Robust Image-Adaptive Data Hiding using Erasure and Error Correction

K. Solanki, N. Jacobsen, U. Madhow, B. S. Manjunath*, and S. Chandrasekaran
Dept. of Electrical and Computer Engineering
University of California at Santa Barbara
Santa Barbara, CA 93106
*Corresponding author: Ph (805) 893-7112
E-mail: manj@ece.ucsb.edu
EDICS : 5-AUTH

***Abstract*— Information-theoretic analyses for data hiding prescribe embedding the hidden data in the choice of quantizer for the host data. In this paper, we propose practical realizations of this prescription for data hiding in images, with a view to hiding large volumes of data with low perceptual degradation. The hidden data can be recovered reliably under attacks such as compression, and limited amounts of image tampering and image resizing. The three main findings are as follows:**
(i) In order to limit perceivable distortion while hiding large amounts of data, hiding schemes must use image-adaptive criteria in addition to statistical criteria based on information theory.
(ii) The use of local criteria to choose where to hide data can potentially cause desynchronization of the encoder and decoder. This synchronization problem is solved by the use of powerful, but simple to implement, erasures and errors correcting codes, which also provide robustness against a variety of attacks.
(iii) For simplicity, scalar quantization based hiding is employed, even though information-theoretic guidelines prescribe vector quantization based methods. However, an information-theoretic analysis for an idealized model is provided to show that scalar quantization based hiding incurs approximately only a 2 dB penalty in terms of resilience to attack.

I. Introduction

The past decade has witnessed a surge of research activity in multimedia information hiding, targeting applications such as steganography (or covert communication), digital rights management, and document authentication. Another important class of applications is the seamless upgrade of communication or storage systems: additional data and meta-content can be hidden in existing data streams, such that upgraded receivers can decode both the original and the hidden data, while existing receivers can still decode the original data. Several techniques have been proposed in the literature that hide information in images and video in a robust and transparent fashion (for comprehensive surveys, see [1], [2], [3]). Much of this activity is geared towards the application of digital rights management, with a focus on devising digital watermarks that are robust to malicious attacks that aim to remove the watermark while preserving the content quality. A number of freeware packages for such attacks are available, such as Stirmark [4], which employ geometric distortions such as random bending, rotation, scaling, translation, and cropping. A number of recent efforts in data hiding focus, therefore, on devising watermarks that survive such attacks (see, for example, [5], [6]). Another potential adversary for the data hider is the steganalyst, who tries to detect the presence of hidden data. Thus, there are significant research efforts both in steganalysis ([7], [8]), and on hiding in a manner that is difficult to detect ([9], [10]).

In this paper, we propose a framework for hiding large volumes of data in images while incurring minimal perceptual degradation. Our work differs from the preceding literature in several ways. First, we seek to embed much larger volumes of data than required for watermarking, targeting applications such as steganography and seamless upgrade of communication and storage systems, rather than digital rights management. Second, because of our target applications, we aim for robustness not against malicious attacks such as Stirmark's geometric attacks, but against "natural" attacks such as compression (e.g., a digital image with hidden content may be compressed as it changes hands, or as it goes over a low bandwidth link in a wireless network). It turns out, however, that our schemes are actually robust against a broader class of attacks than we initially designed for, such as tampering, and a limited amount of resizing. The hiding methods we use are guided by the growing literature on the information theory of data hiding (summarized in the next paragraph), but are adapted to the specific application of hiding in images.

Information-theoretic treatments of the data hiding problem typically focus on hiding in independent and identically distributed (i.i.d.) Gaussian host samples. The hider is allowed to induce a mean squared error of at most D_1, while an attacker operating on the host with the hidden data is allowed to induce a mean squared error of at most D_2. Information-theoretic prescriptions in this context translate, roughly speaking, to hiding data by means of the choice of the vector quantizer for the host data, with the AWGN attack being the worst-case under certain assumptions. This method of hiding was first considered by Costa [11], based on results of Gel'fand and Pinsker [12] on coding with side information (with the host data playing the role of side information). Game-theoretic analyses of data hiding, with the hider and attacker as adversaries, have been provided by Moulin and O'Sullivan [13], and by Cohen and Lapidoth [14]. Estimates of the hiding capacity of an image, based on a parallel Gaussian model in the transform domain, have been provided by Moulin and Mihcak [15]. Chen and Wornell [16]

This research is supported in part by a grant from ONR # N00014-01-1-0380.

present a variety of practical approaches to data hiding, with a focus on scalar quantization based hiding, and show that these schemes are superior to spread spectrum hiding schemes, which simply add a spread version of the hidden data to the host [17]. A scalar quantization based data hiding scheme, together with turbo coding to protect the hidden data, is considered in [18], while a trellis coded vector quantization scheme is considered by Chou et al [19].

Relative to the preceding methods, a key novelty of our approach is that our coding framework permits the use of local criteria to decide where to embed data. The main ingredients of our embedding methodology are as follows.

(a) As is well accepted, data embedding is done in the transform domain, with a set of transform coefficients in the low and mid frequency bands selected as possible candidates for embedding. (These are preserved better under compression attacks than high frequency coefficients)

(b) A novel feature of our method is that, from the candidate set of transform coefficients, the encoder employs local criteria to select which subset of coefficients it will actually embed data in. In example images, the use of local criteria for deciding where to embed is found to be crucial to maintaining image quality under high volume embedding.

(c) For each of the selected coefficients, the data to be embedded indexes the choice of a scalar quantizer for that coefficient. We motivate this by an information-theoretic analysis showing that, for an idealized model [11], scalar quantization based hiding is only about 2 dB away (in terms of resilience to attack) from optimal vector quantization based hiding.

(d) The decoder does not have explicit knowledge of the locations where data is hidden, but employs the same criteria as the encoder to guess these locations. The distortion due to attacks may now lead to insertion errors (the decoder guessing that a coefficient has embedded data, when it actually does not) and deletion errors (the decoder guessing that a coefficient does not have embedded data, when it actually does). In principle, this can lead to desynchronization of the encoder and decoder.

(e) An elegant solution based on erasures and errors correcting codes is provided to the synchronization problem caused by the use of local criteria. Specifically, we use a code on the hidden data that spans the entire set of candidate embedding coefficients, and that can correct both errors and erasures. The subset of these coefficients in which the encoder does not embed can be treated as *erasures at the encoder.* Insertions now become errors, and deletions become erasures (in addition to the erasures already guessed correctly by the decoder, using the same local criteria as the encoder). While the primary purpose of the code is to solve the synchronization problem, it also provides robustness to errors due to attacks.

Two methods for applying local criteria are considered. The first is the block-level Entropy Thresholding (ET) method, which decides whether or not to embed data in each block (typically 8×8) of transform coefficients, depending on the entropy, or energy, within that block. The second is the Selectively Embedding in Coefficients (SEC) method, which decides whether or not to embed data based on the magnitude of the coefficient. Reed-Solomon (RS) codes [20] are a natural choice for the block-based ET scheme, while a "turbo-like" Repeat Accumulate (RA) code [21] is employed for the SEC scheme. We are able to hide high volumes of data under both JPEG and AWGN attacks. Moreover, the hidden data also survives wavelet compression, image resizing and image tampering attacks.

The use of perceptual models and image-adaptation is not new in the watermarking literature. Many of the techniques proposed in the literature are based on a strategy commonly known as *perceptual shaping* (see, for example, [3], [22], and Chapter 7 in [23]). Mostly used in conjunction with spread-spectrum watermarking, perceptual shaping refers to the idea of adjusting the strength of the watermark based on the perceptual sensitivity of a region in the image. All these methods use some model that assigns weights to various regions of the image. This weight determines the strength of the watermark that is added to that part of the image. However, by reducing the strength of the hidden data in the perceptually sensitive area, the robustness of this data against attacks is compromised. It should be noted that the hiding techniques presented in this paper are significantly different from the aforementioned methods. Our approach is based on the idea of not "disturbing" the sensitive coefficients, so as to achieve good image quality without compromising robustness. The number of bits hidden is determined dynamically by the scheme based on the host image content.

We have recently become aware of independent work by Wu and Lui [24], who also propose the concept of *uneven* embedding, where certain transform coefficients are not used for embedding based on a perceptual criteria. Their method, however, requires side information about the hiding locations to be sent to the decoder, which reduces the size of the payload. In contrast, our coding framework obviates the need for sending synchronization data explicitly, while providing great flexibility in terms of the use of application-specific local adaptation criteria (e.g., not hiding data in a sensitive portion of a medical image). In addition, it provides robustness against a variety of attacks such as tampering and resizing.

Note that, while the proposed coding schemes solve the specific insertion-deletion problem that arises in this setting, they do not apply to the more general insertion-deletion channel considered in [25], where the length of the overall symbol sequence can vary. In our situation, the set of candidate coefficients for embedding is the same, and is known to both encoder and decoder: the uncertainty only lies in which of these candidates were actually used for embedding.

Apart from the use of the local criteria and the coding framework, the information-theoretic analysis of scalar quantization based hiding for the idealized model in the paper by Costa [11] is also new. A similar result has been derived in independent work by Eggers et al [26]. In order to compare the theoretical capacity with practically achievable rates, we have also implemented a hiding scheme specifically optimized for AWGN attacks, which gets to within 2 dB of the scalar hiding capacity.

The rest of the paper is organized as follows. In section II, we find the mutual information for the scalar quantization based hiding methods and also derive a decision statistic to be passed to the decoder. In Section III, we introduce our image-adaptive hiding schemes. The coding framework to counter insertions/deletions and errors is described in Section IV followed by a discussion on decoding (Section V). A hiding method op-

timized to AWGN attacks is described in Section VI. Results are presented in section VII and discussed in section VIII.

II. Quantization based data hiding

A. Embedding data in choice of quantizer

Data is embedded in the host medium through the choice of scalar quantizer, as in [16]. For example, consider a uniform quantizer of step size Δ, used on the host's coefficients in some transform domain. Let odd reconstruction points represent a hidden data bit '1'. Likewise, even multiples of Δ are used to embed '0'. Thus, depending on the bit value to be embedded, one of two uniform quantizers of step size 2Δ is chosen. Moreover, the quantizers can be pseudo-randomly dithered, where the chosen quantizers are shifted by a pseudo-random sequence available only to encoder and decoder. As such, the embedding scheme is not readily decipherable to a third party observer, without explicit knowledge of the dither sequence.

Hard decision decoding in this context is performed by quantizing the received coefficient to the nearest reconstruction point of all quantizers. An even reconstruction point indicates that a '0' has been hidden. Likewise, if a reconstruction point lies on an odd quantizer, a '1' has been hidden. However, if more information regarding the statistics of the attack is available, soft decisions can be used to further improve performance. In Section II-B, we compute the capacity of scalar quantization based hiding for the specific case of AWGN attacks. Implicit in our formulation is the use of soft decisions that account for both the quantization noise and the AWGN.

B. Capacity of scalar quantization based data hiding

We now show that our scalar quantization based hiding incurs roughly only a 2 dB penalty for the worst-case AWGN attack. Letting D_1 and D_2 denote the mean squared embedding induced distortion and mean squared attack distortion, the hiding capacity with AWGN attack is given by $C_v = \frac{1}{2}\log(1+\frac{D_1}{D_2})$, in the small D_1, D_2 regime that typical data hiding systems operate [11], [13]. We compare this "vector capacity" (termed thus because the optimal strategy involves vector quantization of the host) to the mutual information of a scalar quantizer embedding scheme with soft decision decoding.

Consider a data hiding system where the information symbol to be embedded is taken from an alphabet $\mathcal{X}$. The host's original uniform quantizer is divided into M uniform sub-quantizers (each with quantization interval $M\Delta$), where $M = |\mathcal{X}|$, a power of two. Thus, $\log_2 M$ bits are hidden per host symbol.

We consider the distortion-compensated quantization embedding scheme of [16] with soft decision decoding. Here, the uniform quantizer is scaled by $\alpha \in (0,1]$, increasing the distance between adjacent quantizers to Δ/α. As such, the embedding robustness is increased by a factor $1/\alpha^2$ (in the squared minimum distance sense), and embedding induced distortion is increased by the same factor. Encoding the information symbol as a linear combination of the host symbol and its quantized value, as in the following, compensates for the additional distortion. Denoting the host coefficient by C, and the hidden message symbol by X, the symbol transmitted by hider is given by

$$Q_X(C) = \alpha q_X(C) + (1-\alpha)C \tag{1}$$

where $q_x(\cdot)$ the scaled uniform quantizer used to embed the information symbol x (with quantization interval $M\Delta/\alpha$). Under an AWGN attack, the received symbol is

$$\begin{aligned} Y &= Q_X(C) + W \\ &= \alpha q_X(C) + (1-\alpha)C + W \\ &= q_X(C) + (1-\alpha)(C - q_X(C)) + W \end{aligned}$$

where W is AWGN with mean zero and variance D_2.

The parameter α achieves a tradeoff between uniform quantization noise and AWGN. The optimal value for α for maximizing the signal-to-noise ratio (SNR) at the decoder, which we have found numerically also to maximize the mutual information $I(X;Y)$, is [16]

$$\alpha_{opt} = \frac{D_1}{D_1 + D_2} \tag{2}$$

The probability density function of the combined additive interferers, $N = (1-\alpha)Z + W$, where $Z \equiv C - q_X(C)$ is the uniform quantization noise, is given by convolving the uniform and Gaussian densities:

$$f_N(x) = \frac{\alpha(2\pi D_2)^{-\frac{1}{2}}}{(1-\alpha)M\Delta} \int_{-\frac{(1-\alpha)M\Delta}{2\alpha}}^{\frac{(1-\alpha)M\Delta}{2\alpha}} \exp(-\frac{(x-\tau)^2}{2D_2})\mathrm{d}\tau \tag{3}$$

We compute the mutual information $I(X;Y) = H(X) - H(X|Y)$ for X uniform over its M-ary alphabet as an estimate of the capacity with scalar quantization based embedding. Thus, $H(X) = \log_2 M$. To find, $H(X|Y)$, we now compute $p_{X|Y}$, the conditional probability mass function of X given Y, and f_Y, the probability density function of Y.

Consider the quantization interval in which the received symbol Y appears, and define its midpoint as the origin. Letting y denote the abscissa, the nearest quantizers appear at $y = \pm\frac{\Delta}{2\alpha}$. Conditioned on the input $X = x$ and host coefficient $C = c$, the distribution of Y is given by $f_{Y|X,C}(y|x,c) = f_N(y - m_x\frac{\Delta}{2\alpha} - k_c\frac{M\Delta}{\alpha})$, with f_N as in (3). Here, $m_x \in \mathcal{M} = \{\pm 1, \pm 3, ..., \pm 2M-1\}$ is uniquely determined by the information symbol x, $k_c \in \mathbb{Z}$ by the host coefficient c, and the hidden quantized host coefficient $q_x(c)$ by the pair (m_x, k_c). Thus we have

$$\begin{aligned} f_{Y|X}(y|x) &= \int_{\mathcal{C}} f_{Y|X,C}(y|x,c) f_C(c)\mathrm{d}c \\ &\propto \sum_{k\in\mathbb{Z}} f_N(y - m_x\frac{\Delta}{2\alpha} - k\frac{M\Delta}{\alpha}) \end{aligned} \tag{4}$$

$$\begin{aligned} f_Y(y) &= \sum_{x\in\mathcal{X}} f_{Y|X}(y|x) p_X(x) \\ &\propto \sum_{m\in\mathcal{M}}\sum_{k\in\mathbb{Z}} f_N(y - m\frac{\Delta}{2\alpha} - k\frac{M\Delta}{\alpha}) \end{aligned} \tag{5}$$

where we have assumed that the host C and message X are statistically independent, and that the host's density f_C is roughly constant on an interval around Y, an assumption that is reasonable in the low distortion regime, where the quantization interval is small with respect to variations in the host's density. This

implies that the density of Y is $\frac{\Delta}{\alpha}$–periodic, so that it suffices to restrict attention to the interval $[-\frac{\Delta}{2\alpha}, \frac{\Delta}{2\alpha}]$, with f_Y normalized accordingly. Applying Bayes' rule, the distribution of X given Y is

$$p_{X|Y}(x|y) = \frac{f_{Y|X}(y|x)p_X(x)}{f_Y(y)} \tag{6}$$

so that we can now compute

$$H(X|Y) = \int_{\mathcal{Y}} \sum_{x\in\mathcal{X}} p_{X|Y}(x|y) \log p_{X|Y}(x|y) f_Y(y) dy$$

and hence $I(X;Y)$.

Due to the exponential decay of the Gaussian density, the summation in (4) is well approximated with only the $k = 0$ term, i.e. the nearest quantization point to y corresponding to x being transmitted. Figure 1 plots the mutual information obtained with 2, 4 and 8-ary signaling, as well as the vector capacity. We observe roughly a 2 dB loss due to the suboptimal scalar quantization encoding strategy.

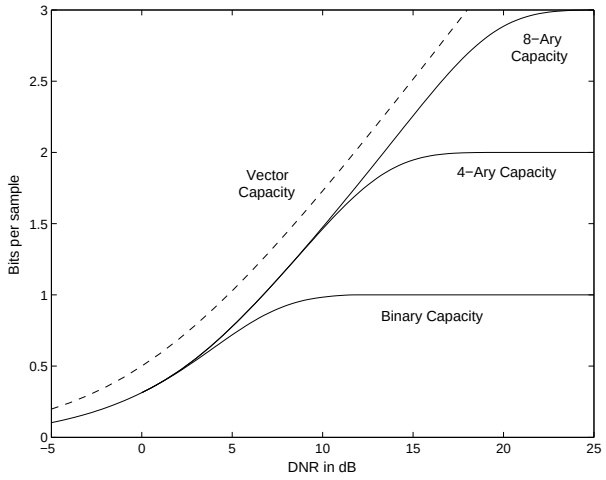


Fig. 1. Gap between scalar and vector quantizer data hiding systems.

C. Soft decision statistic for Distortion Compensated hiding

We conclude our analysis by noting that the soft decision statistic, used by an iterative decoder, is the log likelihood ratio (LLR), given in the following for the case of binary signaling.

$$\Lambda(y) = \log \frac{p_{X|Y}(0|y)}{p_{X|Y}(1|y)} = \log \frac{f_{Y|X}(y|0)}{f_{Y|X}(y|1)} \tag{7}$$

When $\alpha = 1$ and (4) is approximated with $k = 0$ term, the LLR reduces to

$$\Lambda(y) = \log \frac{f_W(y - \frac{\Delta}{2})}{f_W(y + \frac{\Delta}{2})} = \frac{y\Delta}{D_2} \tag{8}$$

We now compute log likelihood ratio (LLR) for any value of $\alpha \in (0, 1]$. We proceed by finding the conditional probability density functions $f_{Y|X}(y|0)$ and $f_{Y|X}(y|1)$, which could be written using (4) as convolution of uniform and Gaussian densities. Again, approximating (4) using the $k = 0$ term, we obtain,

$$f_{Y|X}(y|0) = \frac{\alpha(2\pi D_2)^{-\frac{1}{2}}}{2(1-\alpha)\Delta} \int_{-\frac{(1-\alpha)\Delta}{\alpha}}^{\frac{(1-\alpha)\Delta}{\alpha}} \exp(-\frac{(y-\tau-\frac{\Delta}{2\alpha})^2}{2D_2}) d\tau$$

$$f_{Y|X}(y|1) = \frac{\alpha(2\pi D_2)^{-\frac{1}{2}}}{2(1-\alpha)\Delta} \int_{-\frac{(1-\alpha)\Delta}{\alpha}}^{\frac{(1-\alpha)\Delta}{\alpha}} \exp(-\frac{(y-\tau+\frac{\Delta}{2\alpha})^2}{2D_2}) d\tau$$

The integrals in the above equations can be written as difference of two Q functions, the complimentary cumulative distribution function of a standard Gaussian random variable. We get,

$$f_{Y|X}(y|0) = \frac{\alpha}{2(1-\alpha)} \{Q(\frac{y+\Delta-\frac{3\Delta}{2\alpha}}{\sqrt{D_2}}) - Q(\frac{y-\Delta+\frac{\Delta}{2\alpha}}{\sqrt{D_2}})\}$$

$$f_{Y|X}(y|1) = \frac{\alpha}{2(1-\alpha)} \{Q(\frac{y+\Delta-\frac{\Delta}{2\alpha}}{\sqrt{D_2}}) - Q(\frac{y-\Delta+\frac{3\Delta}{2\alpha}}{\sqrt{D_2}})\}$$

Substituting above equations in LLR expression (7), we get,

$$\Lambda = \log \frac{Q(\frac{y+\Delta-\frac{3\Delta}{2\alpha}}{\sqrt{D_2}}) - Q(\frac{y-\Delta+\frac{\Delta}{2\alpha}}{\sqrt{D_2}})}{Q(\frac{y+\Delta-\frac{\Delta}{2\alpha}}{\sqrt{D_2}}) - Q(\frac{y-\Delta+\frac{3\Delta}{2\alpha}}{\sqrt{D_2}})} \tag{9}$$

Thus we get a relatively simple expression for the soft decision statistic for a general value of $\alpha \in (0, 1]$. The decision-statistic derived here is employed in the iterative decoding of the AWGN optimized hiding (Section V). Note that, while we have used the $k = 0$ term in (4) in deriving these analytical expressions, an arbitrary degree of accuracy can be obtained by considering more terms.

III. Image adaptive data hiding

In order to robustly hide large volumes of data in images without causing significant perceptual degradation, hiding techniques must adapt to local characteristics within an image. Many prior quantization based blind data hiding schemes use global criteria regarding where to hide the data, such as statistical criteria independent of the image (e.g. embedding in low or mid-frequency bands), or criteria matched to a particular image (e.g. embedding in high-variance bands). These are consistent with information theoretic guidelines [15], which call for hiding in "channels" in which the host coefficients have high variance. This approach works when hiding a few bits of data, as in most watermarking applications. However, for large volumes of hidden data, hiding based on such global statistical criteria can lead to significant perceptual degradation. Figure 2 shows 512×512 Harbor image with 16,344 bits hidden using local criteria and with 16,384 bits hidden (a rate of 0.0625 bits/pixel) using statistical criteria (hiding in low frequency band). Both the images were designed to survive JPEG compression at a quality factor of 25. Note that the *statistical criteria* based scheme is one that hides in all the coefficients in a predefined band. In this particular example, a low frequency band comprising of 4 AC

(a) 16,344 bits hidden using local criteria, PSNR = 32.6 dB

(b) 16,384 bits hidden using statistical criteria, PSNR = 31.8 dB

Fig. 2. Local vs Statistical criteria: 512×512 Harbor image with approximately same number of bits hidden using local and statistical criteria. It can be seen that the perceptual quality of the composite image is better in the former.

coefficients was used. It is observed that the perceptual quality as well as the PSNR is better for the image with hidden data using local criteria. Note that though the PSNR is only marginally better (0.8 dB higher), the actual perceptual quality is much better. This illustrates that local criteria must be used for robust and transparent high volume embedding.

Although we do not use specific perceptual models, we refer to our criteria as 'perceptual' because our goal in using local adaptation is to limit perceivable distortion. As evident in the example presented (Figure 2), the employed criterion does succeed in limiting perceptual distortion when hiding a large volume of data. We now describe and extend two image-adaptive hiding techniques, which we had first proposed for uncoded hidden data in [27] and then with a coding framework in [28].

A. Entropy Thresholding scheme

The entropy thresholding (ET) scheme uses the energy (or 2-norm entropy) of an 8×8 block to decide whether to embed in the block or not. Only those blocks whose entropy exceeds a predetermined threshold are used to hide data.

The embedding procedure is outlined as follows. The image is divided into 8×8 non-overlapping blocks, and an 8×8 Discrete Cosine Transform (DCT) of the blocks is taken. Let us denote the intensity values of the 8×8 blocks by a_{ij} and the corresponding DCT coefficients by c_{ij}, where $i, j \in \{0, 1, ..., 7\}$. Thus,

$$\mathbf{c} = \text{DCT}_2(\mathbf{a}) \tag{10}$$

where DCT_2 denotes a 2D DCT.

Next, the energy of the blocks is computed as follows

$$E = \sum_{i,j} \|c_{ij}\|^2, \quad \forall \quad i, j \in \{0, 1, ..., 7\}, (i, j) \neq 0.$$

It should be noted that the DC coefficient is neither used for entropy calculation nor for information embedding. This is because JPEG uses predictive coding for the DC coefficients and hence, any embedding induced distortion would not be limited a single 8×8 block.

The blocks whose energy E is greater than a predefined threshold are selected for information embedding. These blocks are now divided by the JPEG quantization matrix whose entries are computed for a given design quality factor (QF) as per the codec implementation of *independent JPEG group* (IJG) [29]. The design quality factor determines the maximum JPEG compression that the hidden image will survive. Let us denote the quantization matrix entries for a particular quality factor QF as M_{ij}^{QF}, where $i, j \in \{0, 1, ..., 7\}$ and QF $\in \{1, 2, ..., 100\}$, where QF = 100 corresponds to the best quality image. The coefficients c_{ij} used for information embedding are computed as

$$\tilde{c_{ij}} = \frac{c_{ij}}{M_{ij}^{QF}}, \quad \forall \quad i, j \in \{0, 1, ..., 7\}. \tag{11}$$

Next, the coefficients $\tilde{c_{ij}}$ are scanned in zig-zag fashion, as in JPEG, to get one dimensional vector $\tilde{c_k}$ where $0 \leq k \leq 63$. The first n of these coefficients are used for hiding after excluding the DC coefficient ($k = 0$ term). Thus, low frequency coefficients are used for embedding. Bits are hidden using choice of scalar quantizer (Section II). For a binary signature bitstream $\mathbf{b}$, the hidden coefficients $\tilde{d_k}$ are given using the notation in (1) as,

$$\tilde{d_k} = \begin{cases} Q_{b_l}(\tilde{c_k}) & \text{if } 1 \leq k \leq n, \\ \tilde{c_k} & \text{otherwise.} \end{cases} \tag{12}$$

where $b_l \in \{0, 1\}$ is the incoming bit that determines which one of the two quantizers $Q_1(\cdot)$ and $Q_0(\cdot)$ is used.

TABLE I
TYPICAL VALUES OF PARAMETERS USED IN ET SCHEME FOR VARIOUS DESIGN QUALITY FACTORS

Design Quality Factor	Number of coefficients/block	Block Entropy Threshold
75	20	4000
50	14	14000
25	8	25000

The hidden coefficients $\tilde{d_k}$ are reverse scanned to form an 8×8 matrix $\{\tilde{d_{ij}}\}_{i,j=1}^{8}$, and multiplied by the JPEG quantization matrix to obtain $\{d_{ij}\}_{i,j=1}^{8}$. Finally, the inverse DCT of $\{d_{ij}\}_{i,j=1}^{8}$ yields the hidden image intensity values a'_{ij} for that block.

Low frequency coefficients are used to embed in qualifying blocks (i.e., blocks that satisfy the entropy test). Hiding in these coefficients induces minimal distortion due to JPEG's finer quantization in this range. Thus, this scheme employs a statistical criterion by hiding in the frequency subbands of large variance, while satisfying a local perceptual criterion via the block entropy threshold.

In general, compression (quantization of the DCT coefficients) decreases the entropy of the block. Hence, in the uncoded version of the scheme, it is necessary to check that the entropy of each block used to embed information, compressed to the design quality factor, still exceeds the threshold entropy. If a particular block passes the test before hiding but fails the test after the hiding process, we keep it as such, and embed the same data in the next block. However, such a test becomes unnecessary when the ET scheme is used along with a coding framework (Section IV).

The decoder checks the entropy of each 8×8 block to decide whether data has been hidden. Two parameters are shared by the encoder and decoder in this scheme, namely, the block entropy threshold and the set of coefficients used for embedding in a block. As stated, the coefficients are scanned in zig-zag fashion, and only first n are used, excluding the DC coefficient. The parameters values are independent of the host image, and are determined based on the design quality factor used for embedding. Table I shows the values of these parameters used in our experiments.

B. Selectively Embedding in Coefficients scheme

In the Selectively Embedding in Coefficients (SEC) scheme, instead of deciding where to embed at the block level, we do a coefficient-by-coefficient selection, with the goal of embedding in those coefficients that cause minimal perceptual distortion.

Here too, an 8×8 DCT of non-overlapping blocks is taken and the coefficients are divided by the JPEG quantization matrix at design quality factor. Thus, c_{ij} are computed using (10) and then divided by JPEG quantization matrix using (11) to get $\tilde{c_{ij}}$ in the same way as in ET scheme, but the entropy calculation and thresholding steps are skipped. Again, the coefficients are zig-zag scanned (to get $\tilde{c_k}$) and only a predefined low frequency band is considered for hiding (i.e., $1 \leq k \leq n$).

Next, we quantize these coefficient values c_k to nearest integers and take their magnitude to get r_k,

$$r_k = |Q_I(\tilde{c_k})|, \quad 1 \leq k \leq n. \tag{13}$$

We embed in a given coefficient only if r_k exceeds a positive integer threshold t. Embedding is again done using choice of scalar quantizers. We send either $Q_1(\tilde{c_k})$ or $Q_0(\tilde{c_k})$ depending on the incoming bit. Thus $\tilde{d_k}$ can be given as

$$\tilde{d_k} = \begin{cases} Q_{b_l}(\tilde{c_k}) & \text{if } 1 \leq k \leq n, \text{ and } r_k > t, \\ r_k & \text{if } r_k = t, \\ \tilde{c_k} & \text{otherwise.} \end{cases} \tag{14}$$

After reverse scanning, multiplication by JPEG quantization matrix, and inverse DCT, we get the hidden image intensity values a'_{ij} for that block.

A check is required in the scheme when the magnitude of the coefficient lies between t and $t+1$. If the quantized value $Q_{b_l}(\tilde{c_k})$ equals t in (14), then the decoder cannot tell whether this coefficient was not chosen for hiding because of the threshold criteria, or whether b_l was hidden in this coefficient. In coded version of the scheme, this is regarded as an erasure and decoding is performed accordingly. In the uncoded version of the scheme, the same bit b_l is embedded in the next coefficient eligible for embedding. This is done in order to maintain synchronization between encoder and decoder. Note that the decoder simply disregards all coefficients that quantize to a value with magnitude $\leq t$. This check also makes sure that there are no insertions or deletions for JPEG attacks with smaller quantization intervals (higher QFs).

The simplest SEC scheme is the zero-threshold SEC scheme ($t = 0$), where the coefficients that are not quantized to zero are used to embed information. High embedding rates are achieved using this zero-threshold SEC scheme with very low perceptual degradation, which resembles that due to JPEG compression. To understand this intuitively, it should be noted that there are many image coefficients that are very close to zero once divided by the JPEG quantization matrix, and would be quantized to zero upon JPEG compression. Embedding '1' in such coefficients introduces a large amount of distortion relative to the original coefficient size, a factor that seems to be perceptually important. This is avoided by choosing not to use zeros for embedding.

As the threshold increases, fewer coefficients qualify for embedding, and hence less data can be hidden, which provides a tradeoff between hiding rate and perceptual quality. For thresholds $t \geq 2$, it becomes difficult for a human observer to distinguish between the original and composite image, while embedding reliably at fairly high rates. For example, in 512×512 Peppers image, and threshold $t = 2$, one can hide about 2800 bits such that it survives 0.4 bpp JPEG compression (QF=25) and still the composite image is almost indistinguishable from the original one.

In the SEC scheme, we have more control on *where to hide data* compared to the ET scheme, hence it achieves better performance in terms of smaller perceptual degradation for a given amount of data. Another key advantage of the scheme is that it automatically determines the right amount of data to be hidden in an image based on its characteristics.

IV. Coding for insertions and deletions

In the previous section, we noted that use of image-adaptive criteria is necessary when hiding large volumes of data into images. A threshold is used to determine whether to embed in a block (ET scheme) or in a coefficient (SEC scheme). More advanced image-adaptive schemes would exploit the human visual system (HVS) models to determine where to embed information. Distortion due to attack may cause an insertion (decoder guessing that there is hidden data where there is no data) or a deletion (decoder guessing that there is no data where there was data hidden). Such insertions and deletions can potentially cause catastrophic loss of synchronization between encoder and decoder.

In the ET scheme, insertions and deletions are observed when the attack quality factor is mismatched with the design quality factor for JPEG attack. However, for the SEC scheme, there are no insertions or deletions for most of the images for JPEG attacks with quantization interval smaller than or equal to the design interval. This is because no hidden coefficient with magnitude $\leq t$ can be ambiguously decoded to $t+1$ due to JPEG quantization with an interval smaller than the design one. Both the ET and SEC schemes have insertions/deletions under other attacks.

A. Coding Framework

The bit stream to be hidden is coded, using a low rate code, assuming that all host coefficients that meet the global criteria will actually be employed for hiding. A code symbol is erased at the encoder if the local perceptual criterion for the block or coefficient is not met. Since we code over entire space of coefficients that lie in a designated low-frequency band, long codewords can be constructed to achieve very good correction ability. A maximum distance separable (MDS) code, such as Reed Solomon (RS) code, does not incur any penalty for erasures at the encoder. Turbo-like codes, which operate very close to capacity, incur only a minor overhead due to erasures at the encoder. It should be noted that a deletion, which causes an erasure, is about half as costly as an insertion, which causes an error. Hence, it is desirable that the data-hiding scheme be adjusted in such a manner that there are very few (or no) insertions.

Thus, using a good erasures and errors correcting code, one can deal with insertions/deletions without a significant decline in original embedding rate. Reed Solomon codes [20] have been used for ET scheme and Repeat Accumulate codes [21] have been used for the SEC scheme as described in following sections.

B. Reed-Solomon (RS) coding for ET scheme

Reed Solomon codes [20] are MDS codes, such that any k coordinates of an (n,k) RS code can be used to recover the k message symbols, so that the code can correct (n-k) erasures, or half as many errors. The block length n of a Reed-Solomon code must be smaller than the symbol alphabet. More generally, an RS code can correct a pattern of e erasures and r errors as long as $e+2r \leq n-k$, which means that errors are twice as costly as erasures. RS codes use large nonbinary alphabets whose size is a power of 2, so that each symbol can be interpreted as a block of bits. This is well-matched to the block-based ET scheme, where an entire block gets inserted or deleted. Interleaving of the code symbols is required to deal with block erasures at the encoder, which tend to occur in bursts. For example, if an entire codeword were placed in a smooth area of the image, all or most of the symbols would be erased, and it would be impossible to decode this particular codeword at the receiver. The objective of the interleaving is to spread the erasures at the encoder as evenly as possible across codewords, so as to ensure that at least k out of n symbols are received at the decoder with high probability for each codeword. In particular, codewords are arranged in an image in such a way that at least certain code symbols of the codeword are in the center of the image, where the image is most likely to have details.

Let us consider an example of hiding in a 512×512 image. The image is partitioned into 4096 non-overlapping 8×8 blocks. A (128,32) RS code (i.e., rate 1/4) with symbols of size 7 bits is used. 14 coefficients are used per block. Thus there are 2 code symbols per block, and a total of 64 codewords spanning the whole image. The encoder scans the blocks one at a time, evaluates the entropy in the block, and embeds the two code symbols corresponding to the block if it passes the entropy threshold test. Otherwise, the code symbols are erased at the encoder. The rate achieved is computed as follows,

$$\begin{aligned} \text{Rate} &= 64\ \frac{\text{codewords}}{\text{image}} \times 32\ \frac{\text{symbols}}{\text{codewords}} \times 7\ \frac{\text{bits}}{\text{symbol}} \\ &= 14,336\ \text{ bits/image} \\ &= 0.0547\ \text{ bits/pixel (bpp)} \end{aligned}$$

Reed-Solomon codes are not well matched to AWGN channels (where they might more typically serve as an outer code for cleaning up after an inner code matched to the channel), but are ideal for the purpose of illustrating how to deal with the erasures caused by application of local criteria at the encoder and decoder. We now turn to the SEC scheme, where we consider powerful binary codes that are well-matched to AWGN attacks, as well as close to optimal for dealing with erasures.

C. Repeat-accumulate (RA) coding for SEC scheme

Any turbo-like code that operates close to Shannon limit for the erasures channel, while possessing a reasonable error-correcting capability, could be used with the SEC scheme. We used RA codes [21] in our experiments because of their simplicity and near-capacity performance for erasure channels [30]. A rate 1/q RA encoder involves q-fold repetition, pseudorandom interleaving and accumulation of the resultant bit-stream. Decoding is performed iteratively using the sum-product algorithm [31].

The set of candidate coefficients, which governs the length of the RA code, lies within a designated low frequency band. Let us consider an example wherein we want to hide in a 512×512 Lena image. Here, 14 coefficients per block are used (note that this parameter is independent of the host image), giving us a total maximum codeword length of 14×4096 = 57,344 for a 512×512 image. It is observed that about 11,000 coefficients satisfy the zero-threshold test for the Lena image. We choose

a hiding rate of 1/7, which gives us a payload of 8192 bits. This input bitstream is coded using rate 1/7 RA code to form a codeword which is 57,344 bits long. This codeword is now hidden using the local criteria such that if a coefficient does not pass the threshold test, the corresponding code symbol is erased (i.e. not hidden).

V. Decoding

Hard decision decoding is used for JPEG attacks for both the ET and the SEC schemes. For the case of the RA coded SEC scheme under AWGN attack, soft decision or probablistic decoding is employed. It is well known [32] that a soft decisions decoder, leveraging knowledge of attack statistics, outperforms the hard decisions decoder. Hard decision decoding is employed for all other attacks in this paper because a detailed statistical model for these attacks is not available.

A. Hard decision decoding for JPEG attacks

The decoder estimates the location of the embedded data, and uses hard decisions on the embedded bits in these locations. The bits in the remaining locations (out of the set of candidate frequencies) are set to erasures. Since the embedding procedure of both the ET and the SEC scheme is tuned to JPEG, the decoding of embedded data is perfect for all the attacks lesser than or equal to the design quality factor (QF). The coding framework imparts robustness against insertions/deletions as well as occasional errors.

B. Soft decision decoding for AWGN attacks

Soft decision decoding can be employed for RA coded SEC scheme under AWGN attack. The decoder uses the coefficient threshold to determine whether data has been hidden or not. If the coefficient exceeds the coefficient threshold, decoder passes a soft decision statistic computed using (7) to the RA decoder. Otherwise an erasure (LLR, $\Lambda = 0$) is passed. The RA decoder uses the sum-product algorithm [31] to iteratively decode the bits. We now illustrate how the coding framework employed for correcting insertions and deletions, can deal with image tampering.

C. Image Tampering

The coding framework provides flexibility to the encoder in choosing the hiding locations. The code symbols that do not pass the hiding threshold test are *erased at the encoder*. The hiding rate is chosen such that it can deal with insertions/deletions as well as errors due to attacks so that the hidden data is decoded perfectly. This coding framework can also deal with *image tampering* wherein a part of image is replaced by some other image data. Such a tampering can be local or global. In order to survive tampering, the code rate used is further lowered so that we can deal with the errors caused due to the replacement of the image data. Note that code rate is a design parameter shared by encoder and decoder, and hence if tampering attack is anticipated, then a low enough code rate should be chosen beforehand.

Once the hidden bitstream is decoded, localization of the tampered area can be done easily. The decoded bitstream is encoded using the same RA code parameters, so that the originally hidden RA coded stream is reconstructed. Next, the locations in the host image where errors occurred can be found by comparison. If the host image has undergone tampering, then most of the errors would be concentrated at the locations where the tampering was done. Such an ability to robustly decode the bitstream and then localize the tampered area can be useful in medical or forensic applications to detect whether a malicious attacker has tampered with the "evidence".

VI. Hiding optimized for AWGN attacks

In this section we present a scalar quantization based hiding strategy that is specifically tuned to AWGN attacks. The goal is to compare the achievable rates with the scalar capacity bound derived in Section II-B and the vector capacity ([11],[15]). Note that the image adaptive hiding schemes considered so far are not optimized to AWGN attacks. They use a local criteria, so that some of the coding effort is 'used-up' in dealing with insertions and deletions. Also, the DCT coefficients are divided by JPEG quantization matrix, which does not provide equal robustness to all of them against AWGN attacks. In the following we describe the embedding system, which uses scalar quantization based distortion compensated hiding, RA codes, and soft decision decoding using the statistic derived in Section II-C.

As in the theoretical formulations, the problem is to hide in a host in such a way that the data hider induces a mean squared error of at most D_1, while the attacker is allowed a maximum mean squared error of D_2. In order to compare with the information theoretic limits (see, for example, Costa [11] and Moulin and O'Sullivan [13]), we assume that both the encoder and the decoder know the D_1 and D_2 values. We employ the distortion compensated hiding scheme (Section II-B), which has been shown in [16] to achieve capacity for some specific cases. Here, the uniform quantizer is scaled by $1/\alpha$, where $\alpha \in (0, 1]$, and the information symbol is encoded as a linear combination of the host symbol and its quantized value as in (1). Local criteria are not used, and the quantizer step size is kept same for all DCT coefficients (as opposed to using the JPEG quantization matrix). $\alpha \in (0, 1]$ is computed using (2) and is known to both encoder and decoder. RA codes are used to code the input bitstream to generate a huge codeword. This codeword is embedded bit-by-bit in all the coefficients within a designated band using distortion compensation. At the receiver, the soft decisions are computed using (9) and passed to the RA decoder which uses the sum-product algorithm [31] to iteratively decode the bits.

We use this hiding strategy to illustrate that using relatively simple RA codes with distortion compensated hiding, we can reach about 2 dB close to the scalar capacity (Section VII). However, it should be noted that this scheme is not likely to survive other attacks, and cannot be applied practically unless the attack is known to be AWGN.

VII. Results

We now show that using the proposed image-adaptive hiding methods, one can hide a large volume of data with minimal per-

ceptual degradation. We use peak signal-to-noise ratio (PSNR) as an objective metric to quantify the quality of the hidden image. PSNR is defined as,

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)$$

where MSE stands for average mean squared error between the original and the given image. Table II shows the number of bits hidden and the corresponding observed PSNR for various images with data hidden using uncoded zero-threshold SEC scheme. Data is hidden in raw (uncompressed) images, and robustness of these images is characterized by the design QF, which determines the maximum level of JPEG compression the images can survive. It is observed that the PSNR of the hidden image is significantly higher than that of the corresponding JPEG compressed image at the same design QF. Note that, the PSNR is measured with respect to the original uncompressed image in both the cases. For example, the PSNR of JPEG compressed Baboon image at QF = 25 is 25.89 dB, while a much higher PSNR of 32.27 dB is observed for the same image with 25,331 bits hidden at a design QF of 25. Similar behavior has been observed for all the test images. The hidden image quality can be further improved by using higher threshold SEC scheme, which provides us with a trade-off between the image quality and the volume of embedding at a given robustness (determined by design QF). Table III shows the performance of the higher threshold SEC scheme for various images at a design QF of 25. In almost all these cases, it is impossible for a human observer to tell the hidden image apart from the original one.

We now present the performance of our schemes under various attack scenarios. Coding is used in all the attack scenarios (except JPEG compression where uncoded transmission is good enough for error free recovery), so that all the hidden bits can be decoded in spite of the errors due to attack. Note that the 'number of bits' reported in the following sections are actually the 'number of *information* bits' (i.e., the number of bits hidden before coding). Results for both RS-ET and RA-SEC systems have been provided for JPEG and AWGN attacks. For all other attacks, only the RA-SEC system is used. We discuss in Section VIII why RA-SEC system is preferred.

TABLE II

ZERO-THRESHOLD SEC SCHEME: PSNR AND NUMBER OF BITS HIDDEN FOR VARIOUS 512×512 IMAGES AT DIFFERENT DESIGN QUALITY FACTORS. THE NUMBER OF BITS HIDDEN ARE REPORTED FOR UNCODED HIDING.

Image	QF=25		QF=50		QF=75	
	# bits	PSNR (dB)	# bits	PSNR (dB)	# bits	PSNR (dB)
Lena	11,044	34.58	18,786	38.07	31,306	39.90
Peppers	10,447	35.89	18,972	38.03	32,567	39.63
Baboon	25,331	32.27	44,142	34.50	66,911	36.05
Bridge	24,633	32.34	42,615	34.64	63,955	36.32
Couple	15,545	34.05	27,823	36.25	44,227	38.03
Boat	15,234	34.21	26,518	36.47	41,826	38.33

A. JPEG attacks

Since the embedding procedure of both ET and SEC schemes is tuned to JPEG, the decoding of embedded data is perfect for all the attacks lesser than or equal to the design quality factor (QF). Table IV shows the number of bits embedded (with perfect recovery) in uncoded and coded ET and SEC schemes at various design QFs, under JPEG attacks for 512×512 Lena image.

TABLE III

HIGHER-THRESHOLD SEC SCHEME: PSNR AND NUMBER OF BITS HIDDEN FOR VARIOUS 512×512 IMAGES USING DIFFERENT THRESHOLD VALUES AT DESIGN QF=25. USING HIGHER THRESHOLDS PROVIDE VERY GOOD QUALITY HIDDEN IMAGES WITH A LOWER VOLUME EMBEDDING.

Image	Thresold = 1		Thresold = 2		Thresold = 3	
	# bits	PSNR (dB)	# bits	PSNR (dB)	# bits	PSNR (dB)
Lena	4,913	41.43	2,595	44.58	1,820	46.60
Peppers	5,063	41.12	2,810	44.09	1,976	46.18
Baboon	13,065	35.98	5,763	39.92	3,247	43.27
Bridge	11,403	37.19	5,202	41.03	3,185	43.96
Couple	7,329	39.20	3,751	42.76	2,513	45.18
Boat	6,859	39.39	3,362	42.97	2,264	45.46

TABLE IV

PERFORMANCE OF CODED AND UNCODED ET AND SEC SCHEMES UNDER JPEG ATTACKS AT VARIOUS QUALITY FACTORS

QF	attack compr. (bpp)	ET scheme # of bits		SEC scheme # of bits	
		uncoded	coded	uncoded	coded
25	0.42	6,240	4,608	11,044	7,168
50	0.66	15,652	12,096	18,786	13,824
75	1.04	34,880	30,560	31,306	23,893

B. AWGN attacks

Table V summarizes the results for the ET scheme with RS coding and SEC scheme with RA coding against AWGN attack. The number of bits embedded is listed for the 512×512 Lena image. The 'attack power' reported here is the actual power of the added noise converted to the dB scale (i.e., the ratio of variance of the added noise to that of a Gaussian with unit variance). Although the RS code is not the best choice for AWGN, it is adequate for mild attacks. RA-coded SEC scheme uses soft decision statistic of the AWGN for decoding (as in (8) in Section II-B), and performs better than RS coded ET system at higher attack powers. A worst case attack D_2 is assumed by the decoder to compute the soft-decision statistic, and the hidden image is also attacked at the same D_2. Note that if the actual attack is lesser than D_2, the performance would at least be as good as the one reported here.

C. Wavelet compression attacks

Wavelet compression (JPEG 2000) was used to attack the images with hidden data using SEC scheme with RA coding. Table VI gives the number of bits hidden in 512×512 Lena image under various levels of attack compression. Data was hidden in the image using SEC scheme at design quality factor of 25, and 20 coefficients were used per block, scanned in the zig-zag fashion. The JPEG 2000 compression was done using the Jasper codec [33].

TABLE V

PERFORMANCE OF ET SCHEME WITH RS CODING AND SEC SCHEME WITH RA CODING UNDER AWGN ATTACK. FOR THE ET SCHEME, ONE CODEWORD (8 BITS LONG) IS HIDDEN PER BLOCK. 20 AC COEFFICIENTS CONSTITUTE THE CANDIDATE EMBEDDING BAND FOR THE SEC SCHEME.

Attack power (dB)	ET Scheme		SEC Scheme	
	# of bits	RS code (n,k)	# of bits	RA code (1/q)
10.0	7,040	(256,55)	7,447	1/11
12.5	6,528	(256,51)	6,826	1/12
15.0	3,584	(256,28)	6,301	1/13

TABLE VI

PERFORMANCE OF RA CODED SEC SCHEME FOR 512×512 LENA IMAGE UNDER WAVELET COMPRESSION ATTACK

Attack Compression (bpp)	Hiding Rate # of bits	RA code rate (1/q)
0.800	7,447	1/11
0.530	4,096	1/20
0.400	2,730	1/30

D. Image Tampering

The hiding schemes presented here are resilient to image tampered in various ways. Table VII gives the number of bits hidden in 512×512 Lena image when a part of host image is replaced by other image data. Figure 3(a) shows an example attacked image where 20% of the image is cropped out and new image data is put in that place. The hidden data can be decoded even if the tampering is not localized. Figure 3(b) shows Lena image tampered globally, and still all the 6,301 hidden bits can be recovered successfully. Figure 3 (c) shows the localization results for the tampered image of Figure 3 (b).

E. Image Resizing

Image resizing is a popular attack method wherein the image is shrunk to a smaller size and scaled back to its original size so that there is loss of information in the process without causing significant perceivable distortion. Various interpolation methods can be used to resize and the most popular ones are bilinear, bicubic and nearest neighbor interpolations. Again, the RA coded SEC scheme is used for hiding in 512×512 Lena image at design quality factor of 25 and 20 coefficients are used per block. The hidden image survives large amount of resizing using bicubic interpolation method. Table VIII gives the number of bits hidden against the percentage of resizing done using bicubic interpolation. Less data can be hidden when hidden image is resized using other interpolation techniques. Table IX gives the number of bits hidden against bilinear and nearest neighbor resizing attacks. It should be noted that the perceptual quality of the attacked image is also worse in the latter cases, which forbids the attacker from using a higher percentage of resizing with bilinear or nearest neighbor interpolation.

TABLE VII

PERFORMANCE OF RA CODED SEC SCHEME FOR 512×512 LENA IMAGE UNDER IMAGE TAMPERING. HERE, 27 COEFFICIENTS ARE USED PER BLOCK

Percentage of image tampered	Number of bits	RA code rate (1/q)
10 %	9,216	1/12
20 %	5,820	1/19
30 %	4,608	1/24

TABLE VIII

PERFORMANCE OF RA CODED SEC SCHEME FOR 512×512 LENA IMAGE UNDER IMAGE RESIZING ATTACK USING BICUBIC INTERPOLATION

Percentage Resizing	Hiding Rate # of bits	RA code rate (1/q)
10 %	7,447	1/11
15 %	6,826	1/12
20 %	6,301	1/13

TABLE IX

PERFORMANCE OF RA CODED SEC SCHEME FOR 512×512 LENA IMAGE UNDER IMAGE RESIZING ATTACK USING BILINEAR AND NEAREST NEIGHBOR INTERPOLATION

Percentage Resizing	Nearest neighbor interpolation		Bilinear interpolation	
	Number of bits	RA code (1/q)	Number of bits	RA code (1/q)
2 %	6,301	1/13	2,275	1/36
5 %	4,096	1/20	2,155	1/38
10 %	2,275	1/36	1,241	1/66

F. Image-in-Image hiding

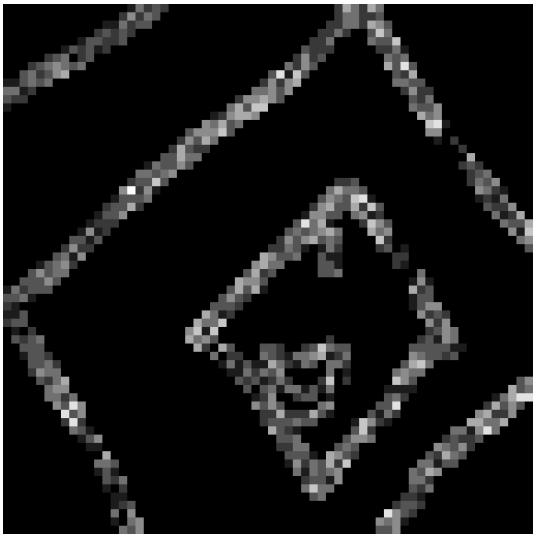
In steganographic applications it is desirable to hide an image called signature image into another image called host or cover image. The hiding techniques developed here allows us to hide large volume of data with perfect recovery and hence can be used to hide large signature images with robustness against JPEG attacks. For example, signature images as large as 256×256 pixels can be hidden in a 512×512 cover image (Figure 4). The uncoded scheme is employed here, because we need robustness only against JPEG compression and higher embedding rate is desirable. First, the maximum number of bits that can be hidden in the host image is determined by going through the image and computing the number of coefficients that satisfy the local criteria at desired design quality factor. Then, the signature image is hidden after being JPEG compressed to a level that its size is smaller than the maximum number of bits that can be hidden.

G. AWGN optimized hiding

For the AWGN optimized hiding scheme discussed in Section VI, we found the minimum distortion to noise ratio (DNR) for which decoding was perfect for a 512×512 image at various RA code rates. Table X compares the DNR observed for simple scalar quantization based hiding (α = 1), and distortion compensated scalar quantization hiding with optimal α ($= \frac{D_1}{D_1+D_2}$)

(a) 20 % of 512×512 Lena image tampered

(b) 512×512 Lena image tampered globally



(c) Localization of tampered area at the decoder for the globally tampered image above

Fig. 3. Global and Localized image tampering and localization of the tampered area

to the theoretical scalar (Section II-B) and vector [15] capacities.

We observe that we are only about 2 dB away from the theoretical scalar capacity using distortion compensated quantization based hiding with RA coding. Most of this gap is probably due to the limits on the performance of the regular RA codes, which exhibit gaps of comparable size (e.g., about 1.5 dB for rate 1/3) from the Shannon limit over the classical AWGN channel as well [21]. An interesting question for future study is whether this gap can be closed further using more powerful codes such as regular and irregular LDPCs [34], [35] and irregular RA codes [30], known to work close to the Shannon limit over the AWGN channel. Another significant observation is that there is a gain of more than 2 dB when distortion compensation scheme is used as compared to the performance without distortion compensation (α = 1).

TABLE X
COMPARISON OF OBSERVED AND THEORETICAL CAPACITIES

RA code rate	Scalar quant. schemes, DNR (dB)		Theoretic Capacity DNR (dB)	
	(α = 1)	(opt. α)	Scalar	Vector
1/3	4.3180	2.1261	0.2500	-2.3107
1/4	3.2790	0.8365	-1.0000	-3.8278

(a) Original 512×512 Harbor image

(b) Composite image

(c) Original 256×256 signature image

(d) Recovered signature image

Fig. 4. Image-in-Image hiding example

VIII. Discussion

The hiding methods presented in this paper are geared towards high volume embedding while preserving the perceptual quality and achieve robustness against JPEG attacks. It should be noted that we use ET scheme with RS coding mainly to explain our ideas of local adaptation and coding framework, while in most practical scenarios, the RA coded SEC scheme is used. The RA-SEC system provides a better performance in terms of robustness and perceptual quality. This is because the turbo-like RA codes operate operate very close to the capacity, and the SEC scheme provides a better control on 'where to hide data'. Soft decision decoding of the RA codes is performed for AWGN attack, and hard decision decoding is performed otherwise.

While the AWGN attack is not common in the watermarking literature, it has been shown in information-theoretic studies ([14],[15]) to be the worst-case attack in certain idealized game-theoretic settings, where the mean squared distortion due to the attack is constrained. The information-theoretic "goodness" of our schemes is therefore demonstrated by our numerical results that show that, by appropriate use of soft decisions, we do approach the information-theoretic hiding capacity (with scalar quantization) under AWGN attacks. Of course, from a practical point of view, hard decisions must be employed for attacks (such as compression) whose statistics are difficult to quantify. Also, there are many attacks that induce large mean-squared distortion, but little perceptual distortion. Examples include Stirmark random bending [4], rotation, cropping, and print-scan. These geometric attacks tend to de-synchronize the decoder. Modifications to the current hiding framework so that it allows re-synchronization of the decoder for these attacks is an avenue of future work.

It can be seen that the proposed hiding schemes survive wavelet based compression and image resizing attacks. This is because these attacks do not entirely destroy the low frequency DCT coefficients where the majority of bits have been hidden. Note that wavelet-based compression does not change the image mean squared error drastically (as opposed to geometric attacks). Hence, based on the arguments of the previous paragraph, it is not surprising that the hidden bits survive this

attack. The same arguments hold true for the image resizing attack when the original image size is known to the decoder, or if the attacker scales the image back to its original size. In spite of this restriction, the presented results are significant because they indicate that the hidden bits can survive errors caused due to interpolation.

The image-in-image hiding presented here uses the fact that we can send a high volume of data with robustness against JPEG compression using uncoded SEC scheme. The signature image is compressed into a sequence of bits and these bits are hidden into the host (disregarding the actual meaning of the bits). The system is designed for the worst anticipated attack. In practice, the attack level is seldom known apriori, and if the actual attack is less severe than the design attack, we are still struck with the design signature image quality. Ideally, we would like an image-in-image hiding scheme that results in graceful improvement in the image quality with less severe attacks. Such schemes require *joint* source-channel coding, which has been studied for the Gaussian channel (see, for example, [36], [37]). Development of similar techniques for data hiding is an important research area. A first attempt at building such gracefully improving image-in-image hiding system is presented in [38], where a hybrid digital-analog (joint source-channel) coding scheme is proposed. It leverages the current image-adaptive hiding framework for sending digital data and involves transmission of the analog residues using a new method.

REFERENCES

[1] M. D. Swanson, M. Kobayashi, and A. H. Tewfi k, "Multimedia data-embedding and watermarking technologies," *Proceedings of the IEEE*, vol. 86, pp. 1064–1087, 1998.

[2] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding — A survey," *Proceedings of the IEEE, special issue on Identification and Protection of Multimedia Information*, vol. 87, no. 7, pp. 1062–1078, 1999.

[3] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proceedings of the IEEE, special issue on Identification and Protection of Multimedia Information*, vol. 87, no. 7, pp. 1108–1126, 1999.

[4] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Workshop Information Hiding, IH'98, LNCS 1525, Springer-Verlag*, 1998, pp. 219–239.

[5] M. Kutter, "Watermarking resisting to translation, rotation and scaling," in *Proc. of SPIE: Multimedia systems and applications*, Nov. 1998, vol. 3528, pp. 423–431.

[6] C.Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation, scale and translation resilient watermarking for images," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 767–782, May 2001.

[7] J. Fridrich, M. Goljan, and D. Hogea, "Attacking the OutGuess," in *Proceedings of ACM Workshop on Multimedia and Security*, Juan-Pins, France, 2002.

[8] K. Sullivan, O. Dabeer, U. Madhow, B.S. Manjuath, and S. Chandrasekaran, "LLRT based detection of LSB hiding," in *Proc. ICIP*, Barcelona, Spain, Sept 2003.

[9] A. Westfeld, "F5 steganographic algorithm," in *4th International Workshop on Information Hiding*, 2001.

[10] N. Provos, "Defending against statistical steganalysis," in *In 10th USENIX Security Symposium*, Washington DC, USA, 2001.

[11] M. H. M. Costa, "Writing on dirty paper," *IEEE Trans. on Info. Theory*, vol. 29, no. 3, pp. 439–441, May 1983.

[12] S. I. Gel'Fand and M. S. Pinsker, "Coding for channel with random parameters," *Problems of Control and Information Theory*, vol. 9, no. 1, pp. 19–31, Jan. 1979.

[13] P. Moulin and J. A. O'Sullivan, "Information-theoretic analysis of information hiding," *IEEE Trans. on Info. Theory*, vol. 49, no. 3, pp. 563–593, Mar. 2003.

[14] A. S. Cohen and A. Lapidoth, "The Gaussian watermarking game," *IEEE Trans. on Info. Theory*, vol. 48, no. 6, pp. 1639–1667, June 2002.

[15] P. Moulin and M. K. Mihcak, "A framework for evaluating the data-hiding capacity of image sources," *IEEE Trans. on Image Processing*, vol. 11, no. 9, pp. 1029–1042, Sept. 2002.

[16] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. on Info. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[17] I. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.

[18] M. Kesal, M. K. Mihcak, R. Koetter, and P. Moulin, "Iteratively decodable codes for watermarking applications," in *Proc. 2nd Int. Symp. on Turbo Codes and Related Topics*, Sept. 2000.

[19] J. Chou, S. S. Pradhan, and K. Ramchandran, "A robust optimization solution to the data hiding problem using distributed source coding principles," in *Proceedings of Conference on Information Sciences and Systems (CISS)*, Mar. 2000.

[20] S. B. Wicker and V. K. Bhargava, *Reed-Solomon Codes and Their Applications*, IEEE Press, 1994.

[21] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for turbo-like codes," in *36th Allerton Conference on Communications, Control, and Computing*, Sept. 1998, pp. 201–210.

[22] C. I. Podilchuk and W. Zeng, "Image adaptive watermarking using visual models," *IEEE Journal of Selected Areas in Communication*, vol. 16, no. 4, pp. 525–539, 1998.

[23] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*, Morgan Kaufmann, 2001.

[24] M. Wu and B. Liu, "Data hiding in images and video: Part I - fundamental issues and solutions," *IEEE Transactions on Image Processing*, vol. 12, no. 6, pp. 685–695, June 2003.

[25] M. C. Davey and D. J. C. Mackay, "Reliable communication over channels with insertions, deletions, and substitutions," *IEEE Trans. on Info. Theory*, vol. 47, no. 2, pp. 687–698, Feb. 2001.

[26] J. J. Eggers, R. Buml, R. Tzschoppe, and B. Girod, "Scalar costa scheme for information embedding," *IEEE Trans. on Signal Processing*, vol. 51, no. 4, pp. 1003–1019, Apr. 2003.

[27] K. Solanki, N. Jacobsen, S. Chandrasekaran, U. Madhow, and B. S. Manjunath, "High-volume data hiding in images: Introducing perceptual criteria into quantization based embedding," in *Proc. ICASSP*, Orlando, FL, USA, May 2002.

[28] N. Jacobsen, K. Solanki, U. Madhow, B. S. Manjunath, and S. Chandrasekaran, "Image adaptive high volume data hiding based on scalar quantization," in *Proc. IEEE Military Comm. Conf. (MILCOM)*, Anaheim, CA, USA, Oct. 2002.

[29] G. K. Wallace, "The jpeg still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[30] H. Jin, A. Khandekar, and R. J. McEliece, "Irregular repeat-accumulate codes," in *Proceedings 2nd International Symposium on Turbo codes and Related Topics*, Sept. 2000, pp. 1–8.

[31] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Info. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[32] J. G. Proakis, *Digital Communications*, McGraw-Hill, 1995.

[33] M. D. Adams and F. Kossentini, "Jasper: A software-based jpeg-2000 codec implementation,".

[34] R. G. Gallager, "Low density parity check codes," *IRE Transactions on Information Theory*, vol. IT-8, no. 12, pp. 21–28, Jan. 1962.

[35] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 32, no. 18, pp. 1645–1646, August 1996.

[36] B. Chen and G. W. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *IEEE Trans. on Communications*, vol. 46, no. 7, pp. 881–890, July 1998.

[37] M. Skoglund, N. Phamdo, and F. Alajaji, "Design and performace of VQ-based hybrid digital-analog joint source-channel codes," *IEEE Trans. on Info. Theory*, vol. 48, no. 3, pp. 1082 –1102, Mar 2002.

[38] K. Solanki, O. Dabeer, B. S. Manjunath, U. Madhow, and S. Chandrasekaran, "A joint source-channel coding scheme for image-in-image data hiding," in *Proc. ICIP*, Barcelona, Spain, Sept. 2003.