ACEAIT-3055
High-Capacity Steganography Using MRF-Synthesized Cover Images

Chaur-Chin Chen and Wei-Ju Lai
Department of Computer Science National Tsing Hua University Hsinchu 30013, Taiwan
e-mail: cchen@cs.nthu.edu.tw

Abstract

Steganography [6][13] refers to embedding information or secret message into media. This paper presents a simple and secure high-capacity steganographic algorithm for information hiding [15]. We synthesize a cover-image texture with four gray levels 32, 96, 160, and 224 of user-selected size based on a Markov Random Field (MRF) model [3]. On the other hand, each byte of the secret information (secret message, image, etc.) is first encrypted based on an exponential modular arithmetic which is then partitioned into two 4-bit words. Each 4-bit word represented as an integer value in [0, 15] is inserted into a pixel in the selected *cover-image* to form a *stego-image*. The embedding capacity for an m by n cover-image could be as high as (m × n)/2, an experiment is illustrated for the proposed methodology.

Keyword: Cover-Image, Encryption, Markov Random Field, Stego-Image.

1. Introduction

Internet has become the most popular way for communication and information broadcast. Most of the Internet users either satisfy or ignore the current security and privacy of communication over network transmission until their information is stolen or misused [6]. Thus motivated, data hiding [5][13] plays an important role recently. On the other hand, the cost of storage space is significantly dropped and the bandwidth of network is good enough to transmit an image that recalls the adoption of an ancient methodology, steganography, to serve as an act of covert communication. Despite Steganography using JPEG compressed images based on the discrete cosine transform (DCT) were widely studied in the past decade [10][12][9] [7], this paper provides a simple, secure, and reversible data hiding system [8] based on the flowchart given in Figure 1. This novel steganographic framework has two distinct properties: (1) the secret message, for example, personal ID, bank account numbers, private letters are first encrypted by using simple algorithms like Hill ciphers or random permutations [11]. The goal is to raise the security level, (2) the cover-image

is chosen from artificial microarray image templates [4] or texture images synthesized from Markov random fields. The advantage is that we can easily generate these cover images to fit the various sizes of secret messages. This report describes an embedding program and an extraction program for steganography including an Ising Markov random field image synthesizer, a random permutation based on Number Theory, in addition to least significant bit replacement (LSB) [1][2].

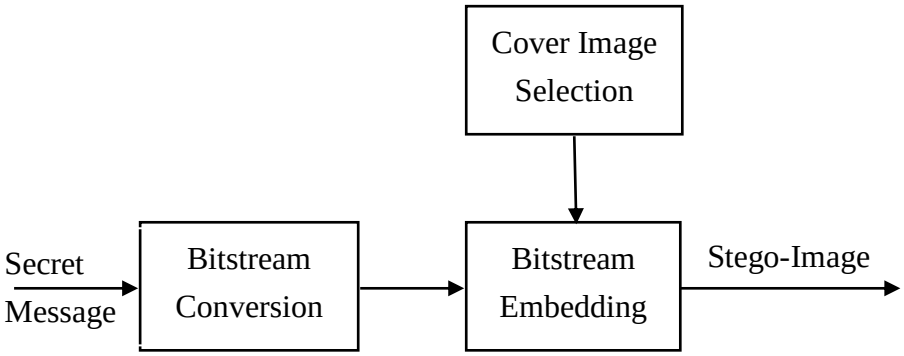


Figure 1. **A Flowchart of Proposed Steganography.**

2. Cover Images Generated by Markov Random Fields

Using Markov Random Field (MRF) Models to synthesize textures is a challenging task. We will review MRF and give algorithms for synthesizing textures [3].

2.1 Background of Markov Random Field

Let *x*, an $M \times N$ texture pattern, be represented as a matrix whose elements take values from the set $A = \{0, 1, \ldots, G-1\}$. Let $\Omega = \{x | x_t = x(i,j) \in A\}$, be the set of all possible texture patterns, and let $S = \{1, \ldots, MN\}$ be the sites of a matrix ordered by a raster scan. A Gibbs random field (GRF) is a joint probability mass function defined on Ω which satisfies

$$P(x) = e^{-U(x)}/Z \tag{1}$$

where $U(x)$ is the energy function and $Z = \sum_{y \in \Omega} e^{-U(y)}$ is the partition function.

A Markov random field is a Gibbs random field whose probability mass function. satisfies the following conditions.

(a) Positivity: $P(X = x) > 0$ *for all* $x \in \Omega$.

(b) Markov Property:

$$P(X_t = x_t | X_r = x_r, r \neq t) = P(X_t = x_t | X_r = x_r, r \in R_t) \quad \forall t \in S,$$

where R_t *is the ordered set of neighbors of site t.*

(c) Homogeneity: $P(x_t | R_t)$ *does not depend on a particular site t.*

Figure 2 defines the relative sites and orders of neighbors of site *t*. A GRF and an MRF are equivalent [3] with respect to a specified neighborhood system.

t:-3	t:-2	t:+4
t:-1	t	t:+1
t:-4	t:+2	t:+3

2	1	2
1	t	1
2	1	2

Figure 2. The relative sites and orders of neighbors of site t.

A Gibbs random field is completely characterized by its energy function. In this paper, a commonly used MRF model whose energy function has the following form is introduced:

$$\mathrm{U(x)} = \sum_{t=1}^{MN} F(x_t) + \sum_{t=1}^{MN} \sum_{r=1}^{c} H(x_t, x_{t:+r}) \tag{2}$$

where $H(a, b) = H(b, a)$ and c depends on the size of the neighborhood. For example, c = 2, 4 for 1storder and 2nd-order neighborhoods, respectively. The generalized Ising MRF model (GIM) is defined below [3]

2.1.1 Generalized Ising Model (GIM)

Let $A = \{0, 1, ..., G-1\}$; the F and H functions of (2) in the generalized Ising model are defined as $F(x_t) = \alpha_{x_t}$ and $H(x_t, x_t{:}+r) = \theta_r I(x_t, x_t{:}+r)$, where $I(a, b) = -1$ if $a = b$ and $I(a, b) = 1$, otherwise.

Simple derivation gives the conditional density:

$$\mathrm{P}(x_t|R_t) = \exp\left[-\alpha_{x_t} - \sum_{r=-c}^{c} \theta_r I(x_t, x_{t:+r})\right]/W$$

$$\mathrm{W} = \sum_{s \in A} \exp[-\alpha_s - \sum_{r=-c}^{c} \theta_r I(s, x_{t:+r})] \tag{3}$$

An algorithm for simulating the generalized Ising model (GIM) is given below [3]. Two 128 × 128 synthesized image textures obtained based on GIM with the parameters $\theta = (1, 1, 1, -1)$ and $\theta = (2, 2, -1, -1)$ are shown in Figures 3(a) and 3(b), respectively.

Algorithm Generalized Ising Algorithm [GIM]

(1) For s=1 to MN, randomly assign a $g \in A$ to each x_s to give an initial image x.

(2) For s=1 to MN Do

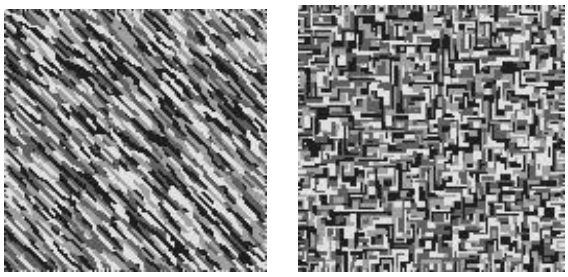
(a) Let $y_t = x_t$ for all $t \neq s$. Choose $g \in A$ at random and let $y_s = g$.

(b) Let $r = \min\{1, P(y)/P(x)\}$, where P is as defined in eq. (1).

(c) x ← y with probability r.

(3) Repeat step (2) until "*convergence*," is achieved, for example, in 50 iterations.

Each of the four parameters of the 2nd-order GIM model is restricted to be between -2 and 2 to avoid the phase-transition phenomenon [3]. In practice, this model assumes that a texture will consist of a small number of gray levels, for example, 8 or less. Each parameter determines a directionality; the larger the negative value of the parameter, the stronger the direction.



(a) (b)

Figure 3. Images with (a) θ=(1,1,1,-1), (b) θ=(2,2,-1,-1).

3. Data Encryption and Character Partition

Suppose that a message consists of characters whose ASCII code is from 0 to 255, for example, "h,i,d,e" is represented as "104,105,100,101". We adopt the concept of *a primitive root mod p=257* [11] as permutation on {0, 1, · · · , 255} to map (encrypt) the message "104,105,100,101" into "159,172,217,103" character by character such as $x \rightarrow y$, for example, 104 → 159, 105 → 172, 100 → 217, 101 → 103, by the following modular computations

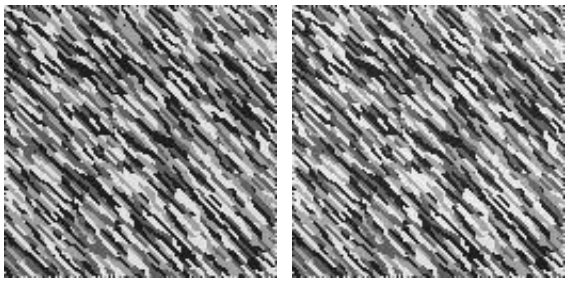
$$y + 1 \equiv g^{x+1} \ mod \ p = 257 \tag{4}$$

We then partition the 8-bit integer into 2 4-bit words to be embedded into two pixels, for example, 159 = 10011111 is splitted into 1001 and 1111. These two 4-bit words are sequentially embedded into the least significant bits of the corresponding pixels whose pixel values are designated as one of the four values {32, 96, 160, 224}.

4. Experimental Results

There are 3 major modules (programs), *embed1.c extract1.c, genising.c,* required to implement this system. The result of embedding a famous article derived from the commencement speech to Stanford University students in 2005 given by Steve Jobs, the late CEO of Apple Inc. are given as follows. The article is given as a plaintext file

steve.txt which consists of 6646 characters in 25 lines and 1094 words. The result of this article is embedded into a 128 × 128 cover-image *rain128.raw* to get a 128 × 128 stego-image *stego2.raw*. The cover-image *rain128.raw* and stego-image *stego2.raw* with the peak signal-to-noise ratio (PSNR) *30.08* shown below look visually quite the same.



(a) (b)

Figure 4. (a) Cover-image and (b) Stego-Image.

5. Conclusion and Discussion

This paper proposes a simple and secure steganographic system based on Number Theory [11] and MRF-Synthesized cover images. Three major programs written in C language are used for this system: (1) genising.c with user-specified parameters is used to generate cover images; (2) embed1.c is used for embedding messages into a cover-image by a user-specified primitive root (mod p=257) to get a stego-image, and (3) extract1.c is used for recovering the hidden message from a stego-image. An experiment for embedding the message of *steve.txt* containing 6646 ASCII characters Into a 128 × 128 texture image is illustrated. The cover-image and the corresponding stego-image has the PSNR value 30.08 and visually look the same. This practical system can not only be used for embedding plaintext files but also be applied to embedding other file formats such as word, excel, pdf, and etc.

6. Acknowledgments

This work is partially supported by Taiwanese grant NSC 101-2221-E-007-125-MY3.

REFERENCES

[1] Chan, C.K. and Cheng, L.M., *Hiding Data In Images By Simple LSB Substitution, Pattern Recognition*, vol. 37, 469-474, 2004.

[2] Chang, F.J., *A Steganographic Method Using MRF-Synthesized Textures as Cover Images*, M.S. Thesis, National Tsing Hua University, Hsinchu Taiwan, April 2011.

[3] Chen, C.C. and Chen, C.C., Texture Synthesis: *A Review and Experiments, Journal of Information Science and Engineering*, vol. 19, no. 2, 371-380, 2003.

[4] Chen, C.C., Kao, C.Y., Chang, C.F., Chu, H.T., and Chen, C.N., *Simple Software for Microarray Image Analysis*, IEEE Proceedings of Computer and Robot Vision, Pid163, Quebec City, Canada, 2006.

[5] Cox, I.J., Miller, M.L., Bloom, J.A., Fridrich, J., and Kalker, T., *Digital Watermarking and Steganography*, Morgan and Kaufmann, 2008.

[6] Johnson, N.F. and Jajodia, S., *Exploring Steganography: Seeing the Unseen*, IEEE Computer Magazine, vol. 31, no. 2, 26-34, 1998.

[7] Liu, C.L. and Liao, S.R., *High-Performance JPEG Steganography Using Complementary Embedding Strategy, Pattern Recognition*, vol. 41, 2945-2955, 2008.

[8] Ni, Z., Shi, Y.Q., Ansari, N., and Su, W., Reversible Data Hiding, *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 16, no. 3, 354-362, 2006.

[9] Provos, N. and Homeyman, P., *Hide and Seek: An Introduction to Steganography, IEEE Security & Provacy*, vol. 1, no. 3, 32-44, 2003.

[10] Pennebaker, W.B., Mitchell J., *JPEG Still Image Compression Standard, New York: Van Nostrand Reinhold*, 1993.

[11] Stinson, D.R., *Cryptography Theory and Practice*, 3rd ed., Chapman & Hall/CRC, 2006.

[12] Westfeld, A., *F5 − A Steganographic Algorithm: High Capacity Despite Better Steganalysis*, International Workshop on Information Hiding, Berlin, vol. 2137, 289-302, 2001.

[13] http://en.wikipedia.org/wiki/Steganography, last access on January 16, 2014.

[14] http://www.cs.nthu.edu.tw/~cchen/steve.txt, last access on January 16, 2014.

[15] Lai, W.J., A High-Capacity Steganographic System Using MRF-Based Texture Synthesis, M.S. Thesis, National Tsing Hua University, Hsinchu, Taiwan, March 2013.