the best plus/minus solution for PM1 used in current JPEG image and makes our steganography system the least detectable. The search for a plus/minus solution through the GA is described as follows:

Step 1: Initialization. Randomly generate k chromosomes P_m. Calculate the fitness of each chromosome.

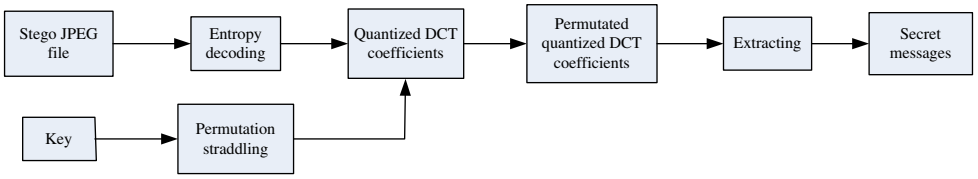

Fig. 4 The diagram of extraction procedure

Step 2: Update the plus/minus solutions (chromosomes) using selection, crossover and mutation as mentioned in Sect. 2.
Step 3: Compute the corresponding fitness of each new chromosome.
Step 4: If the best fitness is smaller than $fitness_{max}$, go to Step 2; else stop and output the best plus/minus solution and its corresponding stego JPEG image.

3.5 Characteristics analysis

The distribution of quantized DCT coefficients of a JPEG image should preserve three characteristics:

1. The coefficient's frequency of occurrence decreases with the increase of absolute values (Westfeld 2001).
2. The decrease of the coefficient's frequency of occurrence decreases with the increase of absolute values, i.e. the difference between two bars in the middle of the histogram is larger than that on the margin of the histogram (Westfeld 2001).
3. The distribution is symmetric around zero (Li et al. 2007).

We can show that J-PM1 preserves the three characteristics. Suppose we have two random variables X, Y, for the observed coefficients before and after J-PM1. $P(X = x)$ denotes the probability when a coefficient in the cover image equals x, and $P(Y = y)$denotes the probability when a coefficient after J-PM1 embedding equals y. We can write the three characteristic properties for some coefficient values:

$$P(X = 1) > P(X = 2) > P(X = 3) > P(X = 4) \tag{3}$$

$$\begin{aligned} P(X = 1) - P(X = 2) &> P(X = 2) - P(X = 3) \\ &> P(X = 3) - P(X = 4) \\ &> P(X = 4) - P(X = 5) \end{aligned} \tag{4}$$

$$P(X = 1) = P(X = -1) \tag{5}$$

$$P(X = 2) = P(X = -2) \tag{6}$$

As mentioned above, PM1 increases or decreases a coefficient by one randomly. If the message bits are uniformly distributed, we deduce,

$$\begin{aligned} P(Y = 1) &= 1/2P(X = 1) \\ &\quad +1/4P(X = -1) + 1/4P(X = 2) \end{aligned} \tag{7}$$

$$\begin{aligned} P(Y = 2) &= 1/2P(X = 2) + 1/4P(X = 1) \\ &\quad +1/4P(X = 3) \end{aligned} \tag{8}$$

$$\begin{aligned} P(Y = 3) &= 1/2P(X = 3) \\ &\quad +1/4P(X = 2) + 1/4P(X = 4) \end{aligned} \tag{9}$$

$$\begin{aligned} P(Y = 4) &= 1/2P(X = 4) \\ &\quad +1/4P(X = 3) + 1/4P(X = 5) \end{aligned} \tag{10}$$

$$\begin{aligned} P(Y = -1) &= 1/2P(X = -1) \\ &\quad +1/4P(X = 1) + 1/4P(X = -2) \end{aligned} \tag{11}$$

We subtract (8) from (7) to get (12), (9) from (8) to get (13), (10) from (9) to get (14).

$$\begin{aligned} &P(Y = 1) - P(Y = 2) \\ &\quad = 1/4P(X = 1) \\ &\qquad +1/4P(X = -1) - 1/4P(X = 2) - 1/4P(X = 3) \\ &\quad = 1/4[P(X = 1) - P(X = 2)] \\ &\qquad +1/4[P(X = 1) - P(X = 3)] \end{aligned} \tag{12}$$

$$\begin{aligned} &P(Y = 2) - P(Y = 3) \\ &\quad = 1/2[P(X = 2) - P(X = 3)] \\ &\qquad +1/4[P(X = 1) - P(X = 2)] \\ &\qquad +1/4[P(X = 3) - P(X = 4)] \end{aligned} \tag{13}$$

$$\begin{aligned} &P(Y = 3) - P(Y = 4) \\ &\quad = 1/2[P(X = 3) - P(X = 4)] \\ &\qquad +1/4[P(X = 2) - P(X = 3)] \\ &\qquad +1/4[P(X = 4) - P(X = 5)] \end{aligned} \tag{14}$$

Based on (3), we know that the right parts of (12), (13) and (14) are positive. Thus, the first characteristic property for comes into existence, i.e.

$$P(Y = 1) > P(Y = 2) > P(Y = 3) > P(Y = 4) \tag{15}$$

We subtract (13) from (12) to get (16),

$$\begin{aligned} &[P(Y = 1) - P(Y = 2)] - [P(Y = 3) - P(Y = 4)] \\ &\quad = 1/4[P(X = 1) - P(X = 2)] \\ &\qquad -1/4[P(X = 2) - P(X = 4)] \end{aligned} \tag{16}$$

Generally, $P(Y = 1) - P(Y = 2) > P(X = 2) - P(X = 4)$, so

$$P(Y = 1) - P(Y = 2) > P(Y = 2) - P(Y = 3) \tag{17}$$

Similarly, based on (4), (13) and (14), we can get,

$$P(Y = 2) - P(Y = 3) > P(Y = 3) - P(Y = 4) \tag{18}$$

Therefore, the second characteristic property for Y also comes into existence,

$$\begin{aligned} P(Y = 1) - P(Y = 2) &> P(Y = 2) - P(Y = 3) \\ &> P(Y = 3) - P(Y = 4) \end{aligned} \tag{19}$$

With (5)–(7) and (11) we know that

$$P(Y = 1) = P(Y = -1) \tag{20}$$

Similarly we can prove these characteristic properties for other values modified by J-PM1, i.e. decreasing occurrence with increasing absolute value (Eq. (15)), decreasing decrease with increasing absolute value (Eq. (19)) and symmetric around zero (Eq. (20)).

4 Experimental results and discussions

In this Section, the experimental results of the proposed method are conducted to show its merits. PM1 used in JPEG images (J-PM1 in brief) and PM1 used in JPEG images with the GA (GA-PM1 in brief) are compared with F5 (Westfeld 2001), Outguess (Provos 2001), Model-based Steganography without (Sallee 2003) and with (Sallee 2005) deblocking, JSteg (http://www.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz.), respectively.

As mentioned in Sect. 1, security and capacity are the two most important criteria in evaluating a steganographic method. Thus, the following experiments focus on these two criteria. Standard 256 gray-level images with size 256×256 are used as covers, such as Lena, Baboon, Couple, Pepper, Woman, Girl, Man, and so on. These images are compressed using an 80 quality factor during JPEG compression for each method.

4.1 Security

Since GA-PM1 is not a LSB flipping method, it makes no sense to consider χ^2 statistical method (Westfeld and Pfitzmann 2000). Moreover, because GA-PM1 does not lead to shrinkage, attacks against F5 (Fridrich et al. 2002a) are not considered. In the following, we will discuss ROB and histogram of each method.

4.1.1 Ratio of blockiness between a stego image and its estimated image (ROB)

From the definition of ROB and Fridrich's theory (Fridrich et al. 2002b), we know that the smaller ROB is, the less detectable the corresponding steganographic technique is. Heuristically, the smaller ROB is, the less blockiness increases, and the less likely the steganographic technique is detected by steganalysis system based on blockiness increment.

We test the ROB of different methods at the following embedding rates expressed in bits per non-zero DCT coefficient (bpc) (Fridrich 2004), i.e., bpc $= 0.1, 0.3, 0.7$.

F5, Outguess, model-based steganography without and with deblocking, JSteg are referred to as F5, OG, MB1, MB2, JSteg respectively. For capacity of OG is mainly less than 0.4 bpc as showed in Sect. 4.2, we compare ROB between all these methods at 0.1 and 0.3 bpc. Figures 5 and 6 show ROB of different steganographic techniques at 0.1 and 0.3 bpc, respectively.

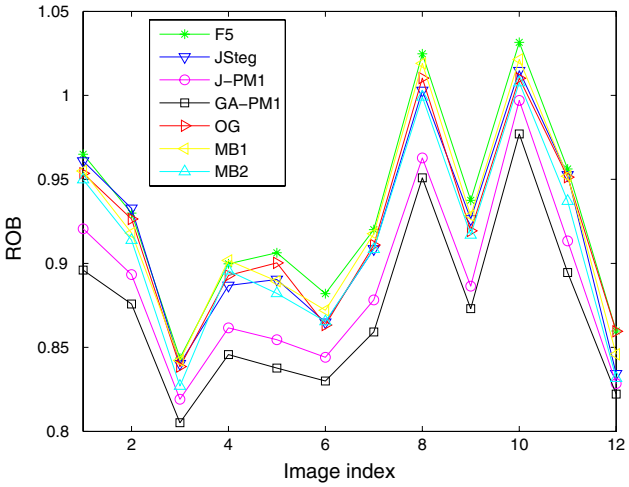

Fig. 5 ROB of different steganographic techniques at 0.1 bpc

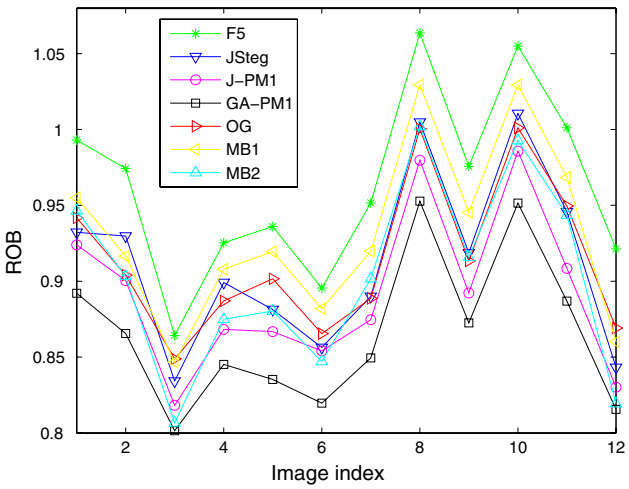

Fig. 6 ROB of different steganographic techniques at 0.3 bpc

From these figures, we can see that J-PM1 has lower ROB than the other five existing steganographic techniques, and GA-PM1 is still lower. That is, considering security in least increment of blockiness perspective, J-PM1 plays a better performance than F5, OG, MB1, MB2 and JSteg, and GA-PM1 outperforms all of them.

Capacity of F5, MB1, and MB2 is around 0.7 bpc, and capacity of JSteg is higher. So we compare ROB of F5, OG, MB1, MB2, JSteg, J-PM1and GA-PM1 at 0.7 bpc for the further research. Figure 7 shows that ROB of JSteg and J-PM1 is comparable, and is better than that of F5, OG, MB1 and MB2 at 0.7 bpc, and ROB of GA-PM1 is the best in this case.

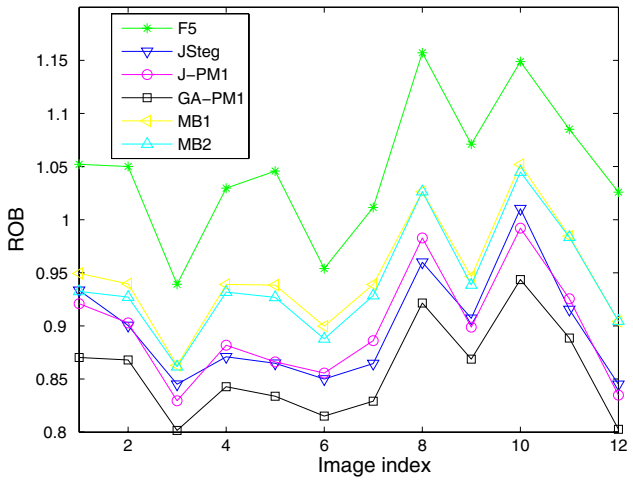

Fig. 7 ROB of different steganographic techniques at 0.7 bpc

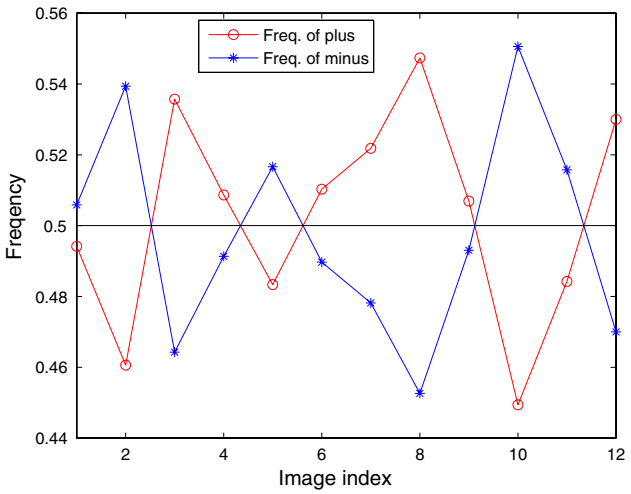

Fig. 9 Frequency of plus operation and minus operation

4.1.2 Histogram

We have analyzed that J-PM1 preserves the characteristics of histogram theoretically. Here, we will show intuitively that GA-PM1 preserves the characteristics of histogram.

As a representative example, Fig. 8 plots distribution of the (2, 1)th quantized AC components for cover image "Lena" and its corresponding stego image with an embedding rate of 0.3 bpc. The red line shows the coefficients distribution of a stego image with GA-PM1, and green bars show that of the cover image. We can see that, GA-PM1 preserves the three characteristics of histogram as presented in Sect. 3.5. This is also true for the other components (e.g., (1, 2)th, (2, 2)th AC components) and the other testing images.

The preservation of characteristics of histogram can also be proved by the probability of plus and minus decided by the GA. From Fig. 9, we can see that the line of frequency of plus and minus sways around 0.5, with deviations no more than 0.05. That is, the probability of plus and minus is nearly the same as the random process in PM1. So the theoretical proof proposed in Sect. 3.5 is also true for GA-PM1. Then, we can conclude that GA-PM1 preserves histogram characteristics.

4.2 Embedding capacity

From Fig. 10, we can see that capacity of F5, MB1 and MB2 is about 0.7 bpc, and capacity of OG is less than 0.4 bpc. Without considering security, capacity of JSteg and GA-PM1 is nearly 1 bpc, higher than the other three (i.e., F5, OG, MB1) techniques.

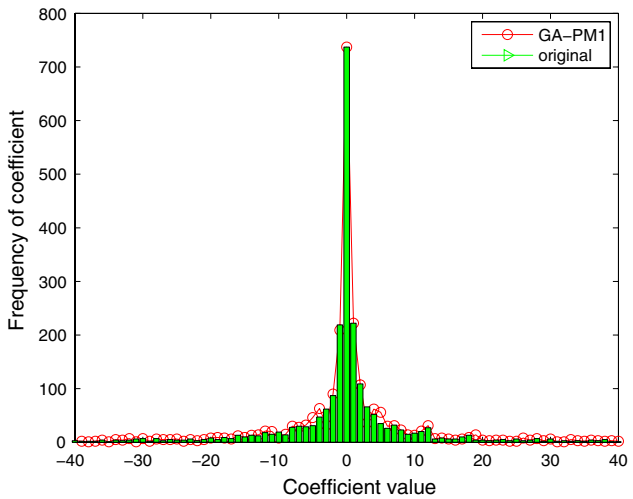

Fig. 8 Distribution of the (2, 1)th AC components

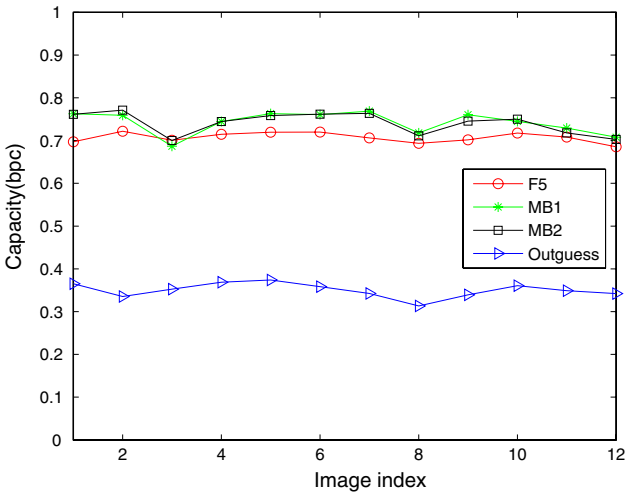

Fig. 10 Capacity of each technique

5 Conclusion

A steganography method used in JPEG images, called GA-PM1 is proposed, which is based on PM1 and GA algorithm. Using PM1 in JPEG images preserves the characteristics of histogram theoretically. By minimizing the ratio of blockiness between the stego image and its corresponding estimated image, the GA helps PM1 decide whether to increase or decrease each coefficient that needs to be modified.

GA-PM1 outperforms current typical steganography methods (i.e., F5, Outguess, MB1, MB2 and JSteg) when considering capacity, and has better security than all of them when loading the same secret message. Abundant experimental results have been provided to illustrate our method's outstanding performance both in security and capacity. Though the experiments use gray scale images as cover media, there is no constraint for the use of GA-PM1 in color images.

Acknowledgments This work was supported in part by National Natural Science Foundation of China (No. 60776794, No. 90604032, No.60702013), 973 program (No. 2006CB303104), 863 program (No. 2007AA01Z175), Beijing NSF(No.4073038) and Specialized Research Foundation of BJTU.

References

Chang CC, Chen TS, Chung LZ (2002) A steganographic method based upon JPEG and quantization table modification. Inf Sci 141: 123–138

Chu SC, Huang HC, Shi Y, Wu SY, Shieh CS (2008) Genetic watermarking for zerotree-based applications. Circuits Syst Signal Process 27 (in press)

Coley DA (1999) An introduction to genetic algorithms for scientists and engineers. World Scientific, Singapore

Fridrich J (2004) Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes. In: Proceedings sixth information hiding workshop'04, LNCS 3200, Springer, New York, pp 67–81

Fridrich J, Goljan M, Hogea D (2002a) Steganalysis of JPEG images: breaking the F5 algorithm. In: Proceedings fifth international workshop on information hiding'02, LNCS 2578, Springer, New York, pp 310–323

Fridrich J, Goljan M, Hogea D (2002b) Attacking the OutGuess. In: Proceedings ACM workshop on multimedia and security'02, France, pp 3–6

Fridrich J, Goljan M, Hogea D (2003) New methodology for breaking steganographic techniques for JPEGs. In: Proceedings SPIE, security and watermarking of multimedia contents V'03, vol 5020, pp 143–155

Goldberg DE (1989) Genetic algorithms in search, optimization and machine learning. Addison Wesley, Boston

Huang HC, Pan JS, Huang YH, Wang FH, Huang KC (2007) Progressive watermarking techniques using genetic algorithms. Circuits Syst Signal Process 26: 671–687

Li B, Huang FJ, Huang JW (2007) Steganalysis of LSB greedy embedding algorithm for JPEG images using coefficients symmetry. In: Proceedings fourteenth IEEE international conference on image processing'07, San Antonio, vol 1, pp 413–416

Michalewicz Z (1994) Genetic algorithms + data structures = evolution programs. In: AI Series. Springer, New York

Pan JS, Sung MT, Huang HC, Liao BY (2004) Robust VQ-based digital watermarking for the memoryless binary symmetric channel. In: IEICE Trans Fundam Electron Commun Comput Sci E-87A:1839–1841

Pan JS, Huang HC, Jain LC, Fang WC (eds) (2007) Intelligent multimedia data hiding: new directions. Springer, Berlin

Petitcolas FA, Anderson RJ, Kuhn MG (1999) Information hiding-a survey. In: Proceedings of the IEEE special issue on protection of multimedia content'99, vol 87, pp 1062–1078

Provos N (2001) Defending against statistical steganalysis. In: Proceedings tenth USENIX security symposium'01, Washington DC, pp 323–335

Sallee P (2003) Model based steganography. In: Proceedings international workshop on digital watermarking'03, LNCS 2939, Springer, New York, pp 154–167

Sallee P (2005) Model-based methods for steganography and steganalysis. Int J Image Graph 5: 167–189

Soukal D (2006) Advanced steganographic and steganalytic methods in the spatial domain. Diss Abstr Int 67-03(Sect B):1532–1704

Westfeld A (2001) F5—a steganographic algorithm (high capacity despite better steganalysis). In: Proceedings fourth international workshop on information hiding'01, LNCS 2137, Springer, New York, pp 289–302

Westfeld A, Pfitzmann A (2000) Attacks on steganographic systems. In: Proceedings third international workshop on information hiding'00, LNCS 1768, Springer, New York, pp 61–76

Zhang XP, Wang SZ (2005) Secure steganographic algorithm in JPEG images (In Chinese). J Electron Inf Technol 27: 1813–1817