A Robust Blind Watermarking Scheme based on Distributed Source Coding Principles *

Jim Chou
University of California - Berkeley
319 Cory Hall
Berkeley, CA 94708

Sandeep Pradhan
University of California - Berkeley
319 Cory Hall
Berkeley, CA 94708

Kannan Ramchandran
University of California - Berkeley
269 Cory Hall
Berkeley, CA 94708

ABSTRACT

We propose a powerful new solution to the multimedia watermarking problem by exploiting its duality with another problem for which we have recently made pioneering constructive contributions. This latter problem is that of distributed source coding, or compression of correlated sources that are distributed. We show how these two seemingly unrelated problems are actually duals of each other. We exploit this duality by transforming our recently introduced powerful constructive framework for the distributed compression problem in [13] to a corresponding dual framework for the watermarking problem. Simulations expose the significant performance gains attained by our proposed watermarking approach and reveal its exciting potential for next-generation watermarking techniques. This can be accredited to the exploitation of the dual roles played by source codes and channel codes in the two problems.

Categories and Subject Descriptors

1 [Multimedia Processing and Coding]: Multimedia Security

General Terms

Digital Watermarking, Data Hiding

1. INTRODUCTION

In the wake of the current Information Technology revolution and the resulting proliferation of digital media content, it is difficult to overstate the importance of multimedia security and copyright protection. Not surprisingly therefore, the field of digital watermarking has generated explosive interest recently. The basic idea behind digital watermarking is to seamlessly insert some information (the digital watermark) into the medium or host signal of interest (e.g., MP3 audio, MPEG video, JPEG image, etc.) such that (a) the watermark distorts the host signal minimally (i.e., its presence in the medium is not noticeable), and (b) the watermark can be reliably recovered even if the medium undergoes a certain amount of degradation as a result of both desirable (e.g. compression, signal processing) and undesirable (e.g. malicious attack) reasons. The motivation behind embedding information into the host medium is that if the embedded information can be reliably recovered, then this information can specify the affiliation between the host and its original owner; thus the information must be embedded in a manner that will preclude others from destroying it easily.

Methods for embedding watermarks are wide and varied; popular methods range from modulating the information onto the least significant bits [1] to using the information as a key for indexing pseudo-random noise sequences which are additively combined with the signal (so-called spread-spectrum based methods [7]). Irrespective of the method, the main goal that each method has in common is to embed the maximum amount of information possible given a fixed distortion constraint between the original and watermarked signals (which characterizes the maximum amount of power in the inserted watermark), while allowing for reliable recovery of the embedded information subject to a fixed-distortion attack. However, until recently, the watermarking problem has for the most part been tackled based more on clever tricks and heuristics than on any fundamental theoretical underpinnings (surprisingly, even the popular "spread-spectrum" based watermarking methods turn out to be highly suboptimal [2]).

Fortunately, a recent thread of inspiring research on the field [2, 11, 10] has exposed the fundamental limitations of earlier approaches, and targeted the theoretical foundations for the problem. Inspired by this, practical systems that are grounded on these principles have very recently been proposed [2, 3] which attain significant improvements in the data hiding capacity over previous (and still popular!) classes such as those based on spread spectrum techniques. Our approach to the data hiding problem is in this class of theoretically-inspired next-generation techniques. Our motivation however is unique even in this class, and is inspired by the insight that the watermarking problem is closely connected with, and in fact a dual to, the problem of distributed compression, for which we have recently proposed a powerful

*Sponsored in part by: NSF MIP 97-03161 (CAREER).

constructive framework in [13]. This latter problem is that of source coding with side information (about the source) available only at the receiver. The watermarking problem turns out to be a dual in the sense that it is equivalent to a channel coding problem with side information about the channel available only at the transmitter [11].

This paper is motivated by the goal of exploiting this duality and leveraging our recent insights on a *constructive* framework for the distributed source coding problem [13] to formulate a dual solution to the watermarking problem. It is important to note that our approaches to both problems are rooted in fundamentally sound information-theoretic principles, but unlike information-theoretic results that are non-constructive and asymptotic in nature, our goal is to formulate practical constructions that can lay the foundations for real-world systems.

At the same time, in the context of the watermarking problem, it is important to understand the limitations of our proposed approach related to the assumed distortion metrics for both the watermarking system and the attack channel. In this work, we confine ourselves to Euclidean-space distortions. Our reasons are pragmatic: useful perceptual metrics for media like images and video are severely lacking, while tractable ways of combating more sophisticated attack channels (such as geometric distortion attacks as in the Stirmark freeware package [12]) are difficult to formalize. Nevertheless, we are confident that "good" code constructions which target Euclidean distortion metrics can be leveraged in the construction of "good" codes that target geometric distortions as well, which we propose to do as part of future work.

This paper's primary motivation is to bridge the large existing gap between current watermarking technology and the theoretical limits for watermark transmission for Euclidean distortion metrics. We will show that codes that achieve rate-distortion bounds for distributed source coding can be used to design codes that achieve capacity for digital watermarking. Furthermore, we will provide experimental results based on Gaussian sources and real images with additive white Gaussian noise (AWGN) attacks to unveil the power of this approach.

This paper is organized as follows: In the first section we introduce both the distributed source coding problem and the digital watermarking problem, providing illuminating "toy" examples to illustrate the dualities between the two. In the following section, we generalize the duality between distributed source coding and digital watermarking to the case where the signals are real and Gaussian distributed. In section 4, we use the duality between the two problems to demonstrate how "good" codebooks can be constructed for the digital watermarking problem based on codebooks used for the distributed source coding problem and vice versa. In the final two sections we provide simulation results to demonstrate the power of our approach and to provide some concluding remarks on future work.

2. DUALITY OF DISTRIBUTED SOURCE CODING AND WATERMARKING

In this section we explore the duality between the distributed source coding problem and the watermarking problem.

2.1 Distributed source coding

Source coding with side information is shown schematically in Fig. 1. In this problem, a discrete source X is to be encoded and transmitted to a receiver which has access to some discrete side information Y. Even though the encoder does not have access to Y, using the joint statistics of X and Y, the encoder can compress at the same efficiency as the case when both the encoder and decoder have access to Y. A non-constructive proof of this result was provided by Slepian and Wolf [14]. Pradhan et. al. [13] later provided a constructive algorithm for realizing the results suggested by the proof. To understand how this remarkable result might be realized in practice, we provide the following illustrative example. More detailed constructions can be found in [13]

Example 1: Discrete Case: Consider X and Y to be equiprobable 3-bit data sets which are correlated in the following way: $d_H(X,Y) \leq 1$, where $d_H(.,.)$ denotes Hamming distance. When Y is known both at the encoder and decoder, we can compress X to 2 bits, conveying the information about the uncertainty of X given Y (i.e., the modulo-two sum of X and Y given by: (000),(001),(010) and (100)). Now if Y is known only at the decoder, we can surprisingly still compress X to 2 bits. The method of construction stems from the following argument: if the decoder knows that X=000 or X=111, then it is wasteful to spend any bits to differentiate between the two. In fact, we can group X=000 and X=111 into one coset (it is exactly the principal coset of the length-3 repetition code). In a similar fashion, we can partition the remaining space of 3-bit binary codewords into 3 different cosets with each coset containing the original codewords offset by a unique and correctable error pattern. Since there are 4 cosets, we only need to spend 2 bits to specify the coset in which X belongs. The four cosets are given as

$$\text{coset-1} = (000, 111), \quad \text{coset-2} = (001, 110),$$
$$\text{coset-3} = (010, 101), \quad \text{coset-4} = (011, 110)$$

The decoder can recover X perfectly by decoding Y to the closest (in hamming distance) codeword in the coset specified by the encoder. Thus the encoder does not need to know Y for optimum encoding.

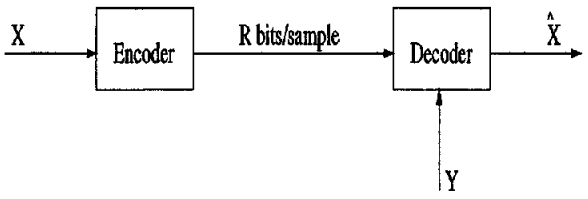

Figure 1: Distributed source coding: only decoder has access to the side information Y.

The above concepts can be generalized to other cases including the encoding/decoding of continuous-valued random variables, where the decoder uses Y to reconstruct the source based on a minimum fidelity criterion. Wyner and Ziv gave a non-constructive proof of the fact that the minimum rate

of encoding [15] for a given fidelity criterion D, is

$$R(D) = min_{\hat{X}=f(U,Y),p(U|X)}[I(U;X) - I(U;Y)] \quad (1)$$

where U is the set of codewords representing X, $I(U;X)$ is the Shannon mutual information [6], and the minimization is carried out over all conditional probability density functions $p(U|X)$ and a function $f(U;Y)$ such that $E(X-\hat{X})^2 \leq D$. Pradhan et. al [13] later provided a constructive algorithm for achieving the rates suggessted by Wyner and Ziv's proof. The main idea of encoding is as follows: (1) build a source code to represent X using nearly $2^{nI(U;X)}$ codewords, (2) partition this set into $2^{nR(D)}$ cosets with each coset containing $2^{nI(U;Y)}$ codewords (the set of codewords act as a channel code for the fictitious channel between U and Y), and (3) find the optimal reconstruction, which is obtained as a function $f(U,Y)$.

To elucidate the encoding construction, consider the case where X is an independent identically distributed (*i.i.d.*) Gaussian random variable with side information Y given by $Y = X + N$ (N is an *i.i.d.* Gaussian random variable independent of X). The side information is therefore a noisy version of X and as in the discrete case, it has been shown [15] that an encoder can be designed to represent X as well as the case where the encoder also has access to Y. The practical method proposed in [13] is to design a source code and partition it into a bank of cosets of channel codes. The source code is designed for the optimal representation of X, and is partitioned into cosets which have good distance properties. The source X is quantized to a codeword (referred to as the active codeword) and the index of the coset containing this codeword is sent to the decoder. With the help of Y, the decoder finds the active codeword in the coset whose index is given by the encoder. A practical example is as follows:

Example 2: Continuous Case: Consider an 8-level fixed-length scalar quantizer as the source coder (see Fig. 2), with the rate of transmission fixed at 1 bit/source sample. To

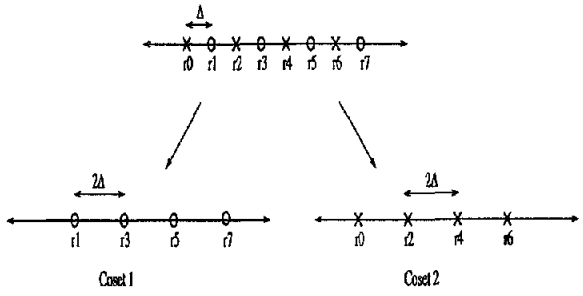

Figure 2: An 8-level quantizer partitioned into 2 cosets containing 4 levels each.

partition the source code, we partition the reconstruction levels into two cosets. The reconstruction levels in coset 1 are denoted as (r_0, r_2, r_4, r_6) and the reconstruction levels in coset 2 are (r_1, r_3, r_5, r_7) (see Fig. 2). The partition is constructed in such a way that the cosets are symmetric and the minimum distance between any two elements within a coset is kept as large as possible. Upon encoding, X is quantized to a codeword in the composite 8-level quantizer and the index of the coset containing this codeword is sent to the decoder. The decoder finds this codeword in the coset whose index is sent by the encoder, as the one which is closest (in the appropriate distance measure) to Y. In doing so, the decoder can reconstruct the value of X to a codeword which is "close" in terms of squared-error distortion.

In the above example, a simple 8-level scalar quantizer was considered for the sake of simplicity. Encoding constructions, however, can be generalized to more sophisticated source coders. For example, the partition can be done in higher-dimensional spaces using trellis codes. This was proposed in [13] where the n-dimensional product space of scalar quantized reconstruction levels is partitioned into cosets of trellis coded quantizer codebooks.

2.2 Watermarking problem

The digital watermarking problem can be formulated as follows. The encoder has access to two signals; the information (an index set), M, to be embedded, and the signal (host) that the information is to be embedded in. The output of the encoder is the watermarked signal. The attacker will attempt to degrade the signal so that the decoder (who wants to authenticate the watermarked signal) will fail to decode the watermark. The attacker has a distortion constraint on the amount of degradation that he can inflict on the signal. The encoder has to be designed such that the attacker needs to inflict an amount of distortion (to destroy the watermark) that will render the signal to be useless. Mathematically, the goal is to solve the following constrained minimization problem:

$$min_{||W-S||^2 \leq D_1, ||Y-W||^2 \leq D_2} P_e(\hat{M}) \quad (2)$$

where $P_e(\hat{M})$ represents the probability of decoding error.

It was shown in [2, 11] that this problem can be viewed as channel coding with side information (about the channel) at the encoder (see fig. 3). We will consider the case when the

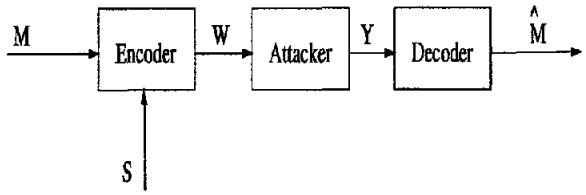

Figure 3: The watermarking problem: the encoder has access to a signal S. M represents the information to be embedded and W represents the watermarked signal.

side information is jointly Gaussian with the noise source. In this case, the attacker is considered to be optimal with respect to his own interests [11] (up to a scaling factor) and the problem becomes the regular AWGN communications problem if both the encoder and decoder have access to the side information Surprisingly, in the case when only the encoder has access to the side information, it has been shown that

this system can *perform as well as when both the encoder and decoder have access to the side information [5].* In fact, this is the key insight which motivates our analysis for the digital watermarking system through exploitation of its dual nature with the distributed source coding problem. Before formulating the details for constructing an encoder/decoder for the watermarking problem, we present an example of channel coding with side-information at the encoder to illustrate key concepts. For this we turn to the problem of writing on defective memories [9], which, like watermarking, can be viewed as channel coding with side-information at the encoder. In this problem the encoder attempts to write data onto faulty registers with one of the registers always being "stuck-at" 0 or 1. The encoder, however, has full knowledge of which register is always "stuck". If the decoder also has this knowledge, then $n-1$ bits can be stored reliably in n registers with one "stuck-at" fault. It was shown in [9] that the encoder can be designed to perform as well as the case when both the encoder and the decoder have access to the "stuck-at" position. We illustrate the method for constructing such an encoder in the following example (which is the dual to Example 1)[1].

Example3: Faulty Registers: Consider a 3-bit memory device, which has one "stuck-at" in any position with a uniform probability of the "stuck-at" being either a one or a zero. When both the encoder and the decoder know the value and the position of the "stuck-at", the encoder can write 2 bits reliably. Now consider the case when only the encoder has access to the "stuck-at". Let $\mathcal{A}$ denote the the set of binary three tuples: $\{0,1\}^3$. We partition $\mathcal{A}$ into cosets of codewords which are compatible with any type of "stuck-at". This partition is the same as the one considered in Example 1. If the encoder wants to write the first message, he chooses a codeword from the first coset which is compatible with the "stuck-at". If the first bit has a "stuck-at" fault of 1, then he chooses [1 1 1] as the codeword to be written on the memory. In this fashion, we can again reliably write two bits (corresponding to the index of the coset), even though the decoder does not have access to the "stuck-at".

The above problem can be generalized into the problem of channel coding with side information, for which the capacity can be calculated. The information-theoretic capacity [9, 8, 5] of such systems is given by

$$C = max_{p(U,S|X)}[I(U;Y) - I(U;S)] \tag{3}$$

where the maximization is over all conditional probability density functions $p(U,X|S)$. The signal S is the side information about the channel and U represents the codeword space. The main steps for encoding are as follows: (1) build a channel code over the space of U, with the number of codewords being nearly equal to $2^{nI(U;Y)}$ (where n is the block length of encoding), (2) partition this channel codeword space into cosets of source codes with each coset containing nearly $2^{nI(U;S)}$ codewords, and (3) choose a codeword, U, to represent S from the coset which has an index equal to the message. The size of the index set should be nearly equal to 2^{nC}, where C is given in (3). The signal, X, which is transmitted over the channel is a function $f(U,S)$.

[1]We use the term "stuck-at" loosely to refer to a faulty memory register which is constrained to always be a 1 or a 0.

Returning to the watermarking problem, we consider the case where the channel is AWGN (see Fig. 3) and the signal (side information, S) is *i.i.d.* Gaussian. In this case, it was shown by [5] that the capacity (3) is given as

$$C = \frac{1}{2}log\left\{\frac{P}{N}+1\right\} \tag{4}$$

where P and N represent the transmitter power constraint and the variance of the channel noise respectively. To achieve capacity, Costa [5] showed that the space of codewords must be of the form $U = X + \alpha S$, where $\alpha = \frac{P}{P+N}$ and N is the variance of the attacker's noise. The codeword U should be chosen in a way that will ensure that the projection of U along the direction orthogonal to S has power P.

The watermarking problem can be posed as a version of the above problem, where the signal (i.e., image, audio, etc.) is given by S, the attack is AWGN (with a distortion constraint of N) and the distortion constraint between the signal and the watermarked signal is given as P. In doing so, the above procedure is reinterpreted as follows: the codeword U is chosen to be a quantized representation of some scaled version of S such that the quantization noise is orthogonal to U (see Fig. 4). This scaling factor can be obtained by the geometrical visualization of the encoding process suggested by [5]. Using geometry, the scaling factor is solved to be

$$\beta S = U + V \tag{5}$$

where V denotes the quantization noise and β is such that U is the "ideal" rate-distortion-quantized version of βS with $\beta = \frac{P+\alpha^2 Q}{\alpha Q}$. As a result, we have a constructive approach for encoding watermarks into a given signal. The encoding process is summarized as follows: (1) pick a channel code over the space of U, (2) partition the channel code into cosets of source codewords with each codeword representing a quantized version of βS and (3) choose the index of the coset in which the signal is to be quantized based on the watermark that is to be encoded into the signal.

Upon decoding, the decoder finds a codeword, γU in the composite channel code (containing nearly $2^{nI(U;Y)}$) which is closest to the received vector (in some sense) where $\gamma = \frac{P+\alpha Q}{P+\alpha^2 Q}$. The coset containing the decoded codeword is declared as the decoded watermark message. It can be shown [5] that the scaling which is done at the encoder provides the required distance property to tolerate the attacker's noise at the decoder.

3. SUMMARY OF DUALITY

The duality between distributed source coding with side information at the decoder and channel coding with side information at the encoder is best illustrated by a set of diagrams. For a more rigorous treatment of the duality and achievable rates, the reader is referred to [4].

In distributed source coding, the set of possible messages to be encoded will roughly lie on a hyper-sphere of L dimensions for L large. The encoding operation then entails

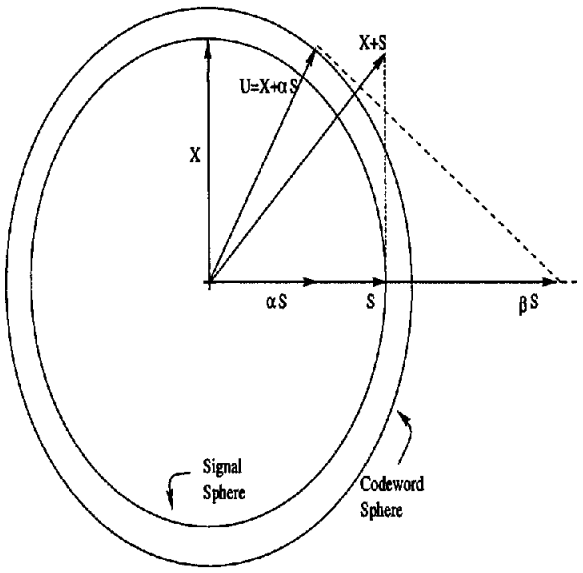

Figure 4: Geometric interpretation of the optimal encoding: U and S denote the codeword chosen for encoding and the signal to be watermarked respectively. This can also be interpreted as choosing that U which is the quantized version of βS as shown. The watermark added onto the signal S is X.

finding a codeword on the codeword sphere (see Fig. 5) that is within the expected distortion to the message point. The selected codeword will belong to one of R cosets, where R is the achievable rate, and the index of the coset will be the encoded message. One will easily notice that the encoding operation for distributed source coding with side-information at the decoder can also be viewed as the decoding operation for channel coding with side-information at the encoder. Using Fig. 5, we see that the message sphere can be viewed as the received signal sphere in the case of channel coding with side information at the encoder. The decoder will then find the codeword on the codeword sphere which is closest (in some sense) to the received signal point. The coset to which the decoded codeword belongs will represent the received message (in the watermarking context, the received message will represent the watermark).

The decoding operation for distributed source coding can be understood with the aid of Fig. 6. The decoder will have both the side-information and the index of the coset available to it. The side-information will roughly lie on a hypersphere given that the dimension is large. To reconstruct the encoded message, the decoder will search through the set of codewords in the coset specified by the encoder, and find the nearest codeword (in some sense). Upon finding the "correct" codeword, the message can be reconstructed as a linear combination of the decoded codeword and the side-information [4]. As to be expected, the encoding operation for channel coding with side information at the encoder parallels the decoding operation for distributed source coding with side-information at the decoder. In the channel coding case, the encoder has both the side-information and the index of the coset (message to be sent) available to it. Again, the side-information will roughly lie on a hyper-sphere and the encoder will find the codeword in the coset specified by the message that is nearest to the side-information (in some sense) (see also Fig. 4). The encoder will then transmit a linear combination of the codeword and the side-information to the decoder.

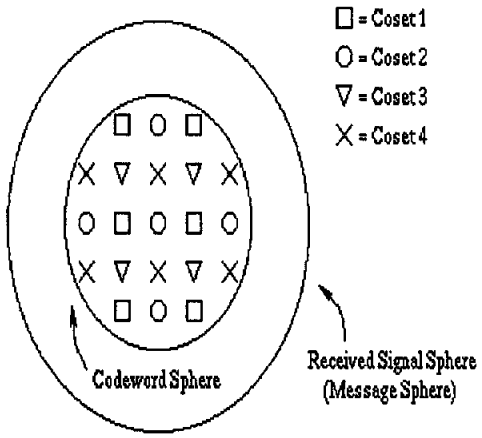

Figure 5: Geometric interpretation of the decoding (encoding) operation for data hiding (distributed source coding) with side information at the encoder (decoder).

The purpose of the above discussion was to illustrate the duality between distributed source coding with side-information at the decoder and channel coding with side-information at the encoder, so that key concepts could be illustrated to inspire "good" codebook constructions. As a result, there is not much rigor in the above discussion. A more rigorous proof of the duality between distributed source coding with side-information at the decoder and channel coding with side-information at the encoder can be found in [4].

4. DESIGN AND CONSTRUCTION

Inspired by the previous section and our earlier work in [13], we can now construct practical solutions to the watermarking problem. We will look at several different codebook constructions to illustrate the necessities in designing a codebook that introduces minimal distortion to the host signal but is robust to the attacker's "noise".

4.1 Codebook design

As mentioned above, the encoding entails taking a channel code and partitioning it into cosets of source codes. Thus, in order to design a good codebook, one needs to use a "good" channel code which can be partitioned into "good" source codes. To illustrate, we will consider three different codebook constructions. The first codebook construction uses Pulse Amplitude Modulation (PAM) as the composite channel code, and partitions the channel code into source code-

words given by a scalar quantizer (SQ); we will refer to this method as PAM-SQ. The next construction uses PAM as the composite channel code, and partitions the channel code into codewords given by a trellis-coded quantizer (TCQ); we refer to this method as PAM-TCQ. And in the final construction we use TCM as the composite channel code, but partition the channel code into source codewords given by a trellis-coded-quantizer; we refer to this method as TCM-TCQ.

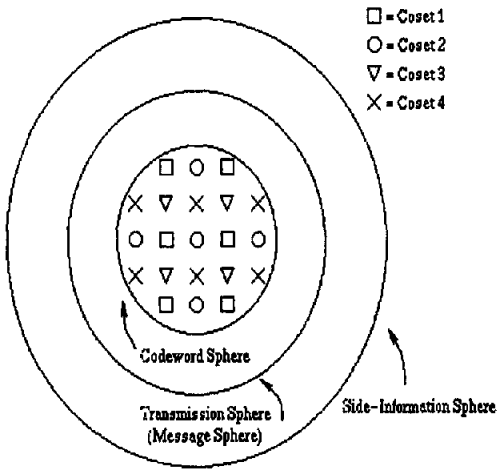

Figure 6: Geometric interpretation of the encoding (decoding) operation for data hiding (distributed source coding) with side information at the encoder (decoder).

The PAM-SQ codebook can be constructed as the product space of 1-dimensional scalar quantizers. The cosets are divided as in Example 2. In this scenario, we can easily calculate the average distortion per sample to be:

$$D = \frac{(2\Delta)^2}{12} = \frac{4\Delta^2}{12} \tag{6}$$

where Δ represents the step-size of the scalar-quantizer. The minimum distance between codewords will be $d_{min} = \Delta$ and the probability of error per dimension is approximately

$$p = Q(\sqrt{\frac{\Delta^2}{2N_o}}) \tag{7}$$

Thus, the probability of decoding a wrong watermark is:

$$P_e = 1 - (1-p)^L \tag{8}$$

where L is the number of watermark bits and assuming that the quantization error is uniformly distributed across each quantization bin and that the attack channel can be modeled as additive white Gaussian noise with variance $N_o/2$. Next, using PAM-TCQ, we can introduce less distortion to the host signal while maintaining the same probability of error. This is a direct result of using a better source codebook (i.e., TCQ). Now, if we use a better channel codebook, we can also decrease the probability of error. Specifically, consider a TCM-TCQ codebook which is generated as a cascade of a rate-2/3 convolutional code and a rate-3/4 convolutional code (see Fig. 7). A scaled version of the host signal is quantized to a codeword in the coset specified by the watermark. Each coset will consist of a TCQ codebook generated by the composite rate-2/4 trellis. The minimum euclidean distance between codewords will be equivalent to the free euclidean distance of the rate-3/4 trellis code. As a result, the probability of decoding the wrong watermark will be approximately:

$$P_e = N_{free} Q(\sqrt{\frac{d^2_{free}}{2N_o}}) \tag{9}$$

where N_{free} represents the number of paths starting at one

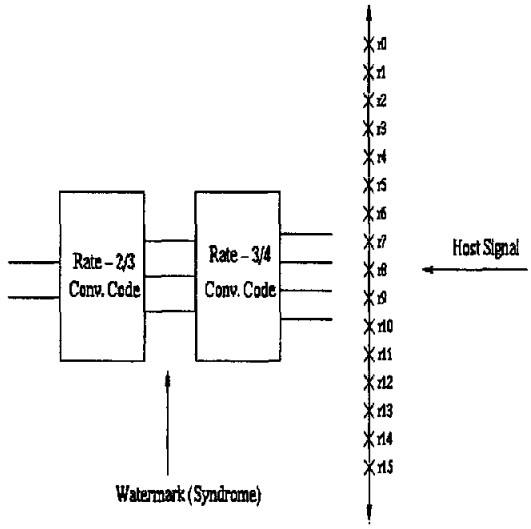

Figure 7: A method for constructing the TCM-TCQ codebook for encoding digital watermarks.

node and ending at the same node with distance d_{free}. The probability of error achieved by TCM-TCQ will in general be lower than either PAM-SQ or PAM-TCQ for a given distortion to the host signal.

From the above analysis, we can see that by choosing good source codebooks, we can limit the distortion introduced to the host signal. On the other hand, by choosing good channel codebooks, we can make the watermark more resistant to attacks. Hence, to design a good watermarking scheme requires a careful choice of source and channel codebooks. In general there are two methods of improving the codebook. The first method involves a direct application of Forward Error Correction (FEC) codes to the watermark message. In general, this has the effect of increasing the euclidean distance between codewords, but has the disadvantage of decreasing the watermark throughput. The other method of improving the codebook is to design a better code in the euclidean space. This will also decrease the probability of

error, but has the advantage of maintaining the watermark throughput.

4.2 Algormithm

We now have a means to formulate a practical digital watermarking algorithm. The steps for encoding are as follows:

(1) Design an appropriate codebook (see Section 4.1)
(2) Scale the host signal by β (see Section 2.2)
(3) Determine the codeword, U, in the coset indexed by the watermark message which is closest to the scaled host signal
(4) Transmit a linear combination of the codeword and host signal.

Similarly, the decoding operation can be enumerated as follows:

(1) Scale the received signal by γ (see Section 2.2)
(2) Find the closest codeword, U, to the scaled received signal
(3) Declare the coset that the codeword lies in as the decoded watermark message.

5. PERFORMANCE

To test our watermarking constructions, we have run simulations on the PAM-TCQ and TCM-TCQ cases. The first set of simulations that we conducted, assumed that the signal was Gaussian and that the attack was AWGN. We fixed the probability of decoding error to be less than 10^{-5} and determined the amount of distortion that was necessary to ensure this probability of error given a fixed-distortion attack. The performance curves representing the signal-to-distortion (i.e., signal power vs. watermark power) ratio vs. signal-to-noise (i.e., signal power vs. noise power) at $P_e(\hat{M}) = 10^{-5}$ and a rate of 1 embedded-bit/sample is given in Fig. 8. It can be seen from the figure that TCM-TCQ outperforms PAM-TCQ significantly. The reason for this performance discrepancy is that TCM-TCQ uses a better channel code than PAM-TCQ.

The next simulation that we conducted is of a more practical use. We consider the case where S is the 'McKinley' image and the attack is confined to JPEG compression. We again tested both PAM-TCQ and TCM-TCQ. At an embedding rate of $\frac{1}{64}$ bits/sample, PAM-TCQ incurred a Peak-Signal-to-Distortion-Ratio (PSDR) of 43.48dB in order to withstand a JPEG compression factor of 75%. On the other hand, TCM-TCQ incurred a PSDR of 44.12dB in order to withstand a JPEG compression quality factor of 55%. Again TCM-TCQ outperforms PAM-TCQ. The above experiments were performed assuming that large amounts of data needed (1 bit/64 samples) to be hidden in the host signal. In various watermarking applications, however, often only a few bits need to be hidden in a given image. To alter our proposed solution to operate at lower embedding rates, we can use an outer channel code to code the syndrome. In the interest of demonstrating the power of our approach, we tried the extreme case of using a length 4096 trivial repetition code, to reduce the embedding rate to 1 bit/ image (512x512). For such a case, both the PAM-TCQ and the TCM-TCQ solutions were able to admit JPEG attacks of (1%) and still recover the one-bit watermark successfully. In Fig. 9 we show the watermarked McKinley image, and in

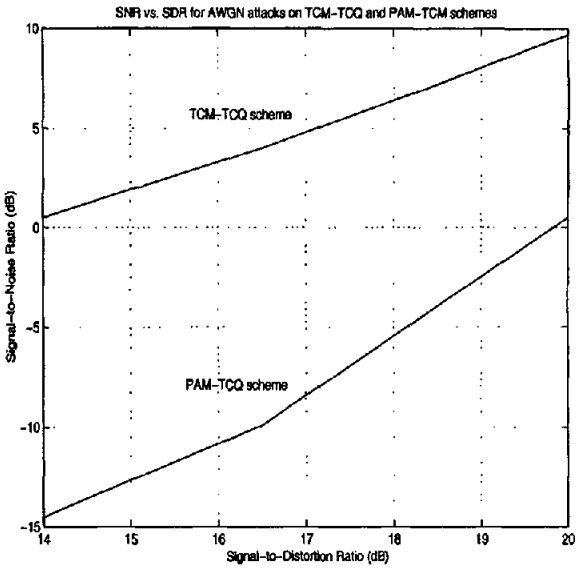

Figure 8: Performance curves of PAM-TCQ and TCM-TCQ given fixed distortion attacks.

Fig. 10 we show the attacked McKinley (Q=1%) image, in which we were able to successfully recover the watermarked bit.

Figure 9: Watermarked McKinley image.

For the above experiments we used relatively simple source and channel codes to achieve our performance. We can use more sophisticated source and channel codes in our framework to attain even better performance. For example, we can use turbo-coded modulation to improve our channel coding performance, and we can go to higher dimensional channel codes (coded QAM, coded n-dimensional simplex signals, etc.) to lower our rates. Furthermore, we can use

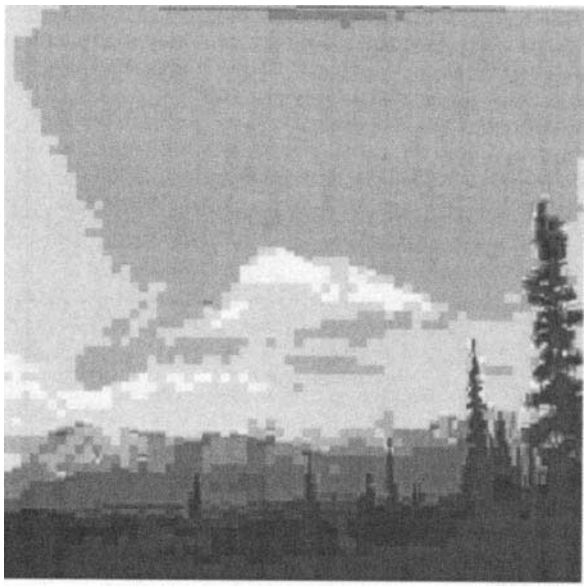

Figure 10: Degraded McKinley image (JPEG compressed, Q=1%). The watermarked bit was successfully recovered.

vector quantization to improve our source codebook.

6. CONCLUSION

In conclusion, we have shown that the watermarking problem and the distributed source coding problem can be viewed as duals of each other, where the former is channel coding with side information at the transmitter and the latter is source coding with side information at the receiver. In the first case, the channel code is partitioned into a bank of cosets of source codes, while in the latter the source code is partitioned into a bank of cosets of channel codes. The two 3-bit binary data examples clearly illustrate the dual nature of the two problems. Through simulation results, it can be seen that with better source and channel codes, our proposed solution will push towards capacity. Furthermore, our solution is easily amenable to codes built for higher-dimensions [13].

As stated in the beginning, all of the results in this paper, targeted Euclidean-space attacks. Current work is in progress to integrate our approach with other methods to address geometric attacks to provide for a more complete watermarking solution.

7. REFERENCES

[1] J. M. Barton. Method and apparatus for embedding authentication information within digital data. *United States Patent #5,646,997*, Issued July 8 1997.

[2] B. Chen and G. W. Wornell. Preprocessed and postprocessed quantization index modulation methods for digital watermarking. *Proc. SPIE Security and Watermarking Multimedia Contents*, 3971, Jan 2000.

[3] J. Chou, S. Pradhan, L. El Ghaoui, and K. Ramchandran. A robust optimization solution to the data hiding problem based on distributed source coding principles. *Proc. of SPIE*, January 2000.

[4] J. Chou, S. S. Pradhan, and K. Ramchandran. On the duality between distributed source coding and data hiding. *preprint*, June 2000.

[5] M. Costa. Writing on dirty paper. *IEEE Trans. on Information Theory*, 29:439–441, May 1983.

[6] T. M. Cover and J. A. Thomas. *Elements of Information theory*. Wiley, New York, 1991.

[7] I. Cox, J. Killian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. on Image Processing*, 6(12):1673–1687, December 1997.

[8] S. Gel'fand and M. Pinsker. Coding for channel with random parameters. *Problems of Control and Information Theory*, 9:19–31, 1980.

[9] C. Heegard and A. El Gamal. On the capacity of computer memory with defects. *IEEE Trans. on Information Theory*, 29:731–739, September 1983.

[10] P. Moulin. The role of information theory in watermarking and its application to image watermarking. *Preprint*, Mar 2000.

[11] P. Moulin and J. O'Sullivan. Information-theoretic analysis of information hiding. *Preprint*, Mar 2000.

[12] F. Petitcolas and M. Kuhn. Stirmark software.

[13] S. S. Pradhan and K. Ramchandran. Distributed source coding using syndromes: Design and construction. *Proceedings of the Data Comression Conference (DCC)*, March 1999.

[14] D. Slepian and J. K. Wolf. Noiseless encoding of correlated information sources. *IEEE Trans. on Inform. Theory*, IT-19:471–480, July 1973.

[15] A. D. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Trans. on Inform. Theory*, IT-22:1–10, January 1976.