An Efficient Informed Embedding Scheme Using Systematic Nested Block Codes over Gaussian Channel

Chi-Yuan Lin, Sheng-Chih Yang, Jyun-Jie Wang and Cheng-Yi Yu
Department of Computer Science and
Information Engineering, National
Chin-Yi University of Technology,
Taichung 411, Taiwan, ROC
chiyuan@ncut.edu.tw

Abstract: We present a high-capacity informed embedding scheme based on a trellis structure for a nested linear block code. This scheme can embed adaptive robust watermarked messages for various applications. Instead of using randomly generated reference vectors as arc labels, this scheme uses the codewords of a nested block code to label the arcs in the trellis structure so that each codeword can carry different amounts of hidden payload. The proposed algorithm attempts to achieve two objectives: first, to minimize the modified position for each watermarked image; second, to perform the proposed embedding algorithm to minimize the amplitude distortion for these modified positions. Additionally, the proposed algorithm can perform iteration to determine a tradeoff between robustness and fidelity using numerous controllable parameters. Finally, the experimental results report the robustness and fidelity performance of this algorithm in AWGN attack channels. The experiment also simulates computational complexity and the proposed section-based informed embedding, which requires less operational complexity compared with Miller's informed embedding.

Key–Words: Data hiding, Informed embedding, Digital watermarking, Robustness

1 Introduction

Because of the large number of applications of the internet and other public communication networks, information hiding has received rising interest, and has played an important role in multimedia technology. The encoding process of data hiding codes, also known as watermarking codes, is to hide or embed a watermark into another host signal, such as a photograph, music, video, or text. The two main requirements of information hiding are fidelity and robustness, that is, the watermark message must not cause severe degradation on the host signal and must suffer from some common signal processing and channel attacks. Fundamental tradeoffs occur among payload, robustness, and complexity. This study developed practical algorithms by analyzing these tradeoffs, robustness, fidelity and complexity. Other design criteria for digital watermarking are payload, security, and detectability.

Referring to [1] and [2] for a comprehensive survey of data hiding codes, the considered watermarking system had no knowledge of the host signal in the receiver, that is, a watermarking system with a blind detector. To embed a watermark in such a system, a host signal can be viewed purely as noise, called blind watermarking, or exploited as side information, called informed watermarking. The corresponding system with blind detector and informed watermarking can be modelled as communication with side information at the transmitter [3], and allows more effective watermark embedding and detection methods. In general, the encoding process of informed watermarking is divided into informed coding and informed embedding. The purpose of informed coding is to choose a message codeword from a collection of possible candidates to represent this watermark. This message codeword must have minimal perceptual distortion to the host signal compared to other candidates. The informed coding is also known as dirty paper codes [4, 5] or channel coding with side information [6, 7, 8], in which the binning scheme is used to achieve the information-theoretic capacity [9, 10]. In informed embedding, the message codeword from informed coding is subsequently modified according to the host signal, attempting to attain an optimal tradeoff between fidelity and robustness in the watermarked image [11, 12, 13, 14, 15]. This study focused on the informed embedding method, in which the watermarked image is a function of the watermarked message and the host signal to achieve near optimal robustness and maintain constant fidelity, or vice versa. Miller et al. [13] proposed a suboptimal

trellis-based embedding algorithm that starts with the host signal and iteratively constructs an updated watermarked signal toward to the interior of the Voronoi region of the message codeword. In [13], an informed embedding algorithm used randomly generated reference vectors as arc labels. A disadvantage is that the generated reference vectors can be selected randomly; therefor, the trellis code is not an optimally structured code. In addition, modification of trellis structure modifies such generated reference vectors. Thus, it is impractical to use generated reference vectors as arc labels. Although this [13] trellis-based algorithm can achieve an excellent tradeoff between the fidelity and robustness in watermarked images, this method is computationally intensive and difficult to implement.

Instead of using randomly generated reference vectors as arc labels as in [13], we modified this trellis structure using the codewords of a nested block code to label the arcs in the trellis. The advantage of using such codewords is that they can be easily obtained in the tradeoff between embedding capacity and message robustness. We subsequently applied the characteristic of the linear nested block codes to the trellis partition. We proposes a modified trellis structure, in which each arc is labeled with nested block codewords for each trellis section. By using the input number and the memory state of a convolutional code, the embedded structure can modify the capacity and robustness of embedded messages. By adjusting the controllable parameters, the user can flexibly make a tradeoff between the embedding fidelity and embedding robustness.

The proposed algorithm is intended to meet two objectives. The first is to minimize the position of the changes of watermarked images in a trellis section by using an optimal quantized algorithm based on a nested block code (Subsection 3.1). The second is to embed a message based on the low-complexity section-based informed embedding (SBIE) algorithm, to minimize the amplitude of watermarked images. The SBIE algorithm is section-based, rather than using an entire trellis in one iteration. The section-based method enables algorithm performance in each section with iterative operation to find the suitable embedding watermarked images. The experiment indicated that the algorithm achieves a lower degree of complexity and excellent results under an AWGN attack, at the cost of robustness. The proposed algorithm can be easily implemented with less complexity, compared with other informed embedding methods. The experiment with the proposed algorithm was compared with that in [13]. First, considering embedded distortion and capacity, the parameters are simulated as a function of watermarked image quality. Second, we report the robustness performance of this algorithm in terms of Gaussian noise. Finally, we briefly tabulate the complexity comparison.

The rest of the paper is organized as follows: Section 2 presents a brief review of trellis-based informed embedding in [13]; Section 3 provides a description of our major work on informed embedding; Section 4 provides experimental results and constructive discussions; and finally, Section 5 offers conclusions.

2 Basic Informed Embedding for Miller's Work

The main goal of informed embedding is to find a good watermarked image, which is inside the decoding region of the message codeword, and has minimal perceptual distortion from the host signal. In general, it is difficult to find this optimal watermarked image. However, several approaches are used to find other suboptimal watermarked images, such as trellis-based informed embedding by Miller et al. [13]. Assuming that each path in the trellis corresponds to a message codeword of a watermark, the trellis-based informed embedding in [13] uses the Viterbi decoder to find a good watermarked image. The geometric interpretation of suboptimal embedding algorithm, as illustrated in Fig. 1, requires iterative updating of the watermarked signal by running the Viterbi decoder to identify a vector c^1 in the first iteration that has the highest correlation with the current watermarked signal, $x^0 = v$.

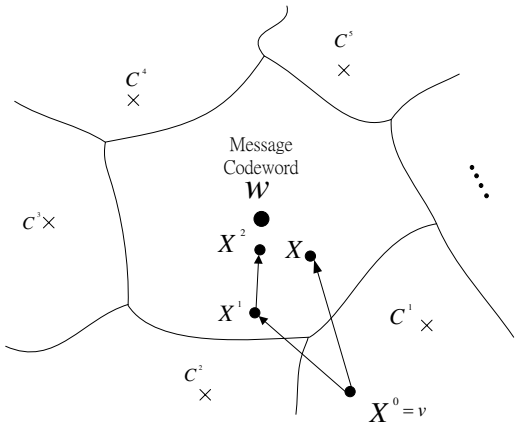

Figure 1: A trellis-based informed embedding [13].

By using vectors c^1 and x^0, we subsequently obtained a new watermarked signal x^1 closer to the decoding region of the message codeword w. The embedding process does not terminate until the final watermarked image falls inside the interior of the

Voronoi region of w. The final watermarked image of this algorithm, x^2 in Fig. 1, may not be the same as the optimal image, x in Fig. 1.

This embedding process is time consuming because Viterbi decoding is usually repeated several times before a final watermarked image is obtained. This paper proposes a trellis-based informed embedding with controllable parameters by modifying the arc labels of the trellis structure in [13]. The basic block diagram of proposed embedding method is shown in Fig. 2.

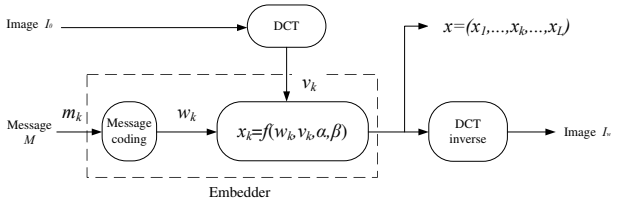

Figure 2: Block diagram of informed embedding based on controllable parameters.

As done in [13], the watermark was embedded in the frequency domain of the host signal, rather than on the host image.

First, a host signal I_o with dimensions $N = 512 \times 512$ was divided into 4096 blocks of size 8×8; subsequently each block was converted into the frequency domain with the DCT transform. The first 12 low-frequency AC coefficients in each block, shown in Fig. 2 of [13], where extracted and concatenated to form the extracted vector v. Every n coefficients of v was subsequently used to embed each bit of an L bits watermark, where $L = 4096 \cdot 12/n$, and forms the watermarked image x.

Finally, we replaced the elements of x into their respective DCT coefficients, and converted all DCT blocks back to the spatial domain, called I_w in Fig. 2. Because the extracted vector v was available at the transmitter, the output of the informed embedding was denoted by $x = f(w, v, \alpha, \beta)$, where robust factor α and step factor β are controllable parameters for message codeword w and extracted vector v, respectively. The embedding goal aims to satisfy two conflicting criteria, that is, x must be perceptually indistinguishable to v, and x must also be sufficiently close to w to enhance robustness.

3 Proposed informed embedding algorithm

For the proposed informed embedding scheme, this study used section-based embedding algorithm instead of the informed embedding algorithm of [13]. The four inputs to the embedder were the extracted vectors from the host sequence $v = \{v_1, v_2, \cdots, v_L\}$, the message codeword $w = \{w_1, w_2, \cdots, w_L\}$, and the controllable factors α and β, where the v_k and w_k are vectors of length n with $1 \leq k \leq L$. The parameters α and β control the quality of the watermarked image regarding fidelity and robustness.

The output of the embedder, watermarked sequence x, was subsequently passed through the attack channels, such as the Gaussian noise, JPEG compression, and so on, as illustrated in Fig. 2. The decoder produced the watermark estimate $\widehat{m} = g(y)$, where y is the extracted vector of the received signal after the channel distortion, as illustrated in Fig. 2. The proposed informed embedding algorithm was based on trellis partition. In time k, the extracted vector v_k of n components is one of the a real space of dimension n. The real space of dimension n was partitioned into 2^m regions by a (n, m) linear block codes Γ in each trellis section. We used a simplex as linear block code. The purpose of using the simplex code is to obtain excellent robustness and space partition. Each trellis section is a mapping from the real space to the code space, which is represented by a codeword index set. It is mapped B is mapped as

$$B : R^n \longrightarrow \{c_1, c_2, \cdots, c_{2^m}\} \tag{1}$$

where $\Gamma = \{c_1, c_2, \cdots, c_{2^m}\}$ denotes the set of 2^m disjoint regions. Each region in the partition is associated with a represented codeword. The set of represented codewords is referred to as the object w_k of an extracted vector v_k. In this study, the measure of distortion mean-squared error (MSE) distortion was as follows:

$$d(x_k, w_k) = E\left[|x_k - w_k|^2\right] \tag{2}$$

where x_k is an arbitrary vector over R^n, and w_k is a message codeword in the kth trellis section. In general, the $d(x_k, w_k)$ common choice is the Euclidean distance or hamming distance.

In the proposed informed embedding, we used a convolutional code to construct a trellis; and subsequently, the codewords of a linear block code as the arc labels in the trellis. First, the trellis structure of a binary (n_{out}, k_{out}, ν) convolutional code with 2^ν states in every depth was constructed, where ν is the memory of the convolutional code. Each trellis section contained $2^{\nu+k_{out}}$ arcs, and these arcs were subsequently labeled by all codewords of a linear block code $\Gamma(n, \nu + k_{out}, d)$, where d is the minimal distance of the code. In this structure, the number of bits embedded in each section is represented as k_{out}. With

v as the number of memory and a controllable parameter to tune robustness, the number of trellis state equals 2^v. Parameter v can be adjusted to enhance the robustness, which increases the complexity. In contrast, a convolutional code with a larger value of k_{out} was chosen to increase the embedding capacity. In the proposed trellis structure, a real space of n dimensions was divided into $k_{out} + v$ blocks in each section corresponding to a simplex codeword. With the outer codes as the convolutional codes, the change of k_{out} leads to a simplex code of longer length. Maintaining $k_{out} + v$ constant, that is, n constant, k_{out} was tuned to increase the capacity, and v was tuned to enhance the robustness. Fig. 3 shows the arc labels of the kth trellis section, in which the arc label m_k/w_k denotes the watermark message m_k and the message codeword w_k. This code trellis was obtained from a $(2, 1, 2)$ convolutional code, and the labels of the trellis arcs are the codewords of a $(7, 3, 4)$ simplex code. Using the trellis structure in Fig. 3 can obtain an adequate space partition for n dimension real space.

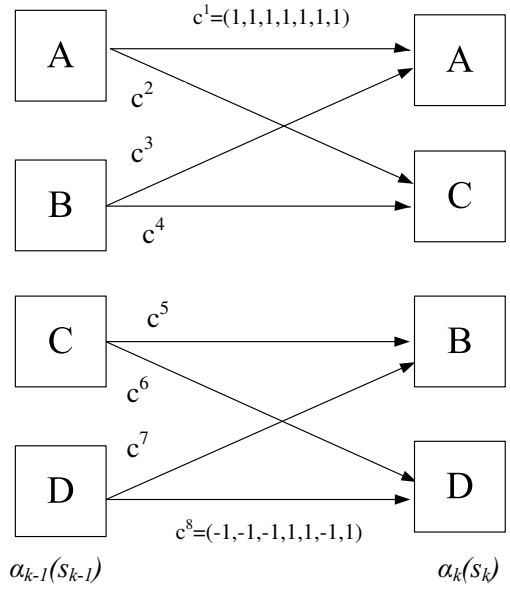

Figure 3: Trellis with eight arcs labelled by a $(7, 3, 4)$ simplex code.

The following section demonstrates the two objectives. The first is to minimize the distortion of the changes by using the quantizer. The second is to minimize the amplitude of stego by using the proposed embedding algorithm.

3.1 Minimizing the distortion in the binary domain

Researchers developed numerous embedding algorithms for minimizing the changes in binary embedding [16, 17, 18]. The subsection presents a discussion on minimizing the distortion of the changes of the watermarking vector w. We first quantize the extracted vector v and watermarking vector w into binary symbols as $Q_2(v)$ and $Q_2(w)$, respectively. To minimize the distortion of the changes of watermarking vector in the binary domain as follows.

Given a nested block code (C_2, C_1), where $C_2 = (n, k_2)$, $C_1 = (n, k_1)$, $m_2 = n - k_2$, $m_1 = n - k_1$ and $k_1 > k_2$, the embedding rate R_m is $(k_1-k_2)/n = (m_2 - m_1)/n$. The primary purpose of the C_2 code is to find the minimum quantization distortion. For an embedder realized using the nested binary codes, the average weight of the toggle vector $E[w(e_{opt})]$ is estimated by the coset leader of a good fine code C_2. The nested block embedding code is constructed as follows. A (C_2, C_1) nested binary embedding code of length n bits is characterized by the use of a parity check matrix

$$H_2 = \begin{bmatrix} H_1 \\ H_\Delta \end{bmatrix} \tag{3}$$

, where $H_\Delta \in \{0,1\}^{(m_2-m_1)\times n}$ and $H_1 \in \{0,1\}^{m_1\times n}$. H_2 and H_1 are two parity check matrices of binary linear fine codes C_1 and coarse codes C_2, respectively, where C_2 is nested in C_1, that is, $C_2 \subset C_1$. The C_2 code is defined as $C_2 = \{u|H_2u = 0\}$, where the vector $u \in F_2^n$. The set consisting of the vector u, corresponding to the identical s_2, is referred to as the coset of the code C_2, defined as

$$C_2^{s_2} = \{u|H_2u = s_2\} = \{c_2 + e_{opt}|c_2 \in C_2\} \tag{4}$$

where e_{opt} denotes the coset leader which represents the minimal Hamming weight in each coset set. The Voronoi set $V_0 = \{e_{opt,i}|i = 1, \cdots, 2^{m_2}\}$ consists of all the coset leaders $e_{opt,i}$ for each coset. The C_1 is partitioned into $2^{m_2-m_1}$ coset of C_2 as

$$C_1 = \bigcup_{H_2 e_{opt,i}^T \in s_\Delta} C_2 + e_{opt,i},\ i = 0, 1, \cdots, 2^{m_2-m_1}-1, \tag{5}$$

where $s_\Delta = [0 \cdots 0\ s_l]$ and $s_l \in \{0,1\}^{m_2-m_1}$. We employ the nested scheme to realize the embedding algorithm and briefly describe the optimal embedding algorithm by using a nested block codes.

Considering the case where, given a host $Q_2(v) = u$, an optimal stego l' with syndrome s_l,

is about to be determined. We assume the existence of a coset leader vector $e_{opt} = u + l'$, closest to sequences u and l'. The host u and the optimal stego l' are of an optimal error vector e_{opt} with a constraint $E[w(e)] \leq n\delta$ $(0 \leq \delta \leq 0.5)$. To describe the quantizer C_2 to determine the optimal stego vector l', we introduce the module operation.

An arbitrary $u \in F_2^n$ can be quantized by the quantizer C_2, and the optimal quantization error e_u can be expressed as a decoding function, as

$$\begin{aligned} e_u &= f_{opt}(s_u) \\ &= f_{opt}(Hu) \\ &= \arg\min_{\hat{u}\in C_2} d(\hat{u}, u) + u \\ &\triangleq u \bmod C_2, \end{aligned} \tag{6}$$

where $f_{opt}(\cdot)$ represents the maximum likelihood (ML) decoding for the quantizer C_2. Determined through ML decoding, the optimal quantization error e_u is added to u to recover the codeword $c \in C_2$, which is closest to the vector u. We further illustrate the quantizer C_2 as C_2's coset, $C_2^{s_\Delta} = C_2 + e_{opt,i} \subset C_1$, where $H_2 e_{opt,i}^T \in s_\Delta$. An arbitrary host vector $u \in F_2^n$ is quantized using $C_2^{s_\Delta}$ as

$$\begin{aligned} e_u &= u \bmod C_2^{s_\Delta} \\ &= u + l \bmod (C_2^{s_\Delta} + l) \\ &= x \bmod C_2 \end{aligned} \tag{7}$$

where the $l \in C_2^{s_\Delta}$ and $x \in F_2^n$.

We offer a low bound D_{bound} to explain that the D_{opt} is limited under the bound D_{bound}. We describe the useful bound for a C_2 code as follows. A $C_2(n, k_2, \lambda_{min})$ code is capable of correcting $t = \lfloor(\lambda_{min} - 1)/2\rfloor$ number of bits, thus a standard array of size $2^{m_2 - m_1} \times 2^k$ can be built in Fig. 4. Alternatively, the required coset leader can be precisely determined to perform binary data embedding, known as optimal embedding. Upon locating all the $\binom{n}{t}$ sequences in the coset leaders, the remaining are of a weight larger than $t + 1$. However, we assume such weight to be identical to $t + 1$, leading to a code referred to as the quasi-perfect code. The average Hamming code weight of the coset leaders within a standard array is given as follows.

$$D_{bound} = \frac{\sum_{i=0}^{t} i\binom{n}{i} + (t+1)(2^{n-k_2} - \sum_{i=0}^{t}\binom{n}{i})}{2^{n-k_2}} \tag{8}$$

The average block distortion D_{opt} of suboptimal or optimal decoding of arbitrary linear code is higher than D_{bound}. However, in the case of a (n, k) perfect linear code, the preceding equation can be expressed as $D_{opt} = (\sum_{i=0}^{t} i\binom{n}{i})/2^{n-k}$.

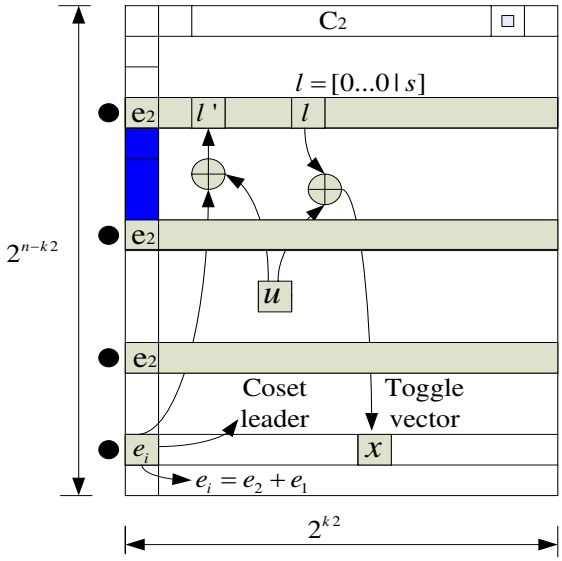

Figure 4: Embedding procedure.

We illustrate an embedding scheme with a standard array for an embedding quantizer. The quantization module in the embedding module attempts to determine the optimal toggle vector e_{opt}. To determine the optimal toggle vector e_{opt}, we use a standard array to explain the embedding procedures. A standard array (Fig. 4) contains. There exists a host corresponding to an arbitrary vector $u \in F_2^n$ of length n bits in the standard array. The syndrome $s_u = Hu^T$ is referred to as the host syndrome. A known binary vector $s_l' = (0 \cdots 0\ s_l)$ of length m_2 bits is intended for embedding. The coset leader $e_{opt} \in F_2^n$ is discovered within a set $C_2^{s_x}$ before a sequence, closest to u with syndrome s_l'. The syndrome s_x is determined by adding the logo vector s_l' to s_u. From the decoding view-point, the coset leader e_{opt} can be discovered through a decoding function, expressed as

$$\begin{aligned} e_{opt} &= f_{opt}(s_x) \\ &= f_{opt}(s_u + s_l') \\ &= f_{opt}(Hu^T + Hl^T) \\ &= f_{opt}(H(u^T + l^T)) \\ &= f_{opt}(Hx^T), \end{aligned} \tag{9}$$

where the stego vector $l = H^{-1}s_l'$ and the notation $f(\cdot)$ are referred to as the ML decoding function. Suppose the existence of a vector $x \in C_2^{s_x}$, which satisfies $s_x = Hx^T$ and represents a coset $C_2^{s_x}$ of the code C_2, which is intended to seek x with minimal weight, that is, e_{opt}, which is expressed as

$$\begin{aligned} e_{opt} &= x + \arg\min_{c\in C_2} d_H(c, x) \\ &= x + Q_2(x) \\ &= x \bmod C_2 \end{aligned} \tag{10}$$

The above formula expresses the third step in the embedding procedures in Fig. 4. Once discovered, the coset leader e_{opt} is added to the host as u, $l' = u+e_{opt}$. $l' \in C_2^{s_l'} \subset C_1$ is the sequence closest to the sequence u within F_2^n dimensional space, and contains the logo sequence s_l'.

Although we adopt a good C_2 code quantizer for optimal embedding, the optimal embedding (i.e. ML decoding) leads to high decoding complexity for a sufficiently large C_2 code. A large value of k_2 renders the ML algorithm, combined with a standard array, infeasible when performing a binary embedding. As an uncommon approach, it is only viable for a small value of k_2.

In the receiver, the received signal is

$$y = l' + N, \tag{11}$$

where N is the channel attack. We also obtain the decoder output as

$$\begin{array}{rcl} l' & = & Q_1(y) \\ & = & Q_1(l' + N), \end{array} \tag{12}$$

where $Q_1(\cdot)$ is a ML decoding function. The l' is used to extract the embedding message s_l as

$$s_l = H_\Delta l'^T. \tag{13}$$

Finally, the receiver perfectly obtains the embedded message s_l.

In accordance with the aforementioned, the optimal togo vector e_{opt} (i.e., the coset leader) is requested to be found for a nested course code C_2 and is intended, in the syndrome domain, to solve the equation $H_2 l^T + H_2 u^T = H_2 x^T$, where $(H_2)^{-1}(0, \cdots, 0\ s_l) = l$. We consider the following a simple and straightforward embedding method. Adopting a systematic nested coding with parity check matrix $H_2 = [P\ I]$ in the code domain, the aforementioned equation is identical to $s_x = H_s x = H_s(u + l)$. Given the arbitrary host u and the logo s_l, the toggle vector x can be determined immediately, assuming that $l = (0, \cdots, 0, s_l)$, the front of s_l is padded $(n - m_2 + m_1)$'s zeros to generate the $l \in C_1$, is a solution and $H_2 l^T = (0, \cdots, 0, s_l)^T$. Finally, we obtain the output $x = u+l$ of the embedder illustrated as follows.

Given a $(C_2(8,4,4), C_1(8,6,2))$ nested block embedding code for binary embedding and consider a systematic parity check matrix

$$H_s = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} H_1 \\ H_\delta \end{bmatrix}, \tag{14}$$

where the H_1 and H_δ are size 2×8. Suppose that there exists a logo vector l of length 8 bits within the C_1 code. Regard $s_l = (11)$ as the logo symbol of length 2 bits intended for embedding into a cover $u = (10001111)$, and with 6 number of 0's padded to its left, a vector l of length 0 bits is hence formed as

$$l = (000000 s_l) = (00000011), \tag{15}$$

where $H_\delta l^T = s_l^T = (10)^T$, and the toggle vector x given by

$$x = l+u = (00000011)+(10001111) = (10001100) \tag{16}$$

The vector $x \in F_2^n$, corresponding to the sequence of length 8 bits, can thus be found. For the systematic form reason, a vector $l = (00000011) \in C_1$, corresponding to the syndrome $s_l = (11)$, can be easily determined. Obtaining the toggle vector x does not guarantee an optimal one. To determine the optimal toggle vector e_{opt}, the toggle vector x can be decoded by ML algorithm as follows.

$$\begin{array}{rcl} e_{opt} & = & x + \arg\min_{c \in C_2} d_H(x, c) \\ & = & (10001100) + (10001110) \\ & = & (00000010). \end{array} \tag{17}$$

Finally, the stego vector is obtained as $l' = u + e_{opt} = (10001111) + (00000010) = (10001101)$ and $H_\delta l'^T = (11)^T$.

3.2 Section-based informed embedding (SBIE) algorithm

Although Subsection 3.1 discussed minimizing the distortion of the changes of the watermarked images, the distortion measure of the mean-squared error yielded uncertain minimization. We used the modified informed embedding algorithm to minimize the distance between the extracted vector and the object vector.

Let $w = (w_1, \cdots, w_L)$ from the optimal binary stego sequence $l' = (l'_1, \cdots, l'_L)$, which obtained by minimizing the distortion of the changes, be a valid path of the trellis, encoded from the watermark $m = (m_1, \cdots, m_L)$, and $v = (v_1, \cdots, v_L)$ the extracted vector from the host signal. Each vector $w_k = 2l'_k - 1$, where $l'_k \in \{0,1\}^n$, is a selected object codeword of length n from the fine code C_1 of the nested block codes $\Gamma(C_1, C_2)$. The embedder produces a watermarked sequence $x = \{x_1, x_2, \cdots, x_L\}$ by a section-by-section trellis-based function $x_i = f(w_i, v_i, \alpha, \beta)$, $1 \le i \le L$, where step factor $\beta \in [0,1]$ and robust factor $\alpha \ge 1$. The geometrical interpretation of the

proposed embedding algorithm in the kth section is shown in Fig. 5, in which the kth component of watermarked image was iteratively updated toward to the decoding region of w_k.

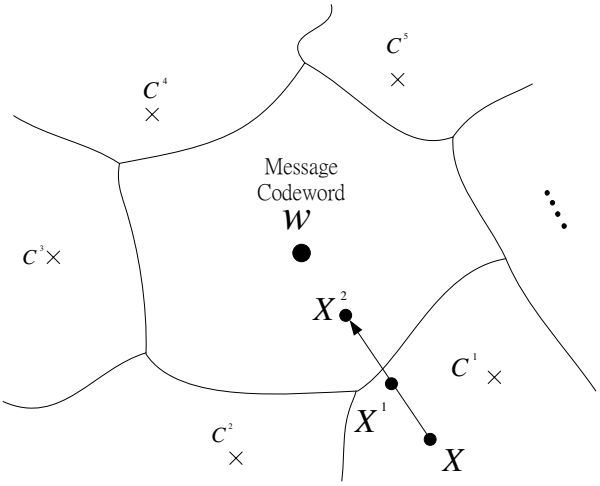

Figure 5: A geometrical interpretation of proposed informed embedding in kth trellis section.

In the kth section of the trellis, we modified the kth component of the extract vector v_k to form the kth component of the watermarked image x_k iteratively. The proposed informed embedding attempts to find x_k to minimize the degradation of x_k from v_k, and simultaneously be closer to αw_k, compared with other candidates αc, $c \in \Gamma$, i.e.,

$$d(\alpha w_k, x_k) \leq d(\alpha c, x_k), \quad c \in \Gamma \text{ and } c \neq w_k, \quad (18)$$

where $d(a, b)$ is the Euclidean distance between a and b. The detailed procedure of finding such x_k is illustrated as follows.

Let h_k be the sign vector between v_k and w_k. That is for each component of v_k and w_k, we define

$$h_{k,i} = \text{sgn}(v_{k,i} \cdot w_{k,i}), \quad 1 \leq i \leq n, \quad (19)$$

where $\text{sgn}(a) = 1$ if $a \geq 0$ and $\text{sgn}(a) = -1$ if $a < 0$. Subsequently, we construct the ith component of x_k as follows: if $h_{k,i} = 1$, then $x_{k,i} = v_{k,i}$, and if $h_{k,i} = -1$, then

$$x_{k,i} = \begin{cases} v_{k,i} - \beta \cdot d(\alpha w_k, v_k), & \text{if } v_{k,i} \geq 0 \\ v_{k,i} + \beta \cdot d(\alpha w_k, v_k), & \text{if } v_{k,i} < 0. \end{cases} \quad (20)$$

In other words, we move v_k toward to w_k by a distance $\beta d(\alpha w_k, v_k)$ for those positions in which v_k and w_k are of opposite signs. If current x_k satisfies (18), we then move on to the $(k+1)$-section, otherwise we substitute v_k by current x_k and repeat the procedures in (19) and (20). The proposed informed embedding causes perceptual degradation of the host signal for distinct α and β, and we can thus adjust the value of α and β to achieve excellent tradeoff between the fidelity and robustness in watermarked images.

The proposed informed embedding algorithm is summarized as follows.

1. Let $k = 1$ and initialize $x_k = v_k$ with a choice of a robust parameter $\alpha \geq 1$ and step parameter $\beta \in [0, 1]$.
2. If the current x_k satisfies the criterion (18), move to Step 4, otherwise substitute v_k by x_k.
3. Update the kth watermarked image x_k by (19) and (20), and move to Step 2.
4. If $k = L$ then terminate, otherwise let $k = k + 1$ and $x_k = v_k$, and move to Step 2.

4 Simulation Results

As shown in [13], a host signal with dimensions $N = 512 \times 512$ was first divided into 4096 blocks of size 8×8; subsequently, each block was converted into the frequency domain using its DCT transform. The first 12 low-frequency AC coefficients in each block, shown in Fig. 2 of [13], were extracted and concatenated, and every $n = 31$ coefficient was subsequently used for embedding each bit of a watermark of $L = 4096 \cdot 12/31 = 1585$-bits. The trellis was constructed by a $(2, 1, 2^{26-m_1}/2)$ convolutional code, and the labels of the trellis arcs were a $(C_2(31, 5), C_1(31, 31 - m_1))$ nested simple code. The nested simple code has an embedding rate $R_m = (26 - m_1)/31$. The parameter m_1 is capable of controlling the tradeoff between watermarking robustness and embedding rate. The experiment can be divided into two sections. We minimize the distortion of the changes of the watermarked images using the algorithm in Subsection 3.1 and then minimize the amplitude of different digital from the extraction using the SBIE algorithm described in Subsection 3.2. Subsequently, the watermarked image quality is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}, \quad (21)$$

where MSE represents the mean square error between the original image I_0 and the watermarked image I_w as

$$\text{MSE} = \frac{1}{\text{N}} \sum_{i=1}^{512} \sum_{j=1}^{512} (I_o(i, j) - I_w(i, j))^2. \quad (22)$$

1. Tradeoff between watermarking robustness and embedding rate over AWGN channel
Considering real-valued x and y, the received pixel y over AWGN channel is given by

$$y = x + g, \tag{23}$$

where g is an additive white Gaussian noise (AWGN), distributed as $N(0, \sigma_g^2)$. Gaussian noise variance σ_g^2 was added to each pixel of the watermarked images. The experiment was repeated for variance σ_g^2, and the bit error rate was computed. The result is shown in Fig. 6.

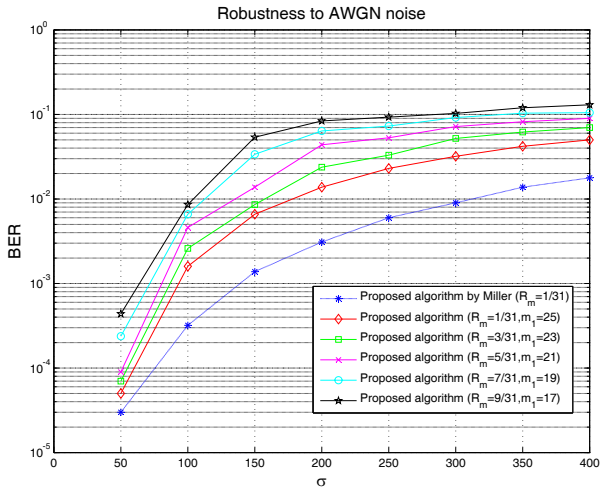

Figure 6: Watermark robustness to AWGN

Fig. 6 shows the results of testing various noise levels with variance ranging from 50 to 400 obtained from the proposed algorithm. The experiment fixed the image fidelity with PSNR $\approx$ 30dB in each case. Although the BER of the proposed algorithm with respect to parameter m_1 is lower than that of Miller[13] under AWGN noise, the embedding capacity of the proposed algorithm is higher than in Miller's work. In the experimental result, the proposed informed embedding with a high embedding rate exhibited inferior BER performance when the controllable parameter m_1 decreased.

2. Performance for parameters α and β
We simulated the fidelity, robustness and complexity by aiming at parameter α and β. The PSNR, as shown in Table 1, is presented as a function of α and increased with α. The parameter α is a constant controlling the embedding strength. We chose α to produce various robustness messages. The degradation in fidelity was measured using MSE distortion. For the large α value, the robustness is greater than for the small α value. Image quality depends on parameter β, the iteration step factor. The higher the value of β , the lower average number of iterations required to reach the expected robustness of the objective codeword, with degrading image quality. Therefore, the value of β can be varied to change the operational complexity when the proposed algorithms are performed.

Table 1: Fidelity experiments with variant α and β

α	0	10
PSNR	29.35	29.33
BER($\sigma = 200$)	$1.2 * 10^{-2}$	$1.35 * 10^{-2}$
β	0.025	0.05
PSNR	29.87	29.82
Number of iteration	8.34	6.32

30	40	50	20
28.38	27.78	26.97	29.02
$2.24 * 10^{-3}$	$8.56 * 10^{-4}$	$12.34 * 10^{-5}$	$0.038 * 10^{-2}$
0.1	0.125	0.15	0.075
28.65	27.71	26.55	29.25
2.43	1.54	1.26	4.58

3. Computational complexity
We compared the algorithm complexity in [13] and that proposed in this paper. The proposed algorithms for minimizing the distortion of the changes of watermarked images (Subsection 3.1) and minimizing the amplitude of the watermarked image (Subsection 3.2) incur major computational complexity. The number of codewords in the trellis section is restricted to the trellis structure of convolutional codes, and the total number of arcs is small. Therefore, the proposed algorithm in Subsection 3.1 easily obtained the optimal codeword candidate and only consumed a number of operational complexities.

For the SBIE algorithm in Subsection 3.2, the Add-Compare-Select (ACS) operation in each section in the memory or the accumulated Viterbi algorithm leads to a more complex embedding algorithm, such as that in Miller's work, compared to that of a memoryless structure of operational complexity. Thus, three complexity parameters in the trellis structure are defined as follows: The decoding process in ours and Miller's algorithms are both based on the Viterbi algorithm. Assuming that there are C_a number of arcs in each trellis section, it is required to calculate the same number of Euclidean distances.

With C_e symbolizing the complexity in evaluating each Euclidean distance, a total complexity of $C_a \times C_e$ is required for each section. In addition, the survival path depends on the ACS operations in each section, and the different informed embedding algorithm yields different number of ACS operations. Consequently, for the same length of the watermarked images, the computational complexity is directly related to the average number of ACS operations in each section. A larger number of ACS operations yields a higher complexity and a longer period to perform the operations. Assuming a trellis with 16 states, each with 2 arcs, the metric accumulated in the previous section pertains to the trellis states and the number of arcs. Because each current state is connected to two arcs, two adders are thus required to perform additions, which necessitates C_a number of adders in each section. Inasmuch as there are two arcs connected to each next state, a comparer is thus required for comparison. In brief, there are 16 next states and 32 arcs in each section, that is, 16 comparers and 32 adders. Hence, the ACS complexity C_s for each section is given as

$$C_s = C_a \times C_e + C_a \times \texttt{adders} + C_a \times \texttt{comparers} \quad (24)$$

As presented in Subsection 3.2, the proposed algorithm is a section-based informed embedding algorithm (i.e., a memoryless informed embedding approach performed independently in each section) that does not require any adder or comparer to perform accumulation operations. However, there are C_a number of comparers required in a search of all arcs for an object message codeword. The resultant complexity is expressed as

$$C_a \times C_e + C_a \times \texttt{comparers} \quad (25)$$

For operational complexity, the proposed algorithm in 3.2 must determine the minimal distance $d(\alpha c_k, x_k)$, regardless of whether the arc operation is closer to the selected codeword αw_k in the trellis section k. The computation required 16 comparer-operations. Finally, we compared with [13] and tabulated as Table 2.

The experimental results in Table 2 confirm that our proposed scheme not only provides high embedding capacity with the adaptive parameter m_1, but also obtains low operational complexity compared to Miller's algorithm.

Table 2: Numbers of operation for proposed algorithm and [13] algorithms

Algorithm	C_s
Proposed SBIE algorithm	$32 \times C_e$+16 comparers
[13]	$32 \times C_e$+ 32 adders+16 comparers

Algorithm	C_t
Proposed SBIE algorithm	$C_s \times 2.9643 \times 1585$
[13]	$C_s \times 72.651 \times 1585$

5 Conclusion

We proposed a modified informed embedding scheme for watermarked images and used a trellis code with a modified trellis structure and nested simplex code to embed messages. These proposed algorithms used nested linear block codewords to label the trellis arcs, and subsequently adjusted the embedding rate and robustness of the watermarked images by using numerous controllable parameters. Although Miller's work offers good bit error rate performance, our experimental results confirm that the proposed algorithm possesses a higher embedding rate and lower complexity than that of Miller's work. Our proposed algorithm provides the two advantages of (i) an adaptive design of watermarked images (i.e., the tradeoff between the BER and the embedding complexity), and (ii) an embedding rate that can be easily altered to meet various application requirements.

Acknowledgements: The research was supported by the National Science Council, Taipei, ROC under Contract NSC-101-2221-E-167-026.

References:

[1] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*, New York: Morgan Kaufmann, 2001.

[2] P. Moulin and R. Koetter, "Data-hiding codes," *Proc. IEEE*, vol. 93, no. 12, pp. 2083–2126, Dec. 2005.

[3] I. J. Cox, M. L. Miller, and A. McKellips, "Watermarking as communications with side information," *Proc. IEEE*, vol. 87, pp. 1127–1141, Jul. 1999.

[4] M. H. M. Costa, "Writing on dirty paper," *IEEE Trans. Inf. Theory*, vol. 29, pp. 439–441, 1993.

[5] Y. Sun, Y. Yang, A. D. Liveris, V. Stankovic, and Z. Xiong, "Near-capacity dirty-paper code design: a source-channel coding approach," *IEEE*

Trans. Inf. Theory, vol. 55, no. 7, pp. 3013–3031, Jul. 2009.

[6] R. Barron, B. Chen, and G. W. Wornell, "The duality between information embedding and source coding with side information and some applicaitons," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1159–1180, May 2003.

[7] S. S. Pradhan, J. Chou, and K. Ramchandran, "Duality bewteen source coding and channel coding and its extension to the side information case," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1181–1203, May 2003.

[8] Chun-Shien Lu, "Towards robust image watermarking: combining content-dependent key, moment normalization, and side-informed embedding," *Signal Processing: Image Communication*, vol. 20, Issue 2, pp. 129–150, Feb. 2005.

[9] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multierminal binning," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1250–1276, Jun. 2002.

[10] M. J. Wainwright, "Sparse graph codes for side information and binning," *IEEE Signal Processing Mag.*, vol. 24, no. 5, pp. 47–57, Sep. 2007.

[11] Claude Desset, Benot Macq, Luc Vandendorpe, "Block error-correcting codes for systems with a very high BER: Theoretical analysis and application to the protection of watermarks," *Signal Processing: Image Communication*, vol. 17, Issue 5, pp. 409–421, May. 2002.

[12] M. L. Miller, I. J. Cox, and J. A. Bloom, "Informed embedding: exploiting image and detector information during watermark insertion," in *IEEE Int. Conf. on Image Processing*, Sep. 2000.

[13] M. L. Miller, G. J. Doerr, and I. J. Cox, "Applying informed coding and embedding to design a robust high-capacity watermark," *IEEE Trans. Image Process.*, vol. 13, no. 6, pp. 792–807, Jun. 2004.

[14] L. Lin, G. Doerr, I. Cox, and M. Miller, "An efficient algorithm for informed embedding of dirty-paper trellis codes for watermarking," in *Proc. IEEE Int. Conf. Image Processing*, Italy, 2005.

[15] VIKAS SAXENA and J.P. GUPTA , "A Novel Watermarking Scheme for JPEG Images", WSEAS Transactions on Signal Processing, Issue 2, Volume 5, pp.74-84, February 2009.

[16] J. J. Wang and H. Chen, "A Suboptimal Embedding Algorithm With Low Complexity for Binary Data Hiding," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 165–168, June. 2012.

[17] J. J. Wang, H. Chen, C.Y. Lin and T. Y. Yang, "An embedding strategy for large payload using convolutional embedding codes," *IEEE International Conference on ITS Telecommunications (ITST)*, pp. 365–369, Nov. 2012.

[18] J. J. Wang, H. Chen, and C. Y. Lin , "An Adaptive Matrix Embedding Technique for Binary Hiding With an Efficient LIAE Algorithm", WSEAS Transactions on Signal Processing, Issue 2, Volume 8, pp.64-75, April 2012.