

Developments in Steganography

Published in the Proceedings of the Third
International Information Hiding Workshop,
Dresden, Germany, September 29-October 1, 1999.
Springer-Verlag Lecture Notes in Computer Science, 1768)
Joshua R. Smith and Chris Dodge

Escher Labs
101 Main Street
Cambridge, MA 02139
USA
jrs@escher-labs.com

Abstract. This paper presents two main results. The first is a new approach to steganography in which data is encoded in correlations among the pixels in an image. Almost all previous steganographic methods encode data in correlations between the pixels and a known external reference signal. This method hints at the existence of public key watermarking techniques, which will be defined.

The other result is a method for greatly increasing the capacity of a printed steganographic channel. Because it is specific to printed images, this method is useful for steganographic problems such as stealth barcoding, but not for digital watermarking. The two results are complementary in that higher noise levels are encountered in the intra-image correlation encoding method, but the second method works by eliminating image-induced noise.

1 Introduction

The first half of this paper introduces a new approach to steganography in which data is encoded in correlations among the pixels in an image, rather than in correlations between image pixels and a known external reference signal. This method hints at the existence of *public key watermarking schemes*. We will define *weak* and *strong* public key watermarking, show that the new method is an example of a weak public key watermarking system, and conjecture that strong public key watermarking systems exist.

The second half of the paper presents *quadcluster encoding*, method of steganographically encoding data in printed form at higher densities than had been possible using the communications theory picture in which the cover image is treated as noise, introduced in [3]. The new method makes use of the fact that although the cover image is unknown to the receiver, it is known perfectly to the encoder; thus treating it as ordinary channel noise is overly pessimistic.

2 Background: Modulation Schemes and Inter-image correlation

In a traditional binary phase shift modulation scheme, each bit b_i is represented by some basis function ϕ_i multiplied by either positive or negative one, depending on the value of the bit. The index i is being used to label the bits, and their associated carriers. The modulated message $S(x, y)$ is added pixel-wise to the cover image $N(x, y)$ to create the stego-image $D(x, y) = S(x, y) + N(x, y)$. The modulated signal is given by

$$S(x, y) = \sum_i b_i \phi_i(x, y)$$

A bit can be recovered by demodulation

$$\langle D, \phi_i \rangle = \langle S + N, \phi_i \rangle = \langle b_i \phi_i + N, \phi_i \rangle = b_i \langle \phi_i, \phi_i \rangle + \langle N, \phi_i \rangle \approx b_i$$

To the extent it is possible, the basis functions should be uncorrelated with (orthogonal to) the cover image N .

$$\langle \phi_i, N \rangle = \frac{1}{n} \sum_{x,y} \phi_i(x, y) N(x, y) \approx 0$$

where n is the total number of pixels summed over.

As pointed out in [2], the cover image N typically has a large non-zero DC component, since by convention luminosity values are taken to be positive, and thus the carrier ϕ should have zero DC component. If the carrier has a non-zero DC component, there will be a large noise contribution from the inner product of the cover image's DC component with the carrier's DC component.

2.1 Direct-Sequence Spread Spectrum

In the implementation of direct sequence spread spectrum in [3], the modulation function consists of a constant, integral-valued gain factor G multiplied by a pseudo-random block ϕ_i of $+1$ and -1 values. Each block ϕ_i has a distinct location in the (x, y) plane. The use of blocks is not a necessity. In some applications, it might be desirable to interleave the carriers so that each one is spread over the entire area of the image.

The embedded data is recovered by demodulating with the original modulating function. A TRUE ($+1$) bit appears as a positive correlation value; a FALSE (-1) bit is indicated by a negative correlation value. Once the carrier phase has been recovered, we project the stego-image onto each basis vector ϕ_i :

$$o_i = \langle D, \phi_i \rangle = \frac{1}{n} \sum_{x,y} D(x, y) \phi_i(x, y)$$

and then threshold the o_i values to recover the bits b_i .

In this scheme, as in many steganographic schemes, information is encoded in correlations between the pixel values and an external reference vector ϕ that must be made available to any decoder.

3 Intra-image correlation

3.1 Motivation: Public Key Watermarking

We will define the term *public key watermarking* to mean a watermarking scheme with the property that the knowledge required to read (or verify) the hidden watermark data provides no additional help in stripping the watermark. This term should not be confused with public key steganography, which Anderson defined to mean the problem of communicating information through a steganographic channel using an asymmetric cryptosystem to keep the contents of the hidden data secret. A public key watermark is not supposed to be secret—it should be readable by anyone in possession of the public key—but only someone with the secret key should be able to remove it and recover a clean, unmarked original.

This is not to say that a public key watermark cannot be jammed: a large amount of noise can always be added blindly, for example. Nor is it to say that access to an oracle that can read watermark bits would not help: then modifications to the image can be tested until the watermark is unreadable.

Definition 1 *A strong public key watermarking scheme has the property that performing the decoding algorithm oneself, using the public key, confers no advantage (or vanishingly small advantage) in stripping the watermark above that provided by access to a watermark-reading oracle or server. With knowledge of the private key, the watermark can be stripped and the original source image recovered.*

Definition 2 *A weak public key watermarking scheme has the property that performing the decoding algorithm oneself, using the public key, does not confer the ability to exactly strip the watermark and recover a clean copy of the original. With knowledge of the private key, the watermark can be stripped and the original source image recovered.*

In either the strong or the weak form, the decoder must in some (strong or weak) sense not be able to determine “where” the data is hidden. A spread spectrum demodulation algorithm that was capable of extracting modulated bits given received data and an encrypted version of the carrier, without ever decrypting the carrier, would probably qualify as a strong public key watermarking system.

We conjecture that strong public key watermarking schemes exist, though producing them will require a novel combination of cryptography and signal processing. The first part of this paper presents a weak public key watermarking scheme.

3.2 Intra-image methods

The technique introduced in this section encodes information in correlations *among* the pixels within an image, rather than in correlations between an image and an external reference. A consequence of this approach is that the ability to

read the message does not confer the ability to exactly strip out the message. Thus, the method is a weak public key watermarking system.

The only previous investigation of intra-image correlation we are aware of is the “texture-block coding” scheme described in [1]. In this scheme, a textured region is identified by a human operator, copied, and then pasted into another area of the image, replacing whatever pixels were originally at the destination location. The shape of the region can be tailored to represent information, for example, the characters “MIT.” The main drawback of this scheme is the requirement to manually identify regions of corresponding texture. For the scheme to be imperceptible, the source and destination regions have to be similar-looking regions of high texture. This means that the information is not stored in a pre-defined set of pixels. Thus an additional drawback is that decoding requires calculating the full 2d autocorrelation function of the image, in order to discover where the information is encoded. The information cannot be placed in a fixed location because its location depends on the human user’s choice.

The method presented in this section is automatic on encode and decode, and does not require a full autocorrelation calculation to perform the extraction. The method is as follows. Break the pixels of the cover image into two disjoint subsets of equal size. Then we can treat the pixels in the first subset as a vector D_1 , and the pixels from the second subset as vector D_2 . Information can be encoded in correlations between these two portions of the image, rather than between the image and an external carrier. A TRUE bit will be represented by positive correlations between the sub-regions, and FALSE will be represented by negative correlations (anti-correlation). Consider two sub-regions of an image that are being used to encode a single bit b (either $+1$ or -1). We can write the first sub-region as $D_1 = N_1 + \chi$ and the second as $D_2 = N_2 + b\chi$, where N_1 and N_2 are portions of the original unmodified “noise” image, b is the bit being encoded ($+1$ or -1) and χ is a random sequence that is generated in the course of encoding, but then may be thrown away. Like classical correlation-based steganography methods, the decoder does not need to know the original image N . The novel feature of this scheme is that the decoder does not need to know χ , either. The decoder does need to know, for each D_1 pixel, which D_2 pixel is associated with it.

The naive decoding algorithm would take the inner product of $\langle D_1, D_2 \rangle$. Because of the large DC components of N_1 and N_2 , however, the signal would be swamped. To see this, we can write an expansion of N_1 into its DC component plus all other terms, $N_1 = N_1^0 + N_1'$, where N_1^0 is the DC component. The naive inner product is

$$\langle D_1, D_2 \rangle = \langle N_1^0 + N_1' + \chi, N_2^0 + N_2' + b\chi \rangle$$

Since images are by convention positive, both images will have large DC components N_1^0 and N_2^0 , which will make a large contribution to this inner product. Subtracting off the DC components before taking the inner product eliminates the large contribution that would otherwise arise. Thus a much better demodulation method is

$$\langle D_1 - \langle D_1 \rangle, D_2 - \langle D_2 \rangle \rangle$$

where $\langle D_1 \rangle$ denotes the mean of D_1 , taken bit-blockwise.

3.3 Geometric interpretation

Classical schemes add a vectorial component to the image and represent information in the angle (typically 0 or 180 degrees) between this component and a known external reference vector.¹ The method presented here represents information in the relative angle between two vectors that are parts of the image; the absolute angle of these vectors may be completely unknown to the receiver.

Because data extraction in traditional methods is accomplished by measuring the angle to a reference vector that must be specified to the receiver, the knowledge required to optimally read data necessarily confers the ability to exactly strip out that data. In order to extract data using the method presented here, the receiver need know nothing about the absolute angle (in Hamming space) at which the data is stored.

3.4 A weak public-key watermarking scheme

Successfully decoding a bit reveals, for the blocks or sets of correlated pixels, whether the correlation is positive or negative. But for any pair of pixels, it does not reveal whether the positive correlation was due to a $(+1, +1)$ change, or a $(-1, -1)$ change. Similarly, negative correlation could be caused by either $(+1, -1)$ or $(-1, +1)$. (An additional source of confusion arises from the image noise.) This information (whether the contribution made by a pair of pixels to the overall positive correlation is due to a $(+1, +1)$ change or a $(-1, -1)$ change), which constitutes the absolute angle at which the data is encoded, is known only by the encoder, not the decoder. A decoder who came into possession of this information could exactly strip out the watermark, and recover the original image.

The absolute angle at which the data is encoded can be viewed as a private key that can be used to strip out the hidden data (that is χ is a private key). The public key needed to read the data is the sequence of paired pixels.

3.5 Attacks

Although the receiver does not *need* knowledge of χ to read the data, this is unfortunately not equivalent to the statement that the receiver *cannot acquire* some knowledge of where the information is stored. Knowledge that a pair of pixels is probably positively correlated, plus knowledge of whether each pixel is greater or less than the mean value of its block, yields some information about χ . An estimate of χ , even if it is not exactly correct, can be used to reduce the power of the watermark.

¹ This is because the inner product $\langle a, b \rangle = |a||b| \cos(\theta)$. If a and b are normalized so that $|a| = |b| = 1$, then the inner product returns the cosine of the angle between a and b .

There are two attacks on the intra-image encoding method that will reliably eliminate the watermark, though at the cost of further damage to the image. The first attack is to bring the correlation of each of the blocks toward zero. If a pair of blocks is positively correlated, then add negative correlation until the correlation score is close to zero. Flip a coin to determine whether to stop on the positive side of zero, or the negative side. (Replacing all positive correlations by negative correlations would have the effect of inverting, but not removing, the watermark).

The second attack is simply to blindly encode some other message. This attack highlights the fact that this scheme is not as asymmetric as one would like. Though the ability to read does not confer the ability to recover the unmarked image, it does confer the ability to overwrite the original watermark.

3.6 Capacity of Intra-Image correlation scheme

For purposes of analysis, this scheme can be mapped onto a classical inter-image correlation method. The channel capacity formula is

$$C = W \log_2\left(1 + \frac{S}{N}\right)$$

The demodulation method is $\langle b\chi + N_1 - \langle N_1 \rangle, \chi + N_2 - \langle N_2 \rangle \rangle = b + \langle N_1, \chi \rangle + b \langle \chi, N_2 \rangle + \langle N_1 - \langle N_1 \rangle, N_2 - \langle N_2 \rangle \rangle$, as compared with $\langle b\phi + N_1, \phi \rangle = b + \langle \phi, N_1 \rangle$ for classical schemes. Assuming that N_1 and N_2 are uncorrelated with ϕ and χ , then $\langle N_1, \chi \rangle = \langle \chi, N_2 \rangle = \langle \phi, N_1 \rangle$. Thus the noise power for the traditional algorithm is $\langle \phi, N_1 \rangle$, and the noise power for the intra-image method is $2 \langle \phi, N_1 \rangle + \langle N_1 - \langle N_1 \rangle, N_2 - \langle N_2 \rangle \rangle$. Clearly the noise power is significantly higher for the intra-image method.

3.7 Implementation

We have used the intra-image correlation method to encode the Gatlin image used in [3] with 100 bits of information.² A gain of 14 percent of the total dynamic range (changes of ± 9 , with the original image ranging from 0 to 64) was required to recover all 100 bits accurately. As expected, the performance is worse than [3], which hid 100 bits in the same image using lower gains. The level of gain required is clearly unacceptable from a perceptibility standpoint. However, the quadcluster noise reduction scheme in the second half of the paper could be combined with the intra-image encoding scheme to improve performance.

4 Quadcluster encoding

A shortcoming of the intra-image encoding method is that it effectively has less signal-to-noise to work with, because (to translate into traditional terminology)

² Specifically, we are using the first 320 x 320 pixels of the Gatlin image, which is available in Matlab via the “load gatlin” command.



Fig. 1. 320 by 320 image encoded with 100 bits using intra-image correlation encoding. As expected, this method requires higher gain to achieve the same level of capacity achieved by the direct sequence scheme in [3].

the demodulating carrier has been corrupted with noise. The quadcluster encoding method allows the effect of cover image noise to be eliminated completely in a printed context. This noise reduction would render the intra-image encoding technique practical (in a printed context).

The cover image is often taken to be noise that limits the performance of the steganographic communication channel, as in the scheme and analysis presented in [3]. Although the cover image is not known in advance by the receiver, the cover image is not strictly speaking noise. For example, the “noise” due to the cover image will be the same each time the image is scanned, unlike the true noise that arises from the circuitry in the scanner, which will be different each time the image is scanned. Furthermore, the cover image is known in advance to the encoder, unlike true channel noise. Because the cover image noise is not truly noise, it turns out not to limit the communications rate. We will make use of this insight, as well as the fact that the cover image does not occupy all the spatial bandwidth that is accessible to modern printers and scanners.

The quadcluster method can be explained as follows. Up-sample the cover image by a factor of 2 in each dimension, so that each original pixel is mapped to a cluster of 4 identical pixels. The up-sampled image now contains 4 times more pixels than the original, since each original pixel has been copied 4 times. We will use the term *sub-pixels* to refer to the 4 identical pixels that were copied from a particular original pixel. Now if we encode data using any modulation-based information hiding scheme, such as the ones described in [3], but leave one sub-pixel in each quadcluster unchanged, we can achieve complete immunity to the effects of “noise” due to the cover image. Without loss of generality, suppose that the unmodified pixel is the top left member of the cluster. To demodulate, we subtract the value of the unmodified top left pixel from the other pixel values in the same cluster, and then perform the usual demodulation

operation of projecting onto the basis function that was used to encode the information. We will describe better implementations of this basic idea later.

Current consumer ink jet printers have resolution specifications of 1440 dots per inch and higher, and photos meant for display on personal computer screens typically need only 72 to 100 dots per inch resolution to display them at reasonable size, so the up-sampled and encoded image can typically be printed at twice the original resolution (that is, exactly the same size as the original) by taking advantage of this unused resolution.

In practice, leaving one sub-pixel unmodified is not the best implementation. In our implementation described below, we generated one carrier “chip” value (+G or -G) for each pixel, not for each sub-pixel. The change made to each sub-pixel is determined by the product of the sub-carrier, the carrier, and the bit value. The sub-carrier values are chosen to sum to zero. This way, the original pixel value can be recovered by averaging the four associated sub-pixels. In our implementation, a fixed sub-carrier pattern of +1,-1,+1,-1 was used (with the labeling starting in the top left corner of the quadcluster and moving around clockwise). A variety of other sub-carrier sequences could be chosen, as long as the constraint that the sub-carrier values in a cluster sum to zero is obeyed. Each pixel can be upsampled by factors other than 2 in each dimension, for example 3x3 for nine sub-pixels, or 4x4 for 16 sub-pixels. In the context of printers with different horizontal and vertical resolution specifications, 8x4 might be a sensible upsampling. In general, the requirement is that it must be possible to estimate the original source pixel value from the sub-pixels.

4.1 Registration

It is a commonplace of the information hiding literature that spread spectrum techniques suffer from a need for precise registration between the encoded image and the demodulating carrier. The quad-cluster method is even more sensitive to misregistration than ordinary spread spectrum methods, and shows that ordinary spread spectrum methods actually do have some robustness to misregistration, when compared with the worst case.

Suppose that an image encoded with the spread spectrum method described in [3] has been placed on the scanner with a slight skew. Consider a 32x32 pixel block at the top left corner of the image, and suppose this block represents a single bit. Suppose further that because of the skew, the carrier being used to demodulate is registered correctly for the top left 16x16 quarter of this block, and misregistered in the rest of the block. A signal strength could be chosen such that this bit would be correctly demodulated, despite the fact that three quarters of the pixels are misregistered. Because this algorithm makes no assumptions about the cover image, treating it as noise, misregistration causes the signal power from the misregistered region to drop to zero, but has on average no effect on the noise power from this region. Since the contents of the cover image are treated as noise, there is a significant noise contribution from the cover image regardless of whether it is properly registered. Misregistered noise is no more harmful than “properly registered” noise.

In the quad-cluster method, misregistration causes a significant increase in cover image noise, as well as a decrease in signal. When the carrier is properly registered, there is zero contribution from cover image noise. When the carrier is “fully misregistered” (meaning every pixel in a carrier block misregistered), the cover image contributes potentially just as much noise as in the ordinary spread spectrum technique. Thus the quad-cluster method is much more sensitive to registration, as measured by the fall off of SNR as a function of the number of misregistered pixels.

4.2 Implementation

The obvious question given the discussion in the previous section is whether it is possible in practice to register the image and the carrier precisely enough to get the benefits of the quad-cluster method. Perhaps the increased noise due to misregistration offsets the noise savings offered by the method.

The answer to the question of whether precise enough registration is possible is a resounding yes. In the 320x320 pixel Gatlin image, shown in figure 2, in which 100 bits were hidden in [3] “on screen” (not printed and scanned) and with perfect registration, we can now hide 1024 bits and extract them with 100 percent accuracy after printing and scanning.

First the dynamic range of the original 6 bit gray scale image was expanded to fit into a standard 8 bit range of brightness values. The original was upsampled from 320 x 320 pixels to 640 x 640 pixels using nearest-neighbor interpolation - i.e. each source pixel was duplicated into a 2x2 block. This image’s dynamic range was then slightly compressed and its bias was slightly raised in order to avoid non-linear clipping distortion caused by pixel saturation. 1024 data bits were encoded with the quad-cluster modulation method at a gain setting of 7.8 percent (changes of +-20 out of 255) using a carrier block size of 10 x 10.

The test image was printed on an HP895C ink jet printer and on an Epson Color Stylus 740 ink jet at maximum nominal resolution (600 dpi and 1440dpi, respectively) on plain copier paper. The final printed size of the image was 4.4” x 4.4”. The scanning was performed on a Hewlett Packard ScanJet4c with a scaling setting of 500 percent, yielding a color-scanned image of approximately 1675 pixels x 1675 pixels. The scanned image was low-pass filtered with a 2D Gaussian kernel and subsequently downsampled, using nearest-neighbor interpolation, to a 640 x 640 image.

This post-processed scanned image was block-aligned and decoded by an algorithm implemented in C. Although the original test image was monochrome, we used a color scanning process to “oversample” the image, thereby smoothing out some of the noise introduced by the scanning circuitry. The red, green, and blue color planes were separately aligned in order to account for non-colinearity of the red, green, and blue detectors in the scanner.

Using this procedure, a decoding accuracy of 100 percent was achieved: all 1024 bits were decoded correctly. Presumably, even better overall performance could be achieved by going to higher data densities and employing an error correcting code. From the perspective of decreasing perceptibility, a gain of 9

(rather than 20) resulted in 98 percent bit decoding accuracy, and rendered the hidden data effectively invisible. The gain of 20 was chosen partly to exaggerate the changes so that they would become visible.



Fig. 2. Unmarked Gatlin image.



Fig. 3. Gatlin image with 1024 bits encoded using quadcluster encoding at a gain level of 7.8 percent ($\pm 20/255$), 10x10 block size. The linear resolution of this image is greater than the original by a factor of 2.

5 Conclusion

The intra-image encoding method is first example of a weak public key watermarking scheme. The ability to read the message does not confer the ability to produce a “clean” copy. As a steganographic method, it suffers from poor performance, due to its high susceptibility to image noise. The quad-cluster method utilizes the advance knowledge of cover image “noise” available to the encoder, plus the high spatial frequencies accessible to modern printers and scanners, to effectively cancel the effects of image noise. The quad-cluster method is certainly very useful by itself for imperceptibly encoding large amounts of data in images. It also should render the intra-image encoding technique usable as a steganographic technique (i.e. a method for transmitting large amounts of data, not as a watermarking technique), by reducing the image noise levels encountered by the intra-image encoding scheme. The application of the intra-image encoding technique for weak public key watermarking in a printed context (in which one might imagine gaining the benefit of the quad-cluster method) is less compelling, since once the image is printed, there is no longer any possibility of recovering a clean original, regardless of what encoding scheme is used.³

In introducing the quad-cluster modulation scheme, we have pointed out that the level of channel noise is substantially different for the transmitter and the receiver. For the transmitter, the cover image is not strictly speaking noise, because the transmitter has exact knowledge of the cover image; for the receiver, it is more accurate to call the cover image noise. Quad-cluster encoding makes use of the higher spatial frequencies accessible to modern printers, but it also makes use of the steganographic channel’s “asymmetric noise levels.” One interesting open question is how the asymmetric noise levels alter the usual channel capacity calculation. Another open problem that arises from the second half of this paper is finding other modulation schemes that make use of the noise level asymmetry of steganographic channels. Finally, finding examples of strong public key watermarking schemes, or proving the non-existence of such schemes, is the most significant open problem posed by this paper.

References

1. W. Bender, D. Gruhl, and N. Morimoto. Techniques for data hiding. *IBM Systems Journal*, 35(3 and 4), 1996.
2. J.P. Linnartz, T. Kalker, and G. Depovere. Modelling the false alarm and missed detection rate for electronic watermarks. In *Proceedings of the Second International Information Hiding Workshop*, Portland, Oregon, 1998.

³ It is natural to wonder whether embedding data with a standard, symmetric watermarking scheme and then printing the encoded picture qualifies trivially as a weak public key watermarking scheme. It does not, because although the ability to read the message does not confer the ability to exactly strip the message, there is no private key that would enable recovery of the original, unwatermarked image.

3. J. R. Smith and B. O. Comiskey. Modulation and information hiding in images. In *Proceedings of the First International Information Hiding Workshop*, Isaac Newton Institute, Cambridge, U.K., 1996.