Scientific Research

An Improved Steganography Algorithm Based on the Human Visual System

Yan WANG[1,2], Lingdi PING[3]

[1]*Dept. of Information Technology and Management, Zhejiang Police Vocational Academy*

[2]*Visiting scholar, College of Computer Science, Zhejiang University, Hangzhou, China*

[3]*College of Computer Science and Technology, Zhejiang University, Hangzhou, China*

Email: wangyan@zjjy.com.cn, ldping@cs.zju.edu.cn

Abstract: We construct and implement a new Steganography algorithm based on the human visual system to hide a large amount of information into color BMP image. By adopting this method, we can further obtain better PSNR. Moreover, we can use different parameter to satisfy different requirements when we need high PSNR or high Capacity. The method is no need of referencing the original image when extracting the hided data from the stego-image. Our experimental results show the proposed method can reach the high capacity with good image quality.

Keywords: steganography; human visual system; BMP image

1 Introduction

Steganography literally means "covered message" and involves transmitting secret messages through seemingly innocuous files. The goal is that not only the message remains hidden, but also that a hidden message is even sent. There are many tools available that can hide messages in images, audio and video files. Steganography is now in common use while cryptography has been the preferred tool for sending secret messages, relying on complex ciphers to prevent detection in the huge bandwidth of the Internet which offers an alternative or complementary approach. Steganography supports hiding messages amongst the huge volume of Internet traffic, in media files where the addition of a hidden message is difficult to detect with the human eye even if the file is viewed [1].

Both approaches can be combined to produce better protection for the information, in this case, when the steganography fails and the message can not be detected if a cryptography technique is used too. Hiding information inside images is a popular technique nowadays.

To hide a message inside an image without changing its visible properties, the cover source can be altered in "noisy" areas with many color variations, so less attention will be drawn to the modifications.

The most common methods to make these alterations involve the usage of the least-significant bit (LSB) developed by Chandramouli [2], masking, filtering and transformations on the cover image. Dumitrescu et al. [3], construct an algorithm for detecting LSB Steganography. Von Ahn *et al.* [4] are the information theoretic scientists; they construct mathematical frameworks based on complex theoretic view to seek the limits of steganography. Other construction is used by HweeHwa Pang [5]; his scheme used hash value obtained from a file name and password and a position of header of hidden file is located. This approach is used by the present work with new modifications. The next interesting application of steganography is developed by Miroslav Dobsicek [6], where the content is encrypted with one key and can be decrypted with several other keys, the relative entropy between encrypt and one specific decrypt key corresponds to the amount of information. Because of the continual changes at the cutting edge of steganography and the large amount of information involved, steganalysists have suggested using machine learning techniques to characterize images as suspicious or non- suspicious developed by Mittal *et al.* [7]. Pavan et al. [8] used entropy based technique for detecting the suitable areas in the document image where information can be embedded with minimum distortion. C. Zhang et al. [9] hides indirectly the secured binary bits along with some selected graphical image bits, based on the neural network algorithm, to get cipher bits. This approach is used by the present work. When using a 24 bit color image, a bit of each of the red, green and blue color components can be used, so a total of 3 bits can be stored in each pixel. Thus, an 800 × 600 pixel image can contain a total amount of 1.440.000 bits (180.000 bytes) of secret information. But using just 3 bit from this huge size of bytes is wasting in size. So the main objective of the present work is how to insert more than one bit at each byte in one pixel of the cover-image and give us results like the LSB [2], [3] (message to be

Project support: The Excellent Young Teachers Program of Zhejiang Provice.

imperceptible). This objective is satisfied by building new steganography algorithm based on an intelligent system to hide large amount of any type of information through bitmap image[4], [6] by using maximum number of bits per byte at each pixel. We discuss two kinds of attacks to be sure that our process for embedded information is worked well. The first discussion concerns to work against visual attacks [7], [10] to make the ability of humans is unclearly discern between noise and visual patterns, and the second discussion concerns to work against statistical attacks[3], [8], [11], [12] to make it much difficult to automate.

2 Data Hiding in Images Based on the Pixel-Value Differencing

In 1999, W.N.Lie and L.C. Chang proposed [13] a method of data hiding which is called adaptive numbers of least significant bits based on the human visual system. They assume that the secret data are in a form of binary bit streams which can be raw or compressed multimedia, or simply a series of texts conveying messages for transmission into a host image. This method is based on the human visual system. Therefore, in such a way that gray value change of each pixel in the host image after embedding is beyond human perception. In addition to this way, the embedded secret data can be extracted without the knowledge of the original host image. Before this research, they have proposed another method which is for embedding multimedia information into a host image subject to the constraint of human visual perception. Although it is adopted in classical LSB, there are different gray levels with adaptive numbers of LSBs for pixels. This adaptivity is self-extractable. For example, there are the numbers of LSBs for embedding which can be figured out without extra overhead from the modified pixel value. So, there is a piecewise mapping function subject to human visual sensitivity of contrast that is used to achieve this purpose. It is main point of their idea. Let's introduce this main function. $N(g)$ is LSB-mapping function and it gives the number of LSBs which can be embedded for each possible gray level g. It is to be devised. After embedding, it must be denoted by g' when let the pixel value, then the condition of no overhead for this adaptation is that the $N(g)$ values before and after pixel modification should remain unchanged. For instance, $N(g')=N(g)$ can explain this concept. In Fig.1, they describe this concept and "h" is an embedded secret. So, according to formula $N()$,we can put a variable "g" into $N()$ and get "g'" which is added some bits. On the other hand, "g'" can be extracted when a receiver uses formula $N()$ to get embedded data.

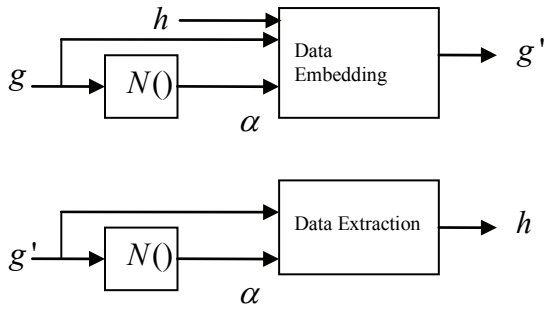


Figure 1. The concept of self-contained number of LSB's for data embedding and extraction

Equation (1) represents the $N(g)$ formula. "g" can be known that which range it is placed. Therefore, we can also know how much number of LSBs is used. For example, if "g" is placed between k_1 and k_2, we can know that two LSBs are used.

$$N(g)=\begin{cases}1 & 0\le g\le k_1\\ 2 & k_1\le g<k_2\\ 3 & k_2\le g<k_3\\ 4 & k_3\le g<256.\end{cases} \quad (1)$$

According to the LSB-mapping function $N(g)$ (1) and the estimated $k_i^{'}$ which is defined at Table I, we can easily calculate how much bits that we can hind in the pixels. The result is described that if "g" is placed between 0 and 87, two LSBs are used. We also can infer three and four LSBs.

Table 1. Instance of modified decision boundaries

$k_i^{'}$	Interval	Number of LSBs used
	None	1 bit
0	0~87	2 bits
88	88~191	3 bits
192	192~255	4 bits

Chang and Tseng proposed to use side information of neighboring pixels to estimate the number of bits which can be carried in the pixel of the host-image to hide the secret data. However, there is still a disadvantage in the research because some embedded secret bits may be extracted wrongly from the stego-image therefore the secret data cannot be recovered. In order to resolve this problem, Li et. al. suggests a method to improve Chang and Tseng scheme. Therefore, they can embed secret data which can be recovered properly from the stego-image. Besides, they proposed a new adaptive LSB substitution based on the pixel-value differencing. In their article [14], two neighboring pixels of the input pixel are used to determine the number of bits which is

embedded in the pixel. Moreover, the secret information is then embedded into the host-image by a simple LSB substitution method with optimal pixel adjustment process (OPAP) proposed by Chan and Cheng [15].

3 Proposed Steganography Algorithm

We develop an improved LSBs substitution method based on Lie and Chang's [13] scheme. In order to reduce the distortion after embedding the secret data, we can apply OPAP [15] into Lie and Chang's scheme. In the Lie and Chang's scheme, the amount of embedded message in each pixel is depended upon the grayscale values g in the cover image. According the LSB-mapping function $N(g)$ in Equation (2) and the estimated $k_i^{'}$ which is defined in Table II, we can easily calculate how much bits that we can hind in the pixels. For instance, if the $g=100, N(g)=3$ is calculated. Then the secret data substitute the LSBs of g directly, and we can obtain the new grayscale value $g^{'}$. The most important is that the received $N(g^{'})$ should be consisted with $N(g)$ in the extraction part. Deservedly, we can perfectly obtain the secret data. However, it will be produced great distortion between the value g and the value $g^{'}$. In order to reduce the distortion, we adopt the OPAP in the replacement procedure. Unfortunately, there will be the error occur when we directly use OPAP. In the OPAP scheme, if the embedded error is great than 2^{k-1} (that is $|g^{'}-g|>2^{k-1}$), we should compensate 2^k for the value $g^{'}$. Therefore it will cause of $N(g)\neq N(g^{'})$. In the other word, the value $g^{'}$ may fall into other interval when the original value g is close to the boundary k_i. In order to solve this problem, we should redefine the interval. By adding the new confusing interval like Equation (3), we can make sure that the embedded value $g^{'}$ fall into the confusing interval, we can let that pixel hide nothing and change the original value g to the nearly confusing value. Finally we can make the LSB-mapping function $N()$ working correctly in both g and $g^{'}$. In addition, we also perfectly extract the secret data in each pixel and increase the PSNR efficiently. Our experiment will prove this scheme is feasible.

$$N(g)=\begin{cases}1 & 0\le g<k_1,\\ 2 & k_1\le g<k_2,\\ 3 & k_2\le g<k_3,\\ 4 & k_3\le g<256.\end{cases} \quad (2)$$

Table 2. Instance of modified decision boundaries

$k_i^{'}$	Interval	Number of LSBs used
	None	1 bit
0	0~87	2bits
88	88~191	3bits
192	192~255	4bits

$$N(g)=\begin{cases}2 & 0\le g<k_1^{'}-c_1,\\ 0 & k_1^{'}\le g<k_1^{'}+c_2,\\ 3 & k_1^{'}+c_2\le g<k_2^{'}-c_3,\\ 0 & k_2^{'}-c_3\le g<k_3^{'}+c_4,\\ 4 & k_3^{'}+c_4\le g<256.\end{cases} \quad (3)$$

4 Implementation of Proposed Algorithm

4.1 Estimation of Image Quality

It can roughly be separated into two parts about estimating the image quality which is altered by compression or data hiding. The first method directly makes use of human vision system. It is simple for a human vision system if there are huge changes in the specific place of the image, such as correction the smooth area in the image. However, it is hard mostly for human eyes to detect the slight difference of the two images in tolerable range. Moreover, the sense of sight in each man is not the same so it is difficult to establish an estimative standard. By using Peak Signal to Noise Ratio (PSNR) to measure the image quality is the other objectively estimative standard. The formula is shown below:

$$PSNR=10\times\log(\frac{255^2}{MSE}) \quad (4)$$

$$MSE=\frac{1}{m\times n}\sum_{i=1}^{m\times n}(S_i-C_i)^2 \quad (5)$$

In Equation (4) and (5), MSE is Mean Square Error of original image and modified image; S_i and C_i are separately represented the original pixel and the altered pixel in the same position of the image; m and n are separately representative of the width and height in the image. By the way, the higher PSNR show the better image quality. In general, it is difficult for human eyes to discriminate the difference between the two images when PSNR is larger than 30dB.

The higher PSNR is not enough to show that the image quality is better in detail. Besides PSNR, we also need the human eyes to assist us in deciding the image quality.

4.2 Results of Experiment

Here we use a random generated bit stream to be our secret message, and choose four classic cover-images

978-1-935068-10-5 © 2010 SciRes.

namely 'Lena', 'Baboon', 'F16'and 'Pepper' which consists of 512 by 512 pixels.

In the following, we improve Lie and Chang proposed [13] scheme which was based the human visual system. Therefore we can embed different number of bits into each pixel of one cover image. If we know how many numbers bit could be replaced in each pixel, we can further employ OPAP [15] scheme. Furthermore, according to results of Lie* and Chang proposed, we can design the different value of $k_1^{'} \sim k_3^{'}$ to handle different requirement such as higher capacity or higher PSNR. However, we need set the additional different confusing interval to overcome error. For example, Table III and Table IV show the new confusing interval can be calculated by the decided parameter $k_i^{'}$. In practice, we can subtract or add certain value according to the number of LSBs used. If the original $k_2^{'}$ is equal to 120, we determine 120 is the medium value in confusing interval. In the original scheme, we should replace three LSBs from a pixel when the pixel value is larger than $k_2^{'}$. So the maximum distortion between g and $g^{'}$ is equal to 4. Thus we can estimate that the confusing interval is at least 4 in order to prevent the overflow of compensation. In general, we can determine 122 is the upper bound of confusing interval and 118 is the lower bound of confusing interval. It is the same as other $k_i^{'}$.

In theory, if a cover-image has a uniform distribution in the grayscale values, we can calculate optimal capacity is

$$\frac{118\times 2+97\times 3+28\times 4}{256}\times 512\times 512=654336$$

bits. However, the original grayscale value g may probably falls into the confusing interval after embedding secret data, so we cannot hide any data in these pixels. Therefore we make the values $g^{'}$ is equal to the boundary value of confusing interval. In addition, we choose the closest one to replace the original value g. Unlike uniform or Gaussian distribution actually a nature image does not have a statistics distribution in its grayscale histogram. Figure 2 shows the histogram of each cover-image. But the example results show that the performance approximates the optimal value. We represent these results of selecting different parameters $k_1^{'} \sim k_3^{'}$ in Table IV.

For the sake of coming to the same based line, we choose the larger PSNR to compare with Li and Leung proposed scheme. Those schemes are all the unfixed LSB replacement. Here, we select

$$k_1^{'}=0, k_2^{'}=120, k_3^{'}=224 .$$

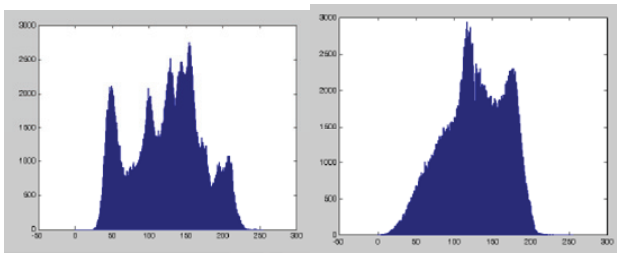
Table V presents the comparison with two previous schemes. The first scheme proposed by Li and Leung [14] which was based on the pixel value differencing , and the other one proposed by Lie and Chang[13] which was utilized side information of neighboring pixels to estimate the number of bits which was carried in the pixel of cover image. Figure 3 also proved our capacity is better than the others in the higher PSNR.

Table 3. Improved decision boundary (a)

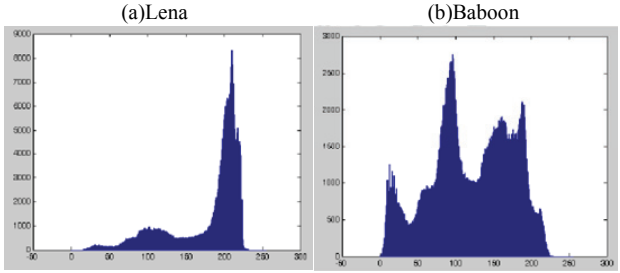
$k_i^{'}$	Interval	Number of LSBs used
	None	1 bit
0	0~117	2bits
	118~122	0bits
120	123~219	3bits
	220~228	0bits
224	229~256	4bits

Table 4. Improved decision boundary (b)

$k_i^{'}$	Interval	Number of LSBs used
	None	1 bit
0	0~85	2bits
	86~90	0bits
88	91~187	3bits
	188~196	0bits
192	197~255	4bits



(a)Lena (b)Baboon



(C)F16 (D) Pepper

Figure 2. The histogram of cover-image

Table 5. Results of image embedding in different set of $k_i^{'}$

$k_1^{'}$	$k_2^{'}$	$k_3^{'}$	Cover image	Capacity	PSNR(dB)
0	88	192	Lena	683,798	40.8112
0	88	192	Baboon	679,200	41.5452
0	88	192	F16	741,383	36.8157
0	88	192	Pepper	618,675	41.5411
0	120	224	Lena	621,177	42.6630
0	120	224	Baboon	625,569	42.6306
0	120	224	F16	647,469	41.7070
0	120	224	Pepper	635,703	42.7833
0	136	224	Lena	594,584	43.2649
0	136	224	Baboon	599,649	43.3260
0	136	224	F16	641,876	41.8185
0	136	224	Pepper	613,266	43.1778
0	200	224	Lena	518,248	45.9118
0	200	224	Baboon	519,859	46.3799
0	200	224	F16	467,706	43.7860
0	200	224	Pepper	519,708	46.0881

Table 6. Results of comparison with other schemes

Cover image	Li, Leung[14]		Lie, Chang[13]	Proposed scheme		
	Capac-ity	PSNR	Capacity	PSNR	Capacity	PSNR
Lena	494,806	42.49	674,595	39.5860	621,177	42.66
Baboon	767,058	37.69	675,500	39.6446	625,569	42.63
F16	474,473	42.20	744,513	38.4913	64,469	41.71
Pepper	503,286	42.52	656,873	39.9805	635,703	42.78

Figure. 3 (a) Original cover-image 'Lena' **(b)Stego-image with** $k_1^{'}=0, k_2^{'}=120, k_3^{'}=224$

5 Discussion

From the above experimental results, we can find that the capacity is deeply affected by the distribution of grayscale histogram. So that is why the image "F16" always has a larger capacity than the other images. In Table V, [14] and [13] have a common problem in the extraction processing. Because the amount of hided data is calculated by the two fixed pixels, they should keep the upper and left pixels changeless. If there is some faults happened in the upper or left pixel, it will influence a whole column or row in the stego-image. But in our scheme, we do not have the same problem because we use an independent hiding method to embed our secret data. Therefore, whatever which pixels id altered, it will not influence the next extracted bit. Tests prove that our proposed method has a better performance than other algorithm like LSB.

6 Conclusions

In this thesis, we propose a high capacity hiding methods to embed secret data into the cover image without notable distortions. When a receiver wants to extract the embedded message, he does not need to reference the original image. The proposed method is based on the scheme proposed by Lie and Chang. Furthermore, we use OPAP and set new confusing interval to improve the original scheme. Finally, we compare the results with the Lie and Li's scheme. Our PSNR can be better than their results with the higher capacity.

References

[1] S-tools(http://digitalforensics.champllain.edu/download/s-tools4.zip).

[2] Chandramouli, R. and N. Memon,. "Analysis of LSB based image steganography techniques," Proc. of ICIP, Thessaloniki, Greece, pp. 7-10, Oct. 2001.

[3] Dumitrescu, S., W. Xiaolin and Z. Wang, "Detection of LSB steganography via sample pair analysis," In: LNCS, Springer-Verlag, New York, Vol. 2578/2003, pp: 355–372, 2003.

[4] Ahn, L.V. and N.J. Hopper, "Public-key steganography. In Lecture Notes in Computer Science of Advances in Cryptology," EUROCRYPT 2004, Vol. 3027 / 2004, Springer-Verlag Heidelberg, pp: 323–341, 2004.

[5] Pang, H.H., K.L. Tan and X. Zhou, "Steganographic schemes for file system and b-tree," IEEE Trans. on Knowledge and Data Engineering, Vol. 16, pp.701–713, 2004.

[6] Dobsicek, M., "Extended steganographic system," In: 8th Intl. Student Conf. on Electrical Engineering. FEE CTU. 2004

[7] Mittal, U. and N. Phamdo, "Hybrid digital-analog joint source-channel codes for broadcasting and robust communications," IEEE Trans. On Info. Theory, vol. 48, pp. 1082 –1102, 2002.

[8] Pavan, S., S. Gangadharpalli and V. Sridhar, "Multivariate entropy detector based hybrid image registration algorithm," IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, pp: 18-23, 2005.

[9] C. Zhang, H.W. Guesgen, W.K. Yeap "Neural Based Steganography, Lecture note in computer science Computational Intelligence. Neural Networks," LNAI 3157, pp. 429–435, Springer-Verlag Berlin Heidelberg 2004.

[10] Moulin, P. and J.A. O'Sullivan, "Information-theoretic analysis of information hiding," IEEE Trans. on Info. Theory, vol. 49, pp. 563–593, 2003.

[11] Amin, P., N. Liu and K. Subbalakshmi, "Statistically secure digital Image data hiding," IEEE Multimedia Signal Processing MMSP05, China, 2005.

[12] Jackson, J., G. Gunsch, R. Claypoole and G. Lamont, "Detecting novel steganography with an anomaly- based strategy," J. Electr. Imag. Vol. 13, 860– 870, 2004.

[13] W.N. Lie and L.C. Chang, "Data hiding in images with adaptive numbers of least significant bits based on the human visual system". Proc.IEEE Internat. Conf. Image Process.1,286-290,1999.

[14] S.L. Li, K.C. Leung, L.M. Cheng and C.K. Chan, "Data hiding in image by adaptive LSB substitution based on the value differencing". Proc. IEEE ICICIC'06 2006.

[15] C.K. Chan and L.M. Cheng, "Hiding data in images by simple LSB substitution", Pattern Recognition 37,469-474, 2004.