SECURE BLIND IMAGE STEGANOGRAPHIC TECHNIQUE USING DISCRETE FOURIER TRANSFORMATION

Faisal Alturki

Department of Electronic Engineering
College of Technological Studies
Shuwaikh, Kuwait 70654
faisaleng@hotmail.com

Russell Mersereau

Center for Signal and Image Processing
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
rmm@ece.gatech.edu

ABSTRACT

In this paper, we present a new steganographic technique for covert communications. The technique embeds the hidden information in the DFT domain after permuting the image pixels in the spatial domain using a key. The permutation process introduces randomness into the cover image and results in a significant increase in the number of transform coefficients that can be used to transmit the hidden information. The hidden information is embedded using quantization technique. The perceptual and statistical properties of the cover and the stego images remain similar for small quantization steps. The security of the system is examined against known-stego attack.

1. INTRODUCTION

Digital steganography is the art of secretly hiding information inside a multimedia signal in such a way that its very existence is concealed. The big revolution in the field of computer technology and speed, make the job of cryptoanalyst easy in breaking complicated codes. Therefore, it is important in some multimedia and covert communication applications not only to encrypt the message, but to conceal its very existence [1].

In steganographic applications emphasis is concentrated on security and on preventing detection of any hidden information. For an overview of current steganographic techniques one can refer to [2]. In this paper we describe a steganographic technique for embedding a large amount of information inside a cover image while maintain high security. The technique is based on the principle of decorrelating the cover image samples. Decorelating the image samples results in a two-dimensional set of uncorrelated and identically distributed samples. These samples have a Gaussian distribution in the transform domain. The scrambling process results in equalization to the image spectrum, i.e., the image energy is uniformly distributed over all frequency bands [3]. Image decorrelation is obtained by using a key, which permutes the pixels locations . Data embedding and extraction is performed using quantization principles. The transform coefficient magnitudes are modified in such a way that they resolve whether a binary "1" or "0" is embedded when divided by a quantization step Δ.

2. EMBEDDING SYSTEM

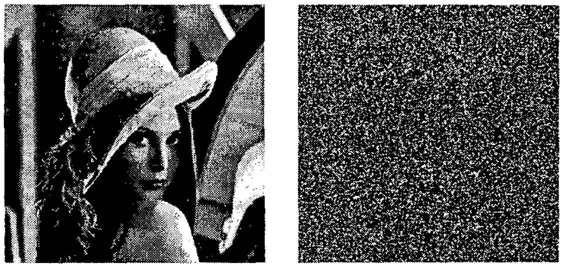
The embedding process starts by decorrelating the image samples, to increase the number of coefficients that can be utilized to embed information. The decorrelation process removes the redundancy in the original image and introduces significant mid and high frequency components into the cover image. The decorrelation process is done using some key $K \in \mathcal{K}$, where $\mathcal{K}$ is some key space. The key scrambles the image pixels so that the resulting image looks like white noise to the viewer. Let $x(n_1, n_2)$ be some natural image of size $M \times N$ where some information is to be embedded inside it. Let $x_\mu(n_1, n_2) = x(n_1, n_2) - \mu$, where μ denotes the average value of $x(n_1, n_2)$ and let $\mathcal{S}$ be some scrambling operator, operating on $x(n_1, n_2)$ in a lossless and one-to-one fashion. $\mathcal{S}$ basically permutes the pixels locations of $x_\mu(n_1, n_2)$. The key $\mathcal{K}$ generates a one-dimensional vector $\mathcal{V}$ of random integers of the same size as the cover image, i.e., one-dimensional vector of size MN. The cover image is also converted into a one-dimensional vector $\tilde{x}$ of size MN. The key scrambles the elements (pixels) of $\tilde{x}$ and then converts the resulting vector into a two-dimensional signal $\overset{*}{x}_\mu(n_1, n_2)$, i.e.,

$$\overset{*}{x}_\mu(n_1, n_2) = \mathcal{S}(x_\mu(n_1, n_2), K), \qquad (1)$$

where $\overset{*}{x}_\mu(n_1, n_2)$ denotes the uncorrelated image. Whitening an image has a great impact on its spectrum in the transform domain, because it spreads the signal energy uniformly over the entire image spectrum. From a communication

0-7803-6725-1/01/$10.00 ©2001 IEEE

theory prospective, if we model the host image as a channel, then the process of whitening the host image increases the number of transform coefficients which can be used for transmit the hidden information. This is equivalent to an increase in the channel transmission bandwidth, which results in an increase in the data embedding rate. Figure 1 shows an example of an input and output images going into and out of a scrambler. Next, we take the DFT of $\overset{*}{x}_\mu(n_1, n_2)$, the transform signal $\overset{*}{X}_\mu(k_1, k_2)$ is decomposed into real and imaginary parts. Half of the information is embedded in the teal part and the other half into the imaginary part. The embedding is done in a way that takes into consideration the symmetry in the DFT coefficients so the resulting image remains real. To embed the data, we quantize the magnitude of both the real and the imaginary parts of the DFT coefficients to the nearest integer multiple of 2Δ, where Δ is some quantization step size. The value of Δ is selected in such a way that it satisfies the following constraints. Δ should be small to maintain good perceptual quality and low perceptual distortion, but large enough to resolve the value of the binary embedded bit unambiguously. For example a coefficient can be rounded to an even integer multiple of Δ to embed binary "1", and to an odd multiple of Δ to embed binary "0". When the embedding process is completed, we combine the quantized coefficients along with their corresponding phases, i.e., their signs and take the inverse DFT. Next we add back the DC value and then descramble the image using the same key K to get the stego image $y(n_1, n_2)$.



(a) Original image. (b) Decorrelated image.

Fig. 1.

The decoding process is straightforward. The average value of the received image is removed. Next, the resulting signal is decorrelated using the same key K. Then the DFT is computed, the magnitudes of the real and the imaginary parts are extracted and each coefficient is divided by Δ. The output of the division is either an even or an odd number as shown below

$$\hat{d}_i = \begin{cases} 1 & \text{if } \lfloor \frac{\hat{Y}_{i,j}}{\Delta} \rfloor \text{ is even} \\ 0 & \text{if } \lfloor \frac{\hat{Y}_{i,j}}{\Delta} \rfloor \text{ is odd ,} \end{cases} \tag{2}$$

where $\hat{Y}_{i,j}$ represents the (i, j)-th received coefficient and $\lfloor \cdot \rfloor$ represents the floor function.

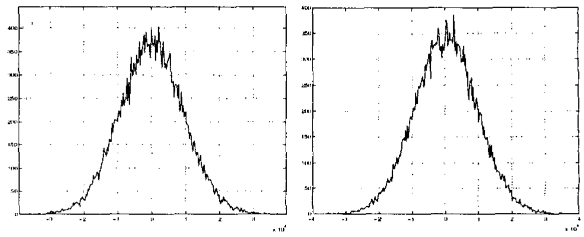
3. SYSTEM ANALYSIS

The image $\overset{*}{x}_\mu(n_1, n_2)$ is modeled as a two-dimensional random process. The pixels amplitudes are modeled as uncorrelated and identically distributed random variables. Each pixel is modeled as a discrete random variable that is uniformly distributed between $\{-\mu, 255 - \mu\}$ i.e.,

$$f_{\overset{*}{x}_{\mu_{i,j}}} = \begin{cases} \frac{1}{256} & \text{if } -\mu \le \overset{*}{x}_\mu \le 255 - \mu \\ 0 & \text{elsewhere,} \end{cases} \tag{3}$$

where $\overset{*}{x}_{\mu_{i,j}}$, represents the spatial coefficient at position (i, j).

Each DFT coefficient is expressed as a weighted sum of uncorrelated identically distributed random variables. By the central limit theorem [4], it is easy to show that the marginal pdf of the real and the imaginary parts of these coefficients follow a Gaussian distribution with zero mean. Figure 2 shows the marginal pdfs of the real and the marginal parts of the DFT coefficients. The process of adjusting the transform coefficients to accommodate the embedded bits introduces noise into the DFT coefficients [5]. This noise is independent of the coefficients and it is modeled as an additive noise $\mathcal{U}$ uniformly distributed between $[-\Delta, \Delta]$ i.e.,



(a) Real part. (b) Imaginary part

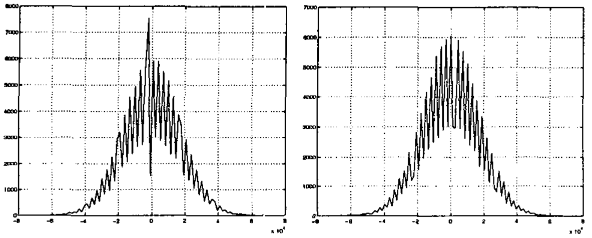
Fig. 2. Histogram of DFT coefficients.

$$\mathcal{Q}[\overset{*}{X}_\mu] = \overset{*}{X}_\mu + \mathcal{U}. \tag{4}$$

The pdf of $\mathcal{Q}[\overset{*}{X}_\mu]$ can be found by convolving the densities of the two components [4]. However, if the value of Δ is selected to be very small, i.e.,

$$\Delta << \{\sigma_{\overset{*}{X}_R}, \sigma_{\overset{*}{X}_I}\} \tag{5}$$

where $\sigma_{\overset{*}{X}_R}$ and $\sigma_{\overset{*}{X}_I}$ denotes the standard deviation for the real part and the imaginary part of $\overset{*}{X}(k_1, k_2)$, then its pdf can be approximated by an impulse. The result of convolving this signal with the pdf of the DFT coefficients can be approximated by a Gaussian pdf. Therefore, the distribution of $Q[\overset{*}{X}_\mu]$ is approximated by a Gaussian distribution with zero mean. Next, we add the binary data which are mapped to polar format with amplitude $\pm\frac{\Delta}{2}$ to get the coefficient of the stego image $Y(K_1, k_2)$. Figure 3 shows the pdf of the quantized coefficients.



(a) Real part. (b) Imaginary part.

Fig. 3. Histogram of the quantized DFT coefficients

4. SYSTEM SECURITY

In steganographic systems, security is related to the prevention of detection of any hidden communication by a hostile party [2]. For known-stego attack it can be shown that a system can never be secure if the cover signal is deterministic [6]. Therefore, the cover signal must have some randomness in order to achieve security. The whitening process introduces uncertainty into the cover image. For an image of size $N \times M$, there are $(NM)!$ possible permutations for pixels locations. Typically images are big enough in size such that $(MN)!$ is a very large number. From steganography prospective, the scrambling process introduces randomness into the host signal. Each key results in a different realization which can be thought of as a different cover image used for data embedding. An observer who may have access to the original image, does not know which permutation was selected to embed the hidden information, that's where the randomness comes into place. If we let $\mathcal{X}$ represents the space of all possible realizations, then $H(\mathcal{X}) > H(x)$, i.e., the uncertainty of the cover source is larger than the uncertainty associated with the original cover image.

For known-stego attacks it is assumed that the attacker has access to the original and the stego images and knows the embedding algorithm. However, the attacker does not have the scrambling key K. The security of the system is verified by examining the distribution of the cover and the stego images in the spatial and transform domain. According to [7] a stego system is called ϵ-secure if

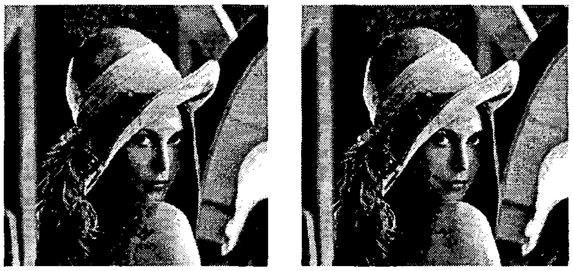
$$D(P_C \| P_S) = \sum P_C \log \frac{P_C}{P_S} \leq \epsilon, \qquad (6)$$

where P_C represents the distribution of the cover object, P_S represents the distribution of the stego object and $D(P_C \| P_S)$ represents the relative entropy between the two probability distributions [8]. For a perfectly secure system we must have $D(P_C \| P_S) = 0$. Furthermore, if an observer examines the difference between the two images, the resulting signal looks like white noise which has Gaussian distribution in both spatial and transform domains. Since the Gaussian noise has the highest uncertainty of all distributions for a given variance [8]. By examining the residual signal it is very difficult to assert whether this signal resulted from covert communication or due to some random transmission noise. Even if an observer suspects that some covert communication is taking place, it is not possible to extract any useful information by observing the difference signal.

5. EXAMPLE AND RESULTS

An example which illustrates the concepts presentedin this paper is given in

this section. The image LENNA of size 512 x 512 is used as a cover image, and the image PEPPERS, also of size 512 x 512, is used as a hidden image. The embedded image is first compressed using any lossy compression scheme such as JPEG compression. The resulting file is converted into a vector of binary data. The binary data is then encrypted to produce the embedded sequence $d(l)$. The data embedding rate is 1 bit per coefficient. Figure 4 shows an example of an original an stego images.



(a) Original image. (b) Stego image.

Fig. 4.

The hidden image is shown in Figure 5.

In this example, we used a value of $\Delta = 1000$, The average value of the standard deviation for the real part of the cover image is $\sigma_{\overset{*}{x}_R} = 15065$, and for the imaginary part

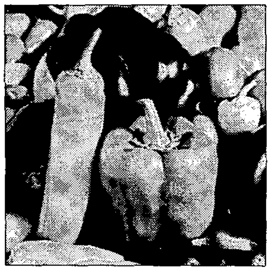
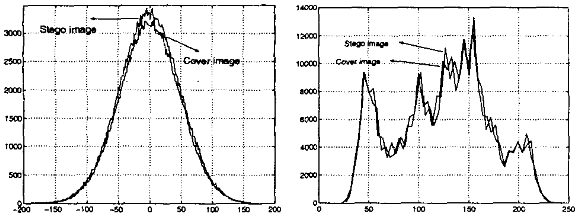


Fig. 5. Hidden image.

is $\sigma_{\dot{x}_I} = 15632$. The average value of the standard deviation for the real part of the stego image is $\sigma_{\hat{y}} = 15088$, while the average value of the standard deviation for the imaginary part of the stego image is $\sigma_{\hat{y}} = 15678$. These values were obtained by averaging over 50 different simulation runs with different keys. Based on the values obtained for σ, the ratio of the pdfs for the real part and the imaginary parts of the original and the stego images is close to one. Therefore, the logarithm of the ratio of the two pdf's is small which indicate that the system is secure. Histograms of the cover and the stego images in spatial and transform domains are shown in Figure 6 The difference between the original image and the stego image and the marginal pdf of the difference (residual) signal in the spatial domain are shown in Figure 7.



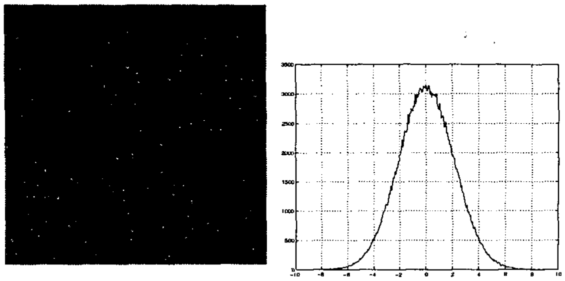
(a) Transform domain.

(b) Spatial domain.

Fig. 6. Histograms of stego and cover images

6. CONCLUSION

A data hiding technique for steganographic applications is presented. The hidden data is embedded in the DFT domain of the cover image. The proposed algorithm attempts to maximize the data embedding rate while maintaining high security against data extraction or detection. Security against stego known attack is examined assuming a passive observer. Future work will concentrate on improving the robustness of the technique considering an active observer.



(a) Residual image.

(b) Histogram of residual image.

Fig. 7.

7. REFERENCES

[1] F. Petitcolas, R. Anderson, and M. Kuhn, "Information hiding: A survey," *Proc. IEEE*, vol. 87, no. 7, pp. 1062–1078, July 1999.

[2] S. Katzenbeisser and F. A. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, Boston, MA, 2000.

[3] T. Liang and J. Rodriguez, "Improved watermarking robustness via spectrum equalization," *Proc of IEEE. ICASSP'2000*, vol. IV, pp. 1955–1958., June 2000.

[4] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw Hill, New York, 1991.

[5] J. Eggers and B. Girod, "Quantization watermarking," *Proc. of IS & T/SPIE: Electronic Imaging II International Conference on Security and Watermarking of Multimedia contents ,San Jose, California*, January 2000.

[6] J. Zöllner, H. Federrath, H. Klimant, A. Pfitzmann, R. Piotraschke, A. Westfeld, G. Wicke, and G. Wolf, "Modeling the security of steganographic systems," in *Information Hiding: Second International Workshop, Lecture Notes in Computer Science volume 1525*, D. Aucsmith, Ed., pp. 344–354. Springer-Verlag, Germany, 1998.

[7] C. Cachin, "An information theoretic model for steganography," in *Information Hiding: Second International Workshop, Lecture notes in Computer Science vol 1525*, D. Aucsmith, Ed., pp. 306–318. Springer-Verlag, Germany, 1998.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley, New York, 1991.