2010-06-01

Information Hiding using Stochastic Diffusion for the Covert Transmission of Encrypted Images

Jonathan Blackledge
Technological University Dublin, jonathan.blackledge@tudublin.ie

Information Hiding using Stochastic Diffusion for the Covert Transmission of Encrypted Images

Jonathan Blackledge

SFI Stokes Professor - http://eleceng.dit.ie/blackledge
School of Electrical Engineering Systems
College of Engineering and the Built Environment
Dublin Institute of Technology

E-mail: jonathan.blackledge@dit.ie

Abstract — **A principal weakness of all encryption systems is that the output data can be 'seen' to be encrypted. In other words, encrypted data provides a 'flag' on the potential value of the information that has been encrypted. In this paper, we provide a novel approach to 'hiding' encrypted data in a digital image. We consider an approach in which a plaintext image is encrypted with a cipher using the processes of 'stochastic diffusion' and the output quantized into a 1-bit array generating a binary image ciphertext. This output is then 'embedded' in a host image which is undertaken either in the lowest 1-bit layer or multiple 1-bit layers. Decryption is accomplished by extracting the binary image from the host image and correlating the result with the original cipher. The approach has a variety of applications including: (i) covert transmission of encrypted images; (ii) authentication and self-authentication of e-documents that are assumed to be communicated over the Internet and are thereby vulnerable to attack (e.g. modification, editing, counterfeiting etc.). The paper includes an address from which interested readers can download a prototype system called *StegoCrypt* developed using the algorithms presented.**

Keywords — **Information Hiding, Covert Encryption, Steganography, Stochastic Diffusion.**

I Introduction

One of the principal weaknesses of all encryption systems is that the form of the output data (the ciphertext), if intercepted, alerts the intruder to the fact that the information being transmitted may be of some importance and that it is thereby worth attacking. It is therefore of significant value if a method can be found that allows data to be transmitted by embedding it in non-sensitive information after (or otherwise) it has been encrypted. This is known as *Steganography* which is concerned with developing methods of writing hidden messages in such a way that no one, apart from the intended recipient, knows of the existence of the message in contrast to cryptography in which the existence of the message itself is not disguised but the content is scrambled [1], [2]. Steganography provides a significant advantage over cryptography alone in that messages do not attract attention to themselves, to messengers, or to recipients. No matter how well plaintext is encrypted (i.e. how unbreakable it is), by default, a ciphertext will arouse suspicion and may in itself be incriminating, as in some countries encryption is illegal.

This paper presents a method of 'hiding' encrypted information in the form of a digital watermark in a colour digital image. In principle, any cipher can be used to do this providing it consists of decimal integer and/or floating point numbers that are, ideally, uniformly distributed. The scheme allows for the authentication and self-authentication of documents such as letters, certificates and other image based data and the applications to which the method can be applied are numerous. For example, the self-authentication of e-documents sent as attachments over the internet provides a unique

facility for many legal and financial transactions that have traditionally relied on paper based documents to secure authenticity. The method also provides a unique way of 'propagating' disinformation in the form of an encrypted document which contains hidden information.

II Steganography

The word 'Steganography' is of Greek origin and means 'covered', or 'hidden writing'. The conversion of a ciphertext to another plaintext form is called *Stegotext* conversion and is based on the use of *Covertext*. Some covertext must first be invented or acquired and the ciphertext mapped on to it in some way to produce the stegotext. This can involve the use of any attribute that is readily available, accessible and appropriate, transformed in such a way as to carry a hidden message. The basic principle is given below:

$$\begin{array}{ccccc} & & \text{Data} & \rightarrow & \text{Covertext} \\ & & & & \downarrow \\ \text{Plaintext} & \rightarrow & \text{Ciphertext} & \rightarrow & \text{Stegotext} \\ & & & & \downarrow \\ & & & & \text{Transmission} \end{array}$$

Note that this approach does not necessarily require the use of plaintext to ciphertext conversion as illustrated above and that plaintext can be converted into stegotext directly. With the wealth of data that is generated and transmitted in todays environment together with the wide variety of digital formats that are used, there is much greater potential for exploiting steganographic methods than were available before the development of the information and communications technologies now available. In other words, we now operate in a covertext rich environment and one can attempt to hide plaintext or ciphertext or both in a host of data types, including audio and video files and digital images, for example. There are some counter measures - *steganalysis* - that can be implemented in order to detect stegotext. However, steganalysis usually requires access to the covertext which is then compared with the stegotext to see if any modifications have been introduced. The approach discussed in this paper does not require *a priori* knowledge of the covertext and thereby eliminates this underlying vulnerability.

III Stochastic Diffusion

The relatively large amount of data contained in digital images makes them a useful 'medium' for undertaking steganography. Consequently digital images can be used to hide messages contained in other plaintext or encrypted images. In this paper, we consider the diffusion of plaintext in terms of the convolution of a plaintext array $p[i]$ with a cipher $n[i]$ to generate a ciphertext given by

$$c[i] = n[i] \otimes p[i] = \sum_j n[j-i]p[j]$$

where $\otimes$ denotes the convolution sum as defined above.

It is well known that for $x \in (-\infty, \infty)$, the solution to the diffusion equation

$$(D\partial_x^2 - \partial_t)c(x,t) = 0, \quad c(x,0) = p(x)$$

where D is the *Diffusivity* and $p(x)$ is the initial condition, is given by [3]

$$c(x,t) = g(x,t) \otimes_x p(x) = \int_{-\infty}^{\infty} g(y-x,t)p(y)dy$$

where $\otimes_x$ now represents the convolution integral (over independent variable x) as defined above and

$$g(x,t) = \sqrt{\frac{1}{4\pi Dt}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Here, the diffusion of $p(x)$ is determined by a convolution with a Gaussian function whose standard deviation is determined by the product of the Diffusivity D and the time t over which the diffusion process occurs. Stochastic diffusion is based on replacing the convolution kernel (i.e. the function g) for a stochastic function n so that

$$c(x,t) = n(x,t) \otimes_x p(x)$$

For any time t, the ciphertext is given by the convolution of the plaintext with the cipher. In this sense, any fixed value of time $t_i, i = 1, 2, 3...$ can be taken to represent the initial value used to generate $n(x)$, i.e. the key used to initiate some cryptographically secure random number generating algorithm.

The inverse problem associated with stochastic diffusion as defined above is not as simple as applying an XOR operation to a ciphertext that has been generated in binary space. In this case we are required to apply a suitable deconvolution process, i.e. to solve the problem: Given c and n, compute p. We can deconvolve by using the convolution theorem giving

$$p(x) = \mathcal{F}_1^{-1}\left[\frac{C(k)N^*(k)}{\mid N(k) \mid^2}\right]$$

where N is the Fourier transform of n, C is the Fourier transform of c, k is the spatial frequency and $\mathcal{F}_1^{-1}$ denotes the (one-dimensional) inverse Fourier transform, i.e. for a piecewise continuous function $f(x)$ with spectrum denoted by $F(k)$,

$$\mathcal{F}_1^{-1}[F(k)] = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(k)\exp(ikx)dx = f(x)$$

and

$$F(k) = \int_{-\infty}^{\infty} f(x) \exp(-ikx) dx$$

This approach requires regularization methods to be adopted in order to eliminate singularities associated with the case when $\mid N \mid^2 \rightarrow 0$ through application of a constrained deconvolution filter, for example [4], [5]. It is not appropriate for encryption because the inverse process is fundamentally ill-conditioned. Instead we consider a method which allows the inverse problem to be solved directly by correlation. To do this the cipher needs to be 'pre-conditioned'. We let

$$m(x) = \mathcal{F}_1^{-1}[M(k)]$$

where $\forall k$

$$M(k) = \begin{cases} \frac{N^*(k)}{|N(k)|^2}, & \mid N(k) \mid \neq 0; \\ N^*(k), & \mid N(k) \mid = 0. \end{cases}$$

The ciphertext is then given by

$$c(x) = m(x) \otimes_x p(x)$$

This result allows us to solve the inverse problem by correlating (denoted by $\odot_x$) c with n. This is based on the following analysis: Using the convolution theorem

$$m(x) \otimes_x p(x) \leftrightarrow M(k)P(k)$$

and, using the correlation theorem,

$$n(x) \odot_x c(x) \leftrightarrow N^*(k)C(k)$$

where $\leftrightarrow$ denotes transformation into Fourier space. Thus

$$N^*(k)C(k) = N^*(k)M(k)P(k)$$

$$= N^*(k)\frac{N^*(k)}{\mid N(k) \mid^2} = P(k)$$

so that

$$p(x) = n(x) \odot_x c(x)$$

The pre-conditioning of a cipher so that decryption can be undertaken using correlation provides a simple solution for utilizing the process of stochastic diffusion to encrypt data. In this paper, the process is applied in 'image space' to watermark a digital image.

IV Digital Image Watermarking

In 'image space', we consider the plaintext to be an image $p(x, y)$ of compact support $x \in [-X, X]$; $y \in [-Y, Y]$. Stochastic diffusion is then based on the following results:

Encryption

$$c(x, y) = m(x, y) \otimes_x \otimes_y p(x, y)$$

where

$$m(x, y) = \mathcal{F}_2^{-1}\left[M(k_x, k_y)\right]$$

and $\forall k_x, k_y$

$$M(k_x, k_y) = \begin{cases} \frac{N^*(k_x,k_y)}{|N(k_x,k_y)|^2}, & \mid N(k_x, k_y) \mid \neq 0; \\ N^*(k_x, k_y), & \mid N(k_x, k_y) \mid = 0. \end{cases}$$

Decryption

$$p(x, y) = n(x, y) \odot_x \odot_y c(x, y)$$

Here, k_x and k_y are the spatial frequencies and $\mathcal{F}_2^{-1}$ denotes the two-dimensional inverse Fourier transform. For digital image watermarking, we consider a discrete array $p_{ij}, i = 1, 2, ..., I; j = 1, 2, ..., J$ of size $I \times J$ and discrete versions of the operators involved, i.e. application of a discrete Fourier transform and discrete convolution and correlation sums.

For a host image denoted by $h(x, y)$, we consider a watermarking method based on the equation

$$c(x, y) = Rm(x, y) \otimes_x \otimes_y p(x, y) + h(x, y)$$

where

$$\|m(x, y) \otimes_x \otimes_y p(x, y)\|_\infty = 1$$

and

$$\|h(x, y)\|_\infty = 1$$

By normalising the terms in this way, the coefficient $0 \leq R \leq 1$ can be used to adjust the relative magnitudes of the terms such that the diffused image $m(x, y) \otimes_x \otimes_y p(x, y)$ becomes a perturbation of the 'host image' (covertext) $h(x, y)$. This provides us with a way of digital watermarking [6] one image with another, R being referred to as the 'watermarking ratio', a term that is equivalent, in this application, to the standard term 'Signal-to-Noise' or SNR as used in signal and image analysis. For colour images, the method can be applied by decomposing the image into its constituent Red, Green and Blue components. Stochastic diffusion is then applied to each component separately and the result combined to produce a colour composite image.

For applications in image watermarking, stochastic diffusion has two principal advantages:

- a stochastic field provides uniform diffusion;
- stochastic fields can be computed using random number generators that depend on a single initial value or seed (i.e. a private key).

a) Binary Image Watermarks

Watermarking a full grey level or colour image with another grey or colour image, respectively, using stochastic diffusion leads to two problems: (i) it can yield a degradation in the quality of the reconstruction especially when R is set to a low value which is required when the host image has regions that are homogeneous; (ii) the host image can be corrupted by the watermark leading to distortions that are visually apparent. Points (i) and (ii) lead to an optimisation problem with regard to the fidelity of the watermark and host images in respect of the value of the watermark ratio that can be applied limiting the type of host images that can be used and the fidelity of the 'decrypts'. However, if we consider the plaintext image $p(x, y)$ to be of binary form, then the output of stochastic diffusion can be binarized to give a binary ciphertext. The rationale for imposing this condition is based on considering a system in which a user is interested in covertly communicating documents such as confidential letters and certificates, for example.

If we consider a plaintext image $p(x, y)$ which is a binary array, then stochastic diffusion using a pre-conditioned cipher $0 \leq m(x, y) \leq 1$ consisting of an array of floating point numbers will generate a floating point output. The Shannon Information Entropy of any array $A(x_i, y_i)$ with Probability Mass Function (PMF) $p(z_i)$ is given by

$$I = -\sum_{i=1} p(z_i) \log_2 p(z_i)$$

The information entropy of a binary plaintext image (with PMF consisting of two components whose sum is 1) is therefore significantly less than the information entropy of the ciphertext image. In other words, for a binary plaintext and a non-binary cipher, the ciphertext is data redundant. This provides us with the opportunity of binarizing the ciphertext by applying a threshold T, i.e. if $c_b(x, y)$ is the binary ciphertext, then

$$c_b(x, y) = \begin{cases} 1, & c(x, y) > T \\ 0, & c(x, y) \leq T \end{cases} \qquad (1)$$

where $0 \leq c(x, y) \leq 1 \forall x, y$. A digital binary ciphertext image $c_b(x_i, y_j)$ where

$$c_b(x_i, y_i) = \begin{cases} 1, & \text{or} \\ 0, & \text{for any } x_i, y_j \end{cases}$$

can then be used to watermark an 8-bit host image $h(x, y), h \in [0, 255]$, for example, by replacing the lowest 1-bit layer with $c_b(x_i, x_j)$. To recover this information, the 1-bit layer is extracted from the image and the result correlated with the digital cipher $n(x_i, y_j)$. Note that the original floating point cipher n is required to recover the plaintext image and that the binary watermark can not therefore be attacked on an exhaustive XOR basis using trial binary ciphers. Thus, binarization of a stochastically diffused data field is entirely irreversible, i.e. equation (1) describes a 'one-way function'.

b) Statistical Analysis

The expected statistical distribution associated with stochastic diffusion is Gaussian. This can be shown if we consider a binary plaintext image $p_b(x, y)$ to be described by a sum of N delta functions where each delta function describes the location of a non-zero bit at coordinates (x_i, y_j). Thus if

$$p_b(x, y) = \sum_{i=1}^{N} \sum_{j=1}^{N} \delta(x - x_i)\delta(y - y_j)$$

then

$$c(x, y) = m(x, y) \otimes_x \otimes_y p(x, y)$$
$$= \sum_{i=1}^{N} \sum_{j=1}^{N} m(x - x_i, y - y_j).$$

Each function $m(x - x_i, y - y_j)$ is just $m(x, y)$ shifted by x_i, y_j and will thus be identically distributed. Hence, from the Central Limit Theorem

$$\Pr[c(x, y)] = \Pr\left[\sum_{i=1}^{N} \sum_{j=1}^{N} m(x - x_i, y - y_j)\right] =$$

$$\prod_{i=1}^{N}{}^{\otimes} \Pr[m(x, y)] \sim \text{Gaussian}, \quad N \to \infty$$

where Pr denotes the Probability Density Function. We can thus expect $\Pr[c(x, y)]$ to be normally distributed and for $m(x, y) \in [0, 1] \forall x, y$ the mode of the distribution will be of the order of 0.5. This result provides a value for the ideal threshold T in equation (1) which for $0 \leq c(x, y) \leq 1$ is 0.5. Note that if $n(x, y)$ is uniformly distributed and thereby represents δ-uncorrelated noise, then both the complex spectrum N^* and power spectrum $\mid N \mid^2$ will also be δ-uncorrelated and since

$$m(x, y) = \mathcal{F}_2^{-1}\left[\frac{N^*(k_x, k_y)}{\mid N(k_x, k_y) \mid^2}\right]$$

$\Pr[m(x, y)]$ will be uniformly distributed. Also note that the application of a threshold which is given by the mode of the Gaussian distribution, guarantees that there is no statistical bias associated with any bit in the binary output, at least, on a theoretical basis. On a practical basis, this needs to be computed directly by calculating the mode from the histogram of the cipher and that bit equalization can not be guaranteed as it will depend on: (i) the size of the images used; (ii) the number of bins used to compute the histogram.

c) Principal Algorithms

The principal algorithms associated with the application of stochastic diffusion discussed in the previous sections are as follows:

Algorithm I: Encryption and Watermarking Algorithm

Step 1: Read the binary plaintext image from a file and compute the size $I \times J$ of the image.

Step 2: Compute a cipher of size $I \times J$ using a private key and pre-condition the result.

Step 3: Convolve the binary plaintext image with the pre-conditioned cipher and normalise the output.

Step 4: Binarize the output obtained in Step 3 using a threshold based on computing the mode of the Gaussian distributed ciphertext.

Step 5: Insert the binary output obtained in Step 4 into the lowest 1-bit layer of the host image and write the result to a file.

The following points should be noted:

(i) The host image is taken to be an 8-bit or higher grey level image which should be of the same size as the plaintext image or else resized accordingly. However, in resizing the host image, its proportions should be the same so that the stegotext image does not appear to be a distorted version of the covertext image. For this purpose, a library of host images should be developed whose dimensions are set according to a predetermined application where the size of the expected plaintext images are known.

(ii) Pre-conditioning the cipher and the convolution processes are undertaken using a Discrete Fourier Transform (DFT).

(iii) The output given in Step 3 will include negative floating point numbers upon taking the real component of a complex array. The array must be rectified by adding the largest negative value in the output array to the same array before normalisation.

(iv) For colour host images, the binary ciphertext can be inserted in to one or all of the RGB components. This provides the facility for watermarking the host image with three binary ciphertexts (obtained from three separate binary documents, for example) into a full colour image. In each case, a different cipher can be used.

(v) The binary plaintext image should have homogeneous margins in order to minimise the effects of ringing due to 'edge-effects' when processing the data in the spectral domain.

(vi) The cipher can be one of any key-dependent random number generating algorithms that are taken to be cryptographically secure.

Algorithm II: Decryption Algorithm

Step 1: Read the watermarked image from a file and extract the lowest 1-bit layer from the image.

Step 2: Regenerate the (non-preconditioned) cipher using the same cipher used in Algorithm I.

Step 3: Correlate the cipher with the input obtained in Step 1 and normalise the result.

Step 4: Quantize and format the output from Step 3 and write to a file.

The following points should be noted:

(i) For colour images, the data is decomposed into each RGB component and each 1-bit layer is extracted and correlated with the appropriate cipher, i.e. the same cipher or three ciphers relating to three private keys respectively.

(ii) The output obtained in Step 3 has a low dynamic range and therefore requires to be quantized into an 8-bit image based on floating point numbers within the range max(array)-min(array).

d) 2-bit Randomization

A principal weakness in the algorithms proposed is that the binary watermark is inserted into the lowest 1-bit layer of the host image. The ciphertext can therefore easily be removed and cryptanalysed. A simple solution to this problem is to randomize the watermark over the lowest 2-bit layers of the host image. Thus, if $c_b(x_i, x_j)$ is the binary watermark of size $X \times Y$ and $0 \leq r(x_i, y_j) \leq 1$ is an array of uniformly distributed floating point numbers (also of size $X \times Y$), we apply the quantization

$$\forall x_i, y_j \in [X \times Y]$$

$$R(x_i, y_j) = \begin{cases} 0, & 0 \leq r(x_i, y_j) < 0.333; \\ 1, & 0.333 \leq r(x_i, y_j) < 0.666; \\ 2, & 0.666 \leq r(x_i, y_j) < 1; \end{cases}$$

The watermark is then 2-bit randomized by simple array addition to give

$$C(x_i, x_j) = c_b(x_i, x_j) + R(x_i, y_j)$$

The 2-bit array C then replaces the lowest 2-bit layer of the host image. The 1-bit watermark is recovered using array subtraction, i.e.

$$c_b(x_i, x_j) = C(x_i, x_j) - R(x_i, y_j)$$

which requires that r can be reproduced using a cipher generating algorithm with a known 'key'.

V StegoCrypt

StegoCrypt is a prototype software system engineered using MATLAB to examine the applications to which stochastic diffusion can be used as given in Algorithm I and Algorithm II. It has been designed with a simple Graphical User Interface as shown in Figure 1 whose use is summarised in the following table:

Encryption Mode	**Decryption Mode**
Inputs: Plaintext image, Covertext image, Private Key (PIN)	*Inputs:* Stegotext image, Private key (PIN)
Output: Stegotext image	*Output:* Decrypted watermark
Operation: Encrypt by clicking on button E	*Operation:* Decrypt by clicking on button D

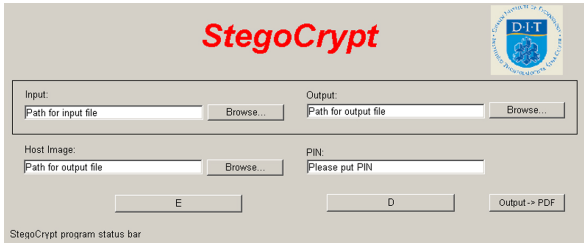

Fig. 1: Graphical User Interface for *StegoCrypt*.

The PIN (Personal Identity Number) can be a numerical string with upto 16 elements. A demo version of *StegoCrypt* is available from `http://eleceng.dit.ie/arg/downloads/Stegocrypt.zip`

Installation is initiated through setup.exe from the root folder in which the downloaded application has been placed (after unzipping the downloaded file *Stegocrypt.zip*) and following the instructions on screen. Figure 2 gives an example of the I/O produced by *StegoCrypt*

VI Conclusion

The principal aim of the approach described in this paper is to encrypt an image and transform the ciphertext into a binary array which is then used to watermark a host image. This provides a general method for hiding encrypted information in 'image-space'. The use of the internet to transfer documents as image attachments has and continues to grow rapidly and it is for this 'market' that the approach reported in this paper has been developed. Inserting a binary watermark into a host image obtained by binarizing a ciphertext of a document provides a cryptographically secure

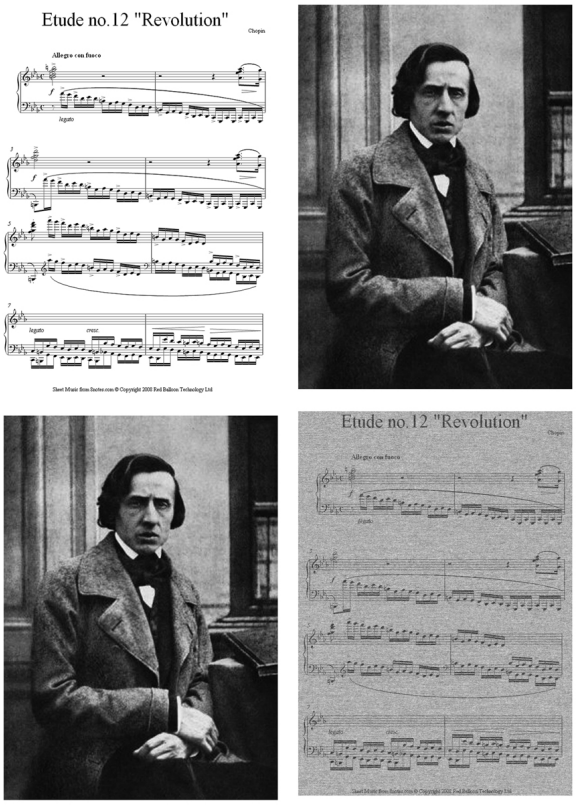

Fig. 2: Plaintext image (top-left), covertext image (top-right), watermarked image - Stegotext (bottom-left) and associated decrypt (bottom-right).

solution. Although the watermark can be easily removed from the covertext image (unless 2-bit randomization is implemented) it can not be decrypted without the recipient having access to the correct cryptographically secure cipher.

References

[1] S. Katzenbeisser and F. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking", *Artech House*, 2000.

[2] N. F. Johnson, Z. Duric and S. Jajodia, "Information Hiding: Steganography and Watermarking - Attacks and Countermeasures", *Kluwer Academic Publishers*, 2001.

[3] G. Evans, J. M. Blackledge and P. Yardley, "Analytical Methods for Partial Differential Equations", *Springer Undergraduate Mathematics Series, Springer-Verlag*, 2000.

[4] M. Bertero and B. Boccacci, "Introduction to Inverse Problems in Imaging", *Institute of Physics Publishing*, 1998.

[5] J. M. Blackledge, "Digital Signal Processing, Second Edition", *Horwood Publishing*, 2006.

[6] I. J. Cox, M. Miller, and J. Bloom, "Digital Watermarking", *Morgan Kaufmann*, 2002.