Methods for data hiding

Jiri Fridrich

Center for Intelligent Systems &
Department of Systems Science and
Industrial Engineering
SUNY Binghamton
Binghamton, NY 13902-6000
fridrich@binghamton.edu
http://ssie.binghamton.edu/~jirif/

Abstract: *An overview of current steganographic methods for secret message hiding and digital watermarking is presented. A general theory of message hiding is used to put various techniques under a common umbrella. By interpreting watermarking as a key-dependent pattern overlaying, a new, robust watermarking scheme is proposed. The method is based on overlaying a pattern with its power concentrated mostly in low frequencies. The pattern is created using a pseudo-random number generator and a cellular automaton with voting rules. The robustness of the method with respect to filtering, JPEG compression, cropping, noise adding, and collusion is studied. The method also overcomes a possible weakness of the method of Cox et al. [Cox]. As part of the future directions, we propose a robust watermarking method which does not require the original image for watermark extraction.*

Keywords: *Steganography, cryptography, digital watermark*

Introduction

The techniques for secret hiding of messages in an otherwise innocent looking carrier message belong to the field of steganography. The purpose of steganography is to conceal the very presence of secret information. Steganography is not a substitute of cryptography. To the contrary, these two fields can tremendously benefit from each other if properly combined. The steganographic algorithm should depend on a secret key so that without the key, it should be impossible not only to extract the hidden message but also to prove its very existence. For obvious security reasons, the key should never be sent with the carrier and should be carrier independent. This is important because if an adversary gets hold of the hiding algorithm, the whole communication channel would become insecure. To make the communication more efficient, the secret information should be compressed before it is hidden in the carrier. This is important because in this way we minimize the amount of information that is to be sent. Also, it is easier to hide random looking message into the carrier than to hide a message with patterns and regularities. Encrypting the compressed message before hiding is recommended and provides a double protection.

The steganographer's job is to make the secretly hidden information difficult to detect given the complete knowledge of the algorithm used to embed the information except some secret key[1]. This so called Kerckhoff's principle is the golden rule of cryptography and is often accepted for steganography as well. In steganography, however, the principle becomes somewhat blurry because the adversary's ability to detect subtle changes in digital data depends on what she knows about the original carrier and its statistical properties. Simply postulating that the eavesdropper (Eve) knows everything there is to know about the carrier is overly optimistic, restrictive, and unachievable in practice. Eve may know for example the exact type of the scanner used by Alice and Bob to exchange messages hidden in digital images. Eve can buy the same brand of scanner and build a detailed model of the noise. However, if Eve builds a really detailed model of her scanner noise, she may overdo her job and actually detect differences between the two pieces of hardware rather than secret messages. On the other hand, if the source of images Alice and Bob use is not known to Eve, she has to learn the noise by intercepting the communication channel and collect enough information to build a model. However, this task is a difficult one because Alice and Bob can be creative and pick the images randomly from some secret shared database or even modify the noise component using a secret key. Another complication is that it is difficult to estimate the noise model from images containing the noise. The noise may be pixel-dependent and the only reliable way would be to scan each image many times to accurately estimate the noise component.

One can also argue that if Eve suspects that Alice and Bob exchange messages, instead of buying the same scanner and spending her time on studying noise statistics, Eve can simply apply her crypt-analysis to any image Alice and Bob exchange *without inspecting the images for the presence of secret information*. After all, Eve's goal is to get the message and not to prove that Alice and Bob communicate using steganography. Even if Alice and Bob are able to exchange 1000 images, it increases the required computational load by a factor of 1000, which will probably be only a marginal increase, if the stego-encryption system used by Alice and Bob is any good.

The field of steganography is probably very old. For a brief article on history of steganography, see and Michael Sattler's page [Sat]. According to one Herodotus story, a Persian nobleman named Histiaeus shaved the head of a trusted slave and tattooed a secret message on his scalp. Once the man's hair had grown back in, Histiaeus dispatched him to his destination with the instruction to shave his head. Other, more recent steganographic methods include invisible ink and microdots. Microdots are blocks of text or images scaled down to the size of a regular dot in a text. Shifting words and changing spacing between lines in text documents or in images containing text can also be used for hiding bits of secret messages [Br1, Br2, Br3]. Today, it seems natural to use digital images, digital video, or sound for hiding of secret messages. The gaps in human visual and audio systems can be used for information hiding. In the case of images, the human eye is relatively insensitive to high frequencies. This fact has been utilized in many steganographic algorithms which modify the least significant bits of gray levels in

[1] This is not a uniformly accepted principle, see Bob Tinsley's [Tin] arguments against Kerckhoff's principle.

digital images or digital sound tracks. Human eye is also insensitive to gradual changes in brightness. By overlaying an irregular pattern with gradual changes over the image, the embedded information is incorporated into low frequencies and becomes more robust with respect to common image processing operations and lossy compression.

Recently, there has been some confusion and misconceptions regarding steganographic methods in connection with digital data. Some authors do not correctly distinguish between invisibility, undetectability and robustness with respect to intentional and unintentional modifications, and the security. By definition, steganographic methods should not visibly modify the carrier. If the carrier is an image, the modifications should be imperceivable to the human visual system, if the carrier is an audio track, the changes should be imperceptible to the human ear. This, however, does not necessarily rule out the possibility that it may be possible to mathematically prove that the carrier has been modified by applying some sophisticated tests. *The purpose of data hiding can vary, and it is the specific application that imposes different requirements and limitations on the appropriate steganographic methods.*

In the next section, we divide data hiding schemes into three categories based on different requirements the techniques must satisfy. The state of the art in data hiding is reviewed in Section 3. A general theory of data hiding schemes, which is based on the paper by Comiskey et al. [Com] is presented in Section 4. We also present a slightly different framework which uses the concept of pattern overlaying. General principles for secure steganographic methods are discussed in Section 5. In Section 6, we discuss the purpose and required properties of watermarking schemes in detail, and in Section 7 a new, robust watermarking scheme is presented. The scheme is based on overlaying a pattern with its power concentrated mostly in low frequencies. The pattern is created using a pseudo-random number generator and a cellular automaton with voting rules. The robustness of the method with respect to filtering, JPEG compression, cropping, noise adding, and collusion is studied. In Section 8, we present a summary and outline future research directions. As part of our future effort, we describe a modification of the watermarking technique proposed by Cox et al. [Cox] to make the watermark extraction process possible without the original unwatermaked image.

The purpose of data hiding

Secret message hiding

Purpose	To send a secret message without raising any suspicion that a secret message is being sent.
Requirements	Secrecy, difficult to detect, large capacity.
Remarks	Always assume Kerckhoff's principle (protect the embedded information by a message and carrier independent key), maximize the capacity, minimize key size.

Watermarks

Purpose	Copyright protection, proof of ownership, fingerprinting (tracing distributed multiple copies)
Requirements	Robustness, difficult to remove even under collusion and Kerckhoff's principle.
Remarks	We do not require that the watermark not be detectable in the sense of maximum secrecy as in the case of covert communication

Captions

Purpose	To add information to an image, such as a caption.
Requirements	Medium capacity, accessible without a password or the original unmodified image, automatic detection and extraction possible.
Remarks	We do not require robustness or secrecy, cannot be combined with image escrow.

Data hiding - previous art

Secret message hiding
One class of data hiding techniques concentrates on permuting entries in binary documents. The most advanced scheme was developed by a Los Alamos research group lead by Handel and Sanford [Han]. The basic idea behind their method is to permute close colors with similar frequency of occurrence. This way, the data embedding process does not create new colors, which is an important issue for palette based images. Naïve least significant bit encoding creates characteristic groups of similar colors in the palette, which can be used for detection of the presence of a secret message [Tin]. Their method is adiabatic in the sense that it almost does not change the entropy of the image. Depending on the carrier image, it also provides a good capacity. The method depends on several parameters which can be adjusted to trade security for capacity and vice versa. In order to recover the secret message, the original image is needed, which means that image escrow is necessary. The authors study how their embedding procedure modifies the carrier's properties with respect to compression. While the image entropy is emphasized as a statistic for evaluating the changes occurring in the carrier image, other common statistical measures and correlations in the image also play an important role and should be taken into account.

On his web page, Bob Tinsley [Tin] proposes a different technique for secret message hiding by permuting the palette entries rather than the image itself. The obvious advantage of this approach is that the raw image data of the carrier is not modified at all.

On the other hand, this method can hide only relatively small number of bits (210 bits for GIF's) and could be easily detected for gray scale images. The permuted palette could also raise immediate suspicion. It is necessary to look at various image processing software tools to see how the palette is actually created. Some software products, such as PaintShop can order the palette entries according to luminance, chrominance, or hue. This method also requires the original image for message extraction.

A large number of secure message hiding techniques is based on modifying the least significant bit(s) of the carrier image. The simplest method of this kind encodes the message into the least significant bit of each pixel. It is very important that the message be first encrypted so that the LSB does not form any pattern and the message is protected. This scheme does not need image escrow. It can be slightly improved if Alice and Bob can agree beforehand on some pseudo-random pattern of the LSB of the original carrier instead of simply assuming that the unmodified carrier has all least significant bits equal to zero. This is equivalent to composing the first cryptographic scheme with a one time pad generated by the pseudo-random generator. The capacity of this scheme can be increased by using two or more least significant bits. Advantages of these schemes are good capacity and simple implementation. The method can also be adapted to JPEG compressed images and other transform based compression schemes, such as wavelets, by simply modifying the least significant bits of the transform coefficients. One big disadvantage of LSB encoding is that when carelessly implemented, it can seriously disturb certain statistical properties of the image. For example, for palette based images, groups of similar colors are created (see the critical evaluation of S-tools by Tinsley [Tin]). Modifications of the LSB of JPEG coefficients as done in J-steg 3.0 can also be easily detected [Tin].

A special steganographic technique for hiding messages into text files is desribed by Peter Wayner [Way]. Wayner uses context-free grammars to hide information into the grammar production rules. The productions can be recovered by parsing.

Hiding information in text documents or images containing text using word shifting, line shifting, and feature coding is described in [Br1, Br2, Br3]. Word shifting is changing the spacing between words, line shifting is changing the spacing between lines, and feature coding encompasses slightly changing symbols. Some software companies are already using similar techniques for shareware distribution of fonts. The customer is given a set of watermarked fonts for evaluation which are replaced with unwatermarked fonts upon registration. The changes to the document and/or the symbols are easy to detect and remove in electronic form. The word and line shifts can be removed simply by printing the document, applying OCR and reprinting the document. The advantages of this approach are that there is no need for binary data and the robustness with respect to copying. Also, the watermarks are not so easy to remove from printed documents.

Watermarking
Current best watermarking algorithms are based on spread spectrum techniques [Dix]. Hartung and Girod [Har] describe methods for marking digital video MPEG 2. Their

method is the direct sequence encoding in the spatial domain. No image escrow needed. Here is a simple summary of their technique.

The watermark sequence $\{w_i\}$, $w_i \in \{-1, 1\}$ is first spread by a "chirp rate" cr to

$$\{b_j\} = w_1, \ldots, w_1, w_2, \ldots, w_2, \ldots, w_i, \ldots, w_i, \ldots,$$

where each symbol w_i is repeated cr times. The gray values of the carrier c_j are modulated

$$s_j = c_j + \alpha\, b_j p_j,$$

where s_j denotes the gray values of the modulated carrier (stego-image), and p_j is a binary pseudo-noise sequence seeded with a secret key, and α is an amplitude factor. Recovery of the watermark is achieved by multiplying the sequence s_j with the pseudo-noise

$$\hat{c}_j = \sum_{i=j.cr}^{(j+1)cr-1} p_i s_i = \sum_{i=j.cr}^{(j+1)cr-1} p_i c_i + \sum_{i=j.cr}^{(j+1)cr-1} p^2{}_i a b_i \cong \Delta + cr.\alpha.w_j,$$

where

$$\Delta = \underset{i}{mean}(s_i) \sum_{i=j.cr}^{(j+1).cr-1} p_i.$$

This term is a correction for a possible non-orthogonality of the noise term and the carrier image. On average, it will be close to zero, because there will be little correlation between the carrier and a pseudo-random sequence. The recovered bit w_j is equal to $sign(\hat{c}_j)$. An improvement in security is obtained by replacing the binary pseudo-random sequence by a Gaussian sequence.

They also describe a method for watermark embedding directly into an MPEG bit sequence. The method is based on LSB encoding in the transform space.

Bender, Gruhl, Morimoto [Ben] describe techniques for data hiding in digital images and audio streams. Their patchwork method is a "stochastic" spread spectrum technique in the spatial domain. No image escrow needed.

Pairs of pixels (a_i, b_i) are pseudo-randomly picked from the carrier image. The expected value of the summation of differences $a_i - b_i$ is zero. The coding of a watermark consists of increasing a_i by one and decreasing b_i by one. Therefore, the expected value of the summation of differences will not be zero, but will depend on the length of the pseudo-random sequence. By making the sequence sufficiently long, we gain certainty that an image has been marked with a watermark. This method gracefully degrades with cropping and affine image transformations. It can be made more robust by replacing the pairs of points with patches of several points. This improves the robustness of the scheme with respect to low-pass filtering and most of common image processing modifications.

Another method of Bender, Gruhl, Morimoto [Ben] is called texture block coding. A small random looking area is copied into a different random looking area of the image. This creates a correlation which will not be disturbed by most image processing operations except cropping. If those areas contain an 8×8 square, the method will also be robust with respect to JPEG compression of any quality ratio. No image escrow is needed. However, no matter how impressive this robustness is, the watermark is easy to detect and remove. Another disadvantage is that it is image dependent, and not all images will have the required random looking areas.

Cox, Kilian, Leighton, Shamoon [Cox] introduced an extremely robust watermark which is based on discrete cosine transform and modifying the ***low*** frequencies by a small amount. In order to recover the watermark, the original image is needed. The authors report that it is possible to reliably extract the watermark from images after 5% JPEG compression! They have also found their technique to be robust with respect to resampling, dithering, cropping, and other common image manipulation operations. The watermark is also shown to be resistant with respect to the collusion attack (combining multiple watermarked documents to remove the watermark).

The authors make the watermark very robust by inserting it into the low frequencies. In a sense, they utilize the relative insensitivity of the human visual system to small, gradual changes in intensity. The watermark is a sequence of 1000 samples from a normal distribution with zero mean and unit variance, $\{w_i\}$, and is encoded into 1000 lowest frequency coefficients $\{v_i\}$ of the discrete cosine transform using the formula

$$v'_i = v_i(1+\alpha w_i),\ \ i = 1,\ldots,1000.$$

The constant α is used to adjust the magnitude of modifications, and was chosen to be equal to 0.1. In order to recover the watermark from a modified image, the modified image is first transformed using a discrete cosine transform to obtain the modified coefficients $\{v_i^*\}$. The watermark w^* extracted from the modified image is compared to the original watermark w using a similarity classification function

$$sim(w^*, w) = \frac{w * w}{\|w *\|}.$$

Based on the value of *sim*, a conclusion as to whether or not the modified image contains the watermark w is made. The authors desribe several improvements which make the watermark extraction process more accurate using robust statistics and by preprocessing w^* before calculating *sim*. The watermark is remarkably robust with respect to analog-digital conversions, requantizing, copying and subsequent scanning, dithering, etc. In a more general scheme, instead of choosing the lowest 1000 frequencies for the watermark embedding, the frequencies are chosen from M lowest frequencies, $M > N$. To wipe out the watermark, one would have to randomize the amplitudes of all low frequencies by the maximum amount allowed by the algorithm. This would, however, result in visible deterioration of the image. The authors study the robustness with respect to collusion by averaging 5 watermarked images and testing the presence of each watermark in the image.

The watermark is equivalent to overlaying a pattern spanned by *N* discrete cosines over the image. The pattern may become visible in approximately uniform areas, and areas with uniform brightness gradient. The pattern is directly related to the watermark sequence. This information may be utilized to remove the watermark beyond detection. The authors do mention that more localized basis functions, such as wavelets, could cause less noticeable modifications. Other possible improvements of this scheme are discussed in Section 7 and 8.

Pitas and Kaskalis [Pit] describe a method for applying signatures to digital images. The pixels of a digital image are divided into two disjoint sets of the same size. Pixels in one set are offset by an amount *k* while the pixels in the other set are left untouched. The authors discuss how *k* depends on the degree of certainty of a successful recovery of the signature (watermark). They also discuss how multiple signatures influence each other and estimate the probability of a successful signature detection in an image containing multiple signatures. Although no robustness study is reported, the similarity of this technique to the patchwork of Bender et al. [Ben] suggests that the method will have similar robustness properties.

Bruyndonckx et al. [Bru] introduce a spatial method for copyright labeling of digital images. One bit of information is embedded into an $n \times n$ square block of pixels. The block is chosen pseudo-randomly. First, the block is analyzed for luminance transitions and the pixels are divided into two zones, zone1, zone2, based on the derivative of luminance (if there is an edge in the block, pixels naturally form two zones). Then a binary pattern (two categories A and B) is further laid over the block (usually a checker board $n/2 \times n/2$). Pixels are thus divided into four groups depending on which zone and category they belong to. One bit of information is embedded by modifying the mean values m_{1A} , m_{1B} , m_{2A} , m_{2B} in the four groups 1A, 1B, 2A, and 2B:

$$m_{1B} - m_{1A} = L,\ m_{2B} - m_{2A} = L \text{ encodes bit 1}$$
$$m_{1A} - m_{1B} = L,\ m_{2A} - m_{2B} = L \text{ encodes bit 0.}$$

The number L is called the embedding level. According to the authors, L should be less than 9 to prevent the watermark from being visible. The pixel values are adjusted uniformly (i.e., the same change is made to all pixels from each zone and category). In order to keep the mean value in each zone intact, two more equations (constraints) are added.

Although the authors report their watermark to be robust with respect to JPEG compression and some kernel image processing operations, the watermark will not survive cropping. Also, it should not be difficult to test all $n \times n$ pixel blocks for presence of patterns imposed by the categories. This means that the watermark can probably be easily removed.

Zhao and Koch [Zha] propose a watermark based on JPEG compression scheme. One bit of information is embedded into middle frequencies of pseudo-randomly chosen 8×8 pixel blocks. In each block, a triple of frequencies obtained by discrete cosine transform is chosen out of 18 predetermined frequencies. Their coefficients are modified so that their mutual relationship encodes one bit of information. Since the 18 predetermined frequencies are chosen from the middle range, this method will be less robust and more vulnerable to noise than the method of Cox et al. [Cox] described above. Zhao and Koch describe another method designed for black and white images. Inn their method, the relative frequencies of black and white pixels in pseudo-randomly selected 8×8 blocks encode one bit of information. Advantage of this scheme is that no image transformation is involved. On the other hand, the method is vulnerable to collusion of several watermarked images. To overcome this problem, Zhao and Koch propose to choose distributed blocks instead of square blocks. However, this makes the method much more sensitive to noise.

M. Kutter, F. Jordan, F. Bosen [Kut] describe a method for a digital signature of color images using amplitude modulation. The authors hide the signature into the blue channel of color images. The logic behind this is that the human visual system is least sensitive to the blue channel. Given a bit a of the signature, it is encoded into the blue channel according to the formula:

$$
\begin{aligned}
&B_{ij} \leftarrow B_{ij} + (2a-1)L_{ij}q, \text{ where} \\
&L_{ij} = 0.299R + 0.587G + 0.114B.
\end{aligned}
$$

The factor q can be used to adjust the robustness and visibility of the signature. The pixels used for embedding are chosen pseudo-randomly according to some key K – the initial seed. The recovery of the embedded bit proceeds as follows. First, an estimate for the value of the blue component is made by averaging the blue components of four neighboring pixels. Then the average is compared to the value present in the marked image. Depending on their relative values, the bit a is obtained. In order to increase the robustness of the technique, the bit a is encoded on more than positions in the image. The technique can be easily extended to encode more than one bit. In order to make the retrieval more reliable, the authors add two bits, 0 and 1 to each signature. This enables them to calculate a better threshold for recovering the secret bits. Also, the pattern of randomly selected pixels enables them to recover the signature from images which have been affine transformed.

Theory of data hiding techniques

The theory is based on the paper by Comiskey and Smith [Com], who give a nice interpretation of data hiding schemes. Carrier image C is modulated by adding a linear combination of basis functions B_i to obtain a stego-image S. The functions C, S, and B_i are defined on a rectangular lattice of points (x, y) and take on integer values (in the case of gray-scale images and palette-based images) or triples of integer values for other color images.

$$\text{Stego-image (S)} = \text{Original carrier image (C)} + \text{Modifications (M)}. \tag{1}$$

$$\text{Modifications} = \sum_{i=1}^{n} m_i B_i(x, y) ,$$

where $m_i \in \{-M, -M+1, \ldots, M-1, M\}$, and B_i are defined on the image domain (usually a rectangle).

Security

Security is either guaranteed by keeping the original carrier image C secret or by making the basis functions B_i pseudo-random (key-dependent) or both. We note that if we insist that the original image be kept secret, it effectively becomes a part of a (large) key.

Recovery

Recovery of the hidden message is made possible either because we know the original image, or by making the basis functions orthogonal to the carrier image. If the original unmodified carrier image is available, we can simply subtract $S - C$ and solve the system of equations for the n unknown m_i :

$$S - C = \sum_{i=1}^{n} m_i B_i(x, y)$$

If the supports of the basis functions are disjoint, no equations need to be solved and the unknown m_i's can be readily recovered. In data hiding schemes without image escrow (without knowing the original image) we must require B_i to be orthogonal to the carrier image C. Recovery of m_i is then simply achieved by projecting[2] the stego-image S on each basis function B_i

---

[2] The dot product is defined in the usual way as $\left\langle B_i, B_j \right\rangle = \sum_{x,y} B_i(x, y) B_j(x, y),$ where (x, y) runs through all pixels in the image.

$$\langle S, B_i \rangle = \langle C, B_i \rangle + \sum_{j=1}^{n} m_j \langle B_j, B_i \rangle \cong \sum_{j=1}^{n} m_j \langle B_j, B_i \rangle.$$

In order to recover the hidden message, we need to solve this system of n equations. If we make the basis functions B_i orthogonal to each other, no equations need to be solved and we have an estimate $\hat{m}_i$ for m_i

$$\hat{m}_i = \frac{\langle S, B_i \rangle}{\langle B_i, B_i \rangle}.$$

This estimate will not be exactly equal to m_i for the following reasons. First, the basis functions will not generally be exactly orthogonal to the carrier C unless we design them so. If we make them exactly orthogonal, they will be carrier-dependent and thus become part of the key. Second, the stego-image S may be intentionally or unintentionally manipulated. In order to make the recovery process more robust, we suggest to calculate all the estimates $\hat{m}_i$ and threshold them to the discrete values $-M$, $-M + 1$, …, $M - 1$, M. In most data hiding schemes, the values m_i are of binary nature alternating between $-G$ and G without taking any intermediate values. Comiskey and Smith [Com] simply calculate all the estimates $\hat{m}_i$, $i= 1,\dots, n$, and threshold them with respect to the median of their maximum and minimum values. As they note, for this to work we need to make sure that not all m_i are the same.

Functions B_i can be spatially localized or localized in frequency space, or localized in both spatial and frequency domain. They can also be spread in space or frequency. Different choices of the basis functions lead to different data hiding schemes. For example, the simplest implementation of LSB encoding corresponds to B_i 's with a one-pixel support and $M = 1$.

Information hiding can also be interpreted as pattern overlaying. Imagine that C is watermarked by overlaying an almost translucent key-dependent pattern over it. All LSB schemes can be associated with patterns with large high frequency content, while spread spectrum methods use patterns with specific structures. Data embedding based on Eq. (1) utilizes patterns generated by a linear combination of some basis functions. The coefficients in the linear combination are directly related to the watermark content. This direct connection between the pattern and the watermark content may allow easier watermark removal depending on the original image. A different strategy would be to design a mechanism for building a pattern with low frequency content starting with a bit sequence – the watermark. It is important that the pattern depend sensitively on the watermark (see Section 7).

Secret message hiding

From T. Aura [Aur]: To embed a small message of the order of 8 bits or so, just keep scanning an image till a certain password-dependent message digest hash function returns the required 8-tuple of bits. This has the advantage of absolute secrecy tantamount to one time pad used in cryptography. We guarantee the same error distribution and undetectability. Although the scheme satisfies the requirements of a steganographic holy grail, it is time consuming, has very limited capacity, and is not applicable to image carriers for which we only have one copy.

Aura [Aur] suggest to play it safe and change only a small fraction of the carrier bits. For example, modify each hundredth pixel in the carrier by one gray level. Depending on the image noise, these changes will hopefully be compatible with the uncertainties involved with any statistical model of the image.

In this paper, we argue that before we can claim any secret message hiding technique as secure, we need to carefully investigate the carrier image and its statistical properties. The noise component may not be uniform within the image but may depend on the pixel position in the image. For example, pixels corresponding to a bright white color will probably be always saturated at 255 even though the overall model of the noise can be Gaussian with a non-zero variance. An example of such an image is shown in Figure 1. The image has been scanned on a scanner. Figure 2 is a black and white image with black pixels corresponding to even gray levels and white pixels corresponding to odd values of gray levels. One can clearly see a large patch of odd pixels which

Figure 1 A scanned image

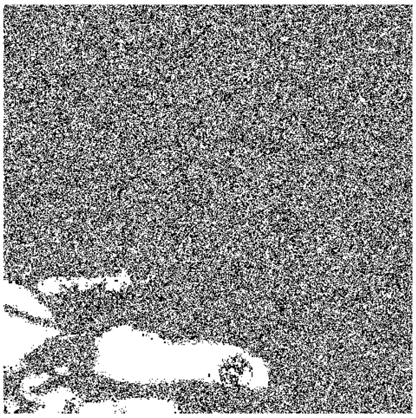

Figure 2 The same image with white pixels corresponding to odd gray levels and black pixels corresponding to even gray levels

corresponds to pixels saturated at 255 in the original image. Even if we play it safe and modify only a small fraction of pixels in the image, we will introduce some suspicious noise into the overflowed patch. This problem with over/under flow can of course be avoided by a more careful choice of the carrier image, preprocessing the carrier, or by instructing the steganographic scheme to avoid the over/under flowed areas.

While we agree that it is probably impossible to get a complete model of the carrier noise, and that the search for the perfect steganographic method will probably never be complete, we insist that all good secret hiding schemes must be based on some model of the noise. If it is known that scanned images exhibit larger noise correlations in the horizontal direction and smaller correlations in the vertical direction, while the probability distribution for each pixel, which is neither overflowed nor underflowed, is Gaussian with certain standard deviation, then we have to take this evidence into account and tailor our secret message hiding scheme so that the carrier modifications are consistent with the statistical evidence. It is certainly possible that somebody will, with great effort, create even more sophisticated noise model and detect the presence of messages, but only at the price of painstaking time-consuming and possible expensive investigation. It is rather unfortunate but understandable, however, that most detailed technical information regarding noise in CCD arrays and scanners is proprietary and kept secret by private companies [Whi].

To give an example how one can incorporate a statistical evidence into the construction of a secret message hiding scheme is as follows. Let us assume that the noise component of pixels with gray levels within the range [L, H] can be modeled with a uniformly valid probability density, f, symmetric around zero. If the secret plain-text message $\{p_i\}^N_{i=1}$ is encrypted, the cipher-text $\{c_i\}^N_{i=1}$ should be a random sequence of ones and zeros. By averaging several scanned versions of the carrier image, we obtain a “zero noise” image Z. Using a pseudo-random number generator, we can choose at random N pixels in Z with their gray levels in [L, H]. Then, we can modify the LSB of those pixels by the amount of $(2b_i - 1)\,|\eta_i|$, where η_i is a random variable with probability distribution f. The remainder of the pixels will be modified by η_i. The modifications should be consistent with the statistical model.

Digital watermarks

The most important requirement for digital watermarks is their robustness with respect to common image processing procedures, such as kernel filtering, lossy compression, digital - analog conversion, resampling, cropping, affine transformations, removing / adding features, noise adding, copying, and scanning. The second important property of digital watermarks is their non-removability. This could be achieved by making the algorithm for watermark embedding cryptographically strong using a secret key. The purpose of a digital watermark is to prove the ownership of digital data. Robustness with respect to these operations can only be achieved at the price of making detectable (i.e., not subtle) changes to the image. It is probably impossible to create an undetectable (in steganographically secure sense) watermark which is robust.

If we can design an algorithm for embedding of robust watermarks, the watermarks can be used for solving a dispute about an ownership of some digital data. Let us say that A creates a digital image and watermarks it with his key. If B gets hold of the watermark image and wants to steal the image (i.e., claim the ownership), the best he can do is to embed his watermark into the image and claim that he can prove the ownership of the image. If the watermark is robust, it is easy for A to prove the ownership to a judge because A can detect his watermark in the image marked by B, while B cannot do that for the image that A has. Since we trust the watermarking algorithm that it cannot be completely removed unless one modifies the image beyond recognition, this proves the ownership of the image. We repeat that the watermark's robustness with respect to changes plays a crucial role in this dispute because if B can filter out the original watermark, the watermark would be useless. To prevent creating a forged original by subtracting a chosen watermark pattern (the IBM attack [Cra]), the watermark must depend on the image in a non-invertible manner.

What does it mean to prove that watermark is there? It is to show that certain relationship among pixels (the reconstructed watermark) can only occur by a pure chance with an extremely low probability.

Another purpose of data embedding is to mark images so that large modifications of the image, such as removing and/or adding objects can be detected. In this case, the watermark would carry some information about the features present in the image. For example, we may robustly encode values of several "hash functions" of the original image. The word hash is in quotes because we actually do not want an absolute sensitivity with respect to the image data, but some other functions which would be non-collision in some robust sense. For example, we could take a correlation of the original image with one or more secret masks. While this would be effective against removing / adding of features, it may not work as well for, say exchanging two small human faces in a photograph.

A new watermarking scheme

In this section, we propose a new robust watermarking technique with the intention to remove a possible weakness in the scheme introduced by Cox et al. Their watermarking method can be interpreted as overlaying a pattern which is formed by a linear combination of discrete cosines. The watermark values are used directly as coefficients of that linear combination. The watermark may become visible (or at least detectable) in those areas of the carrier image which were originally uniform or had a uniform brightness gradient[3]. The watermark cannot be readily removed because the discrete cosines do not generally form a set of linearly independent functions on proper subsets of the image. Nevertheless, this situation is potentially dangerous and could be used to mount an attack on the watermark. Let us assume that a square area containing K pixels reveals some approximation to the watermark. Since we know that the watermark is spanned by the lowest 1000 coefficients, we can write K equations and narrow down the possibilities for the watermark sequence by a large margin. In the equation below, $f_r = (f_{1r}, f_{2r})$ denotes the r-th 2d frequency, and A_r denotes the r-th unknown coefficient.

$$g_{ij} = \sum_{r=1}^{1000} A_r \cos\left(\frac{\pi f_{1r}(2i+1)}{2N}\right) \cos\left(\frac{\pi f_{2r}(2j+1)}{2N}\right), \ (i, j) \in \text{Area}$$

A simplified version of the technique described in [Cox] has been successfully attacked in [Fri]. The above attack utilizes the fact that the watermark is localized in a relatively small number of coefficients of a publicly known transformation – the discrete cosine transformation. If the choice of the basis functions was kept secret, this type of attack would not be possible. In order to make this practical, however, we would have to design orthogonal bases which would depend on parameters – a secret key and a hash of the image. A technique that is based on incomplete orthogonal bases is studied in [Fri].

Another possibility how to overcome the problem described above is to view the watermarking scheme as pattern overlaying. We do not have to use patterns formed by a linear combination of discrete cosines but we could utilize general key-and-image-dependent patterns with their power concentrated mostly into low frequencies in order to guarantee robustness. This idea is the basis of a new watermarking technique described below.

We start with a procedure using which a string of bits (author's ID + image hash) can be transformed into a smooth, almost transparent pattern to be overlaid over the carrier image. We require a sensitive dependence between the watermark and the resulting pattern. The pattern should not exhibit traces of any regular building blocks. Also, patterns generated by two different

[3] Quite a large percentage of images do contain such areas.

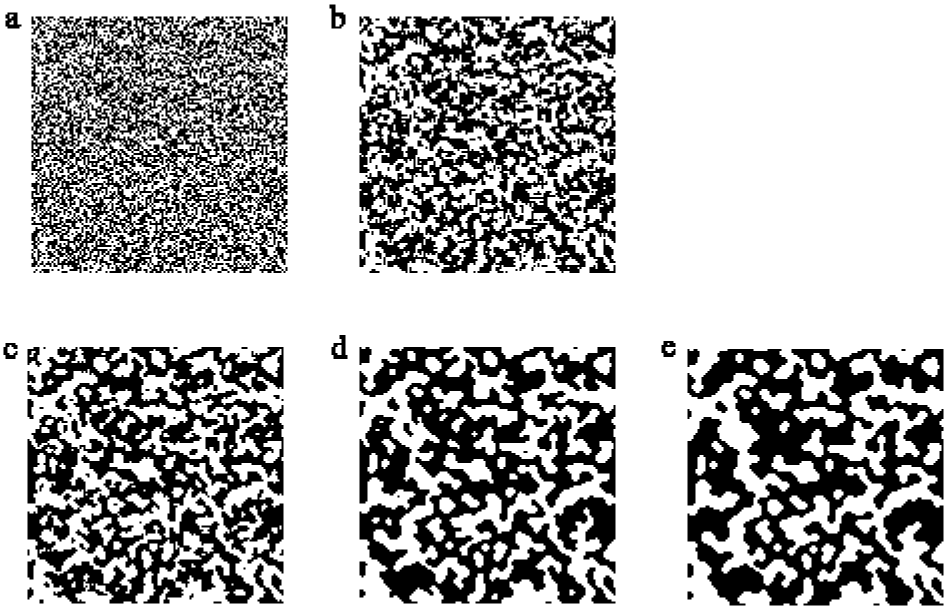

Figure 3 The initial pattern a) after 1, 2, 5, and 35 iterations of a cellular automaton with voting rules

watermarks should be uncorrelated. While we realize that there probably are many ways how to achieve these goals, we propose to seed a random number generator with the watermark to create an initial black and white two-dimensional random pattern. The pattern will be further processed to eliminate high frequencies. This can be done for example by applying low-pass filters to the initial pattern. Another possibility would be to use the initial pattern as an initial condition for some chaotic spatio-temporal dynamical system which has the tendency to create large scale smooth structures. Still another possibility would be to use cellular automata with appropriate rules. For example, it is known that voting rules and their modifications [Tof] can coalesce random patterns into large-scale structures.

In Figure 3a we show an example of a randomly generated initial pattern. Since the purpose of this study is to show the plausibility of the proposed watermarking scheme, we used a simple random number generator supplied with Matlab V. A secure implementation of this scheme would use the concatenation of author's ID and a hash of the original image (to thwart the IBM attack [Cra]). The initial pattern was initialized with 0's and 1's with the same probability 50%. The random black and white pattern was then processed by a cellular automaton with the voting rule [Tof]. According to this rule the center element follows the majority of its neighbors. In particular, we count the number, P, of 1's in the 3×3 neighborhood of the center (including the center) and set the center to 0 if $P < 5$, otherwise we set the center to 1. The cellular automaton with voting rules always stops after finitely many steps (for a 128×128 image, we found this number to be always less than 40). The resulting pattern together with three intermediate patterns is shown in Figure 3b – 3e. As can be seen, the random pattern has coalesced into several connected areas forming an irregular pattern which depends on the seed of the random number generator (and therefore on the watermark sequence) in a complicated way. As the last step, we applied a smoothing operation with a 7×7 kernel eight times. The color depth of the resulting image was decreased to 16. Examples of five different patterns are shown in Figure 4 . Before we add these patterns to an image, we subtract 8 from each pixel value of the pattern so that the pixels of the original unwatermarked image are modified by no more than ±8 gray levels.

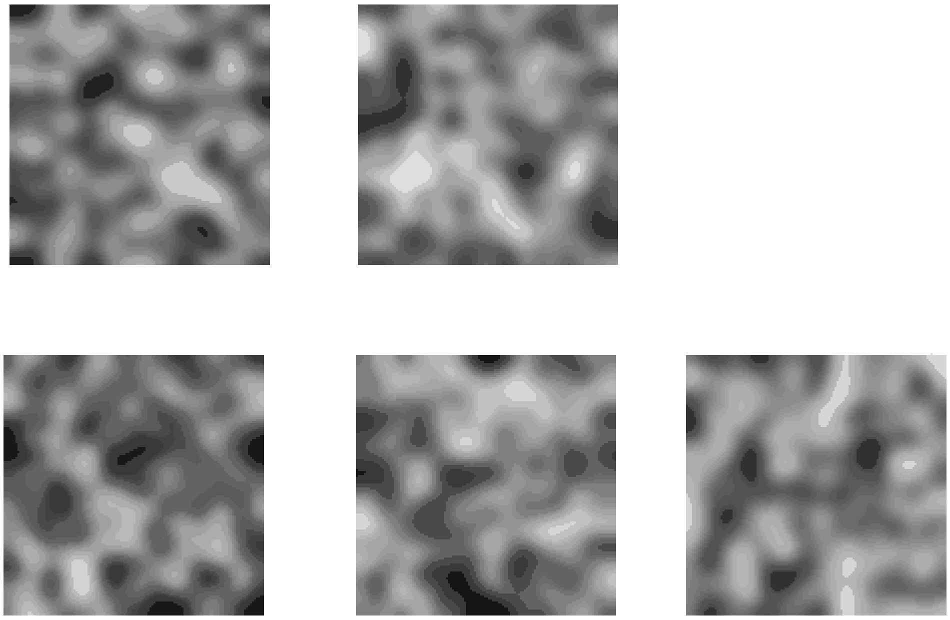

Figure 4 Examples of five watermarking patterns. The contrast was enhanced to make the patterns visible.

Since the overlaid image has most of its power concentrated in low frequencies, we expect excellent robustness properties similar to the method proposed by Cox et al. [Cox]. Since our overlaid pattern depends on the key in a complicated way, even if the watermark shows in regions of nearly constant luminosity, it does not reveal any information about the key if a cryptographically strong random number generator is used. Another advantage of this technique is that it avoids transformations, which results in a faster and easy implementation.

Our new scheme for embedding digital watermark data can be described as follows: Watermark sequence = author's ID + image hash $\rightarrow$ seed for a random number generator $\rightarrow$ initial two-dimensional black and white random pattern $\rightarrow$ cellular automaton $\rightarrow$ smoothing algorithm $\rightarrow$ adding the pattern to the image.

Watermark retrieval: Subtract the watermarked image from the original and calculate the correlation between the difference and the smoothed pattern. Based on the value of the correlation, decide about the presence of the watermark. In our experiments, the correlation in the Fourier space performed better than in the spatial domain. Denoting the original unwatermarked image as *X*, the watermarked image *X'*, the modified watermarked image X^* the following function was used to decide about the presence of a watermark:

$$sim(X^*, X') = \frac{D^* \cdot D}{\sqrt{D^* \cdot D^*}\sqrt{D \cdot D}},$$

where $D = Y' - Y$, $D^* = Y^* - Y$, and the symbols Y, Y', and Y^* denote the lowest 1024 discrete cosine coefficients corresponding to X, X', and X^*, respectively.

Since it appears rather difficult to analyze the mechanism how cellular automata coalesce randomly scattered points into connected clusters, we base our analysis on a series of computer experiments only. Our conclusions regarding the robustness of the method are based solely on computer experiments. Clearly, this is not satisfactory, and a more exact analysis of the method needs to be done. This will be an important part of the future research.

We determined experimentally that for images X and Y watermarked by a different watermark, $sim(X, Y)$ is mostly confined to the interval [-0.05, 0.05] with the maximal observed value of 0.175.

Figure 5 The original test image (left) and a watermarked copy (right)

Robustness with respect to JPEG compression. The carrier image with 128×128 pixels and 256 gray levels and its watermarked version are shown in Figure 5. The marked image has been compressed using JPEG compression method with quality factors 50%, 20%, 10%, and 5%. The compressed images are shown in Figure 6.

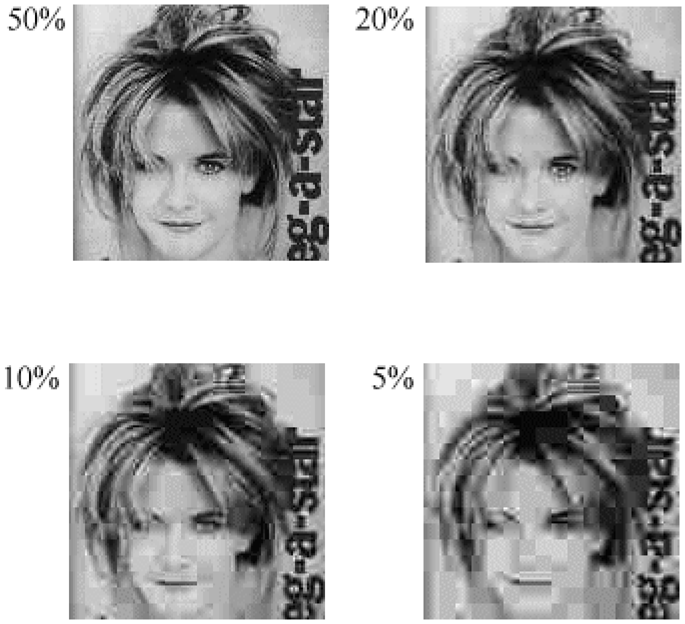

Figure 6 The watermarked image JPEG compressed with 50%, 20%, 10%, and 5% quality factor.

As can be seen, the quality factor 5% severely distorts the original image. Nevertheless, the watermark can still be detected. The corresponding values of *sim* are shown in the table below.

quality factor	50%	20%	10%	5%
sim	0.97	0.84	0.62	0.38

The table clearly shows that common JPEG compression with quality factor equal to 50% almost does not disturb the watermark. As low quality factor as 5% still leaves the watermark detectable because the value of *sim* = 0.38 is twice as much as the largest similarity for two random watermarks. A more detailed statistical analysis of this statement is clearly needed and it will be part of the future research.

Figure 7 Test image after downsampling by a factro of 2 and upsampling

Robustness to mosaic filtering. The test image 128×128 has been downsampled to a smaller image 64×64 and then upsampled again to the previous size (see Figure 7).

Although 75% of information has been lost in this process, the value of *sim* for the mosaic filtered image was 0.44, which indicates the presence of the watermark.

Robustness to blurring. We used PaintShop Pro and its “blur more” filter to blur the test image. The resulting blurred image is shown in Figure 8. The similarity number 0.43 shows the presence of the watermark.

Figure 8 Blurred image

Robustness to median filtering. PaintShop’s median filter was used to create Figure 9. The similarity value for the filtered image was 0.71. This value indicates the presence of the watermark..

Figure 9 Median filtered test image

Robustness to cropping. Figure 10 shows a white rectangular area which has been cropped from a watermarked image. The cropped watermarked area forms about 23% of the total image area. The cropped image was completed by the unwatermarked image and the similarity function applied to that image returned the value 0.48. This shows that even though more than 77% of image information has been discarded due to cropping, the watermark remains detectable.

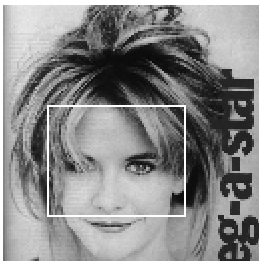

Figure 10 The white rectangle indicates the cropped area

Robustness to uniform noise. We added 30% of uniform noise into the test image (see Figure 11) and tested for presence of the watermark. The similarity value of 0.57 clearly indicates the watermark's presence.

Figure 11 The test image after adding 30% of uniform noise

Robustness with respect to collusion attack. In this test, we created 5 different watermarks and averaged the watermarked images into a single image. The (enhanced) watermark patterns are shown in Figure 4. Then, we tested the averaged image for presence of the five watermarks. The similarity values are shown in the table below.

Watermark #	*w*1	*w*2	*w*3	*w*4	*w*5
sim	0.42	0.44	0.48	0.56	0.44

We also tested the averaged image for presence of 20 randomly generated watermarks. In neither case was the value of *sim* greater than 0.17.

Conclusion and future directions

Based on the requirements for secrecy, robustness, and detection, we distinguish in this paper three major different purposes of data hiding: secret communication, watermarks, and captions. Current data hiding methods are briefly described and critically evaluated. We discuss a theoretical framework for data hiding based on the paper by Comiskey et al. [Com] and provide another, alternative framework based on key-dependent pattern overlaying. A possible security weakness of the method proposed by Cox et al. [Cox] is pointed out. The attack uses the fact that areas of the image which are almost uniform or have an almost constant brightness gradient may show a portion of the watermark pattern. Since the watermarked pattern is formed by a linear combination of the 1000 lowest frequency discrete cosines, a small area in which the watermark pattern shows may be utilized to distort the watermark beyond reliable detection. To overcome this weakness, we propose a new robust watermarking method based on pattern overlaying. Since the pattern will be formed in a sensitive way based on the watermark sequence, even if the watermark pattern shows in uniform areas, it is not possible to mount an attack. The watermark bit sequence is used for initializing a pseudo-random generator to create a random black and white initial pattern of the same size as the image. A cellular automaton with voting rules is applied till a convergence to a fixed point is obtained. The voting rule coalesces random patches into connected areas. The pattern is further filtered by a smoothing filter to move the main portion of the power to low frequencies. The gray levels of the final pattern are scaled to a small range and the pattern is finally added to the image. The watermarked image shows no visible degradation caused by the overlaid pattern, yet the pattern is embedded in a robust sense. It is possible to prove the presence of the pattern in images after filtering, JPEG compression with as low as 5% quality factor, cropping, resampling, blurring, downsampling, and noise adding. The watermark also appears to be resistant with respect to the collusion attack (averaging several watermarked images to remove the watermark). Since the relationship between the watermark sequence and the final watermark pattern is quite complex, it is difficult to perform a mathematical analysis of the robustness. Therefore, our conclusions regarding the robustness of the method are based solely on computer experiments. Clearly, this is not satisfactory, and a more exact analysis of the method needs to be done. This will be an important part of the future research.

The proposed watermarking method requires the original, unwatermarked image in order to retrieve the watermark. A large portion of our future effort will be directed to the design of a technique with the same degree of robustness which would not require the original image to recover the watermark. At present, we are investigating the following design.

Robust watermarking scheme without the need for the original image (undergoing effort)

Watermarking schemes which do not require the original image for watermark extraction always create some correlations in the image such that the probability of occurrence of those correlations by pure chance is very small. Bender's texture block mapping technique (see Section 3) achieves that by cropping a random looking area in the image and pasting it in some other random looking area. However, this watermark can be readily detected and removed. We need to create correlations which would be protected by a secret password. We choose to work in the Fourier space and make modifications to the coefficients of discrete cosine transform rather than to the pixels.

We base our method on the scheme of Cox et al [Cox]. Let $0 < \alpha < 1$ be a constant, and let us define the sequence

$$\{x_i\}_i,\ x_i = \frac{1+\alpha}{1-\alpha} x_i,\ i = -\infty, \cdots, \infty.$$

Further, define a function *ind*(x) as

$$ind(x) = 1 \quad \text{if } x_i \le x < x_{i+1},\ i \text{ even}$$
$$ind(x) = 0 \quad \text{if } x_i \le x < x_{i+1},\ i \text{ odd.}$$

It is easy to see that any x can be modified by adding or subtracting at most αx to change its $ind(x)$.

Transform the image to be watermarked using a discrete cosine transform. Choose the 1000 lowest frequency coefficients v_i and modify them by $\pm\alpha w_i$ so that

$$ind(v'_i) = (-1)^i, \; v' = v_i(1 + \alpha w_i), \; i = 1, \ldots, 1000.$$

Perform these modifications in a specific order determined by a pseudo-random generator (*the order in which the 1000 coefficients are chosen is important*). The properties of the sequence $\{x_i\}_i$ guarantee that the required changes will never have to be greater than $\pm\alpha v_i$. However, we may deliberately make the modifications larger in an attempt to make the distribution of the modifications as close to a Gaussian distribution as possible. To prove to a judge that there is a watermark in the image, you present the seed for your pseudo-random number generator and calculate $ind(c_i)$ for the 1000 low frequency coefficients in the order specified by the generator. Then correlate this sequence with $\{(-1)^i\}_{i=1}^{1000}$ and argue that the chances of obtaining this degree of correlation by pure chance are remote. If the judge questions your proof by accusing you of cooking up the seed so that a high correlation value is obtained, you can argue that since a cryptographically strong pseudo-random generator was used, there is no way to efficiently solve this inverse problem.

By keeping the constant α as part of the key, we further decrease the chances of an adversary to remove the watermark. Detail investigation of this scheme is part of our future research effort.

Acknowledgments

The work on this paper was supported by an SBIR grant *AF97043 Innovative C4I Technologies* provided by Rome Laboratory, Air Force Material Command, USAF. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Rome Laboratory, or the U. S. Government.

References

[Aur] Aura T., Invisible communication. In *Proc. of the HUT Seminar on Network Security '95*, Espoo, Finland, November 1995. Telecommunications Software and Multimedia Laboratory, Helsinki University of Technology.

[http://deadlock.hut.fi/ste/ste_html.html], [ftp://saturn.hut.fi/pub/aaura/ste1195.ps]

[Ben] Bender W., D. Gruhl, and N. Morimoto, "Techniques for Data Hiding." In *Proc. Of SPIE*, vol. 2420, p. 40, 1990.

[Br1] Brassil J., S. Low, N. Maxemchuk, L. O'Goram, "Document Marking and Identification using Both Line and Word Shifting," Infocom95. [ftp://ftp.research.att.com/dist/brassil/1995/infocom95.ps.Z]

[Br2] Brassil J., S. Low, N. Maxemchuk, L. O'Goram, "Electronic Marking and Identification Techniques to Discourage Document Copying," Infocom94. [ftp://ftp.research.att.com/dist/brassil/1994/infocom94a.ps.Z]

[Br3] Brassil J., S. Low, N. Maxemchuk, L. O'Goram, "Hiding Information in Document Images," CISS95. [ftp://ftp.research.att.com/dist/brassil/1995/ciss95.ps.Z]

[Com] Comiskey, B. O. and J.R. Smith, "Modulation and Information Hiding in Images," in: *Information Hiding, First International Workshop*, edited by Ross J. Anderson. Cambridge, U.K., May 30–June 1, 1996, Proceedings. Lecture Notes in Computer Science, Vol. 1174, Springer-Verlag, 1996 [http://sunsite.informatik.rwth-aachen.de/dblp/db/conf/ih/ih96.html]

[Cox] Cox J. I, J. Kilian, T. Leighton, and Shamoon T., Secure Spread Spectrum Watermarking for Multimedia," NEC Research Institute, Technical Report 95-10.

[Cra] Craver, S., N. Memon, B.-L. Yeo, and M. Yeung. "Can invisible watermarks resolve rightful ownerships?" *Proceedings of the IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases V*, San Jose, CA, USA, Feb. 13–14, 1997, vol. 3022, pp. 310–321.

[Dix] Dixon R. C., *Spread Spectrum Systems with Commercial Applications*. Wiley, New York, 1994.

[Fri] Fridrich, J., "On Digital Watermarks," paper for the 2nd Information Hiding Workshop in Portland, Oregon, April 15–17, 1998.

[Gir] Girod B. and F. Hartung, "Watermarking Method and Apparatus for Compressed Digital Video", US Patent application, 1996. [http://www-nt.e-technik.uni-erlangen.de/~hartung/watermarking.html]

[Han] Handel and Sanford [http://www.lanl.gov/users/u078743/embed1.htm]

[Han] Maxwell T. Sandford II, Jonathan N. Bradley, and Theodore G. Handel. The data embedding method. In *Proc. of the SPIE Photonics East Conference*, Philadelphia, September 1995.

[Har] Hartung F. and B. Girod, "Digital Watermarking of Raw and Compressed Video", *Proc. European EOS/SPIE Symposium on Advanced Imaging and Network Technologies*, Berlin, Germany, Oct. 1996.

[Har] Hartung F. and B. Girod, "Watermarking of MPEG–2 Encoded Video Without Decoding and Re-encoding", *Proc. Multimedia Computing and Networking 1997 (MMCN 97),* San Jose, CA, February 1997.

[Har] Hartung F. and B. Girod, "Digital Watermarking of MPEG–2 Coded Video in the Bitstream Domain", *Proc. ICASSP 97*, Munich, Germany, April 1997.

[Har] Hartung F. and B. Girod, "Copyright Protection in Video Delivery Networks by

Watermarking of Pre-Compressed Video”, *Proc. European Conference on Multimedia Applications, Systems and Technologies (ECMAST 97)*, Milano, Italy, May 1997.

[Kah] Kahn D., *The Codebreakers*, The Macmillan Company. New York, NY 1967.

[Koc] Koch E., J Zhao, Towards Robust and Hidden Image Copyright Labeling, *Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing* (Neos Marmaras, Halkidiki, Greece, June 20–22, 1995) and [http://www.igd.fhg.de/www/igd-a8/pub/IEEE_Hidden.ps]

[Kur] C. Kurak, J. McHugh, “A Cautionary Note On Image Downgrading,” *IEEE Eighth Annual Computer Security Applications Conference*, 1992. pp. 153–159.

[Nor] Norman B., *Secret Warfare*, Acropolis Books Ltd. Washington, DC 1973.

[Sat] Michael Sattler’s page [http://www.indstate.edu/msattler/sci -tech/comp/privacy/topics/steganography.html].

[Sch] Schnydel R. G. van, A. Z. Tirkel, and C. F. Osborne, “A Digital Watermark,” *In Int. Conf. Image Processing*, vol. 2, p. 86–90. IEEE, 1994.

[Tin] Bob Tinsley [http://www.sun.rhbnc.ac.uk/~phac107/]

[Tof] Toffoli, T. and N. Margolus, *Cellular Automata Machines*, MIT Press, 1987.

[Too] Tools For Privacy copyright 1995 Smart Publications [ftp://ftp.crl.com/users/ro/smart/TFP/steganography.html]

[Way] Peter Wayner [http://www.funet.fi/pub/crypt/old/mimic/Peter Wayner].

[Whi] Whitfield, A. Personal discussion. April 1997.

[Zim] Zim H. S., Codes and Secret Writing, William Marrow and Company. New York, NY, 1948.

[Zha] Zhao J., E Koch, Embedding Robust Labels Into Images For Copyright Protection, *Proc. Int. Congr. on IPR for Specialized Information, Knowledge and New Technologies* (Vienna, Austria, August 21–25, 1995) and [http://www.igd.fhg.de/www/igd-a8/pub/EmbedLabel.ps]