A ROBUST DATA HIDING SCHEME FOR IMAGES USING DFT

Mahalingam Ramkumar, Ali N. Akansu and A Aydın Alatan

Department of Electrical and Computer Engineering
New Jersey Institute of Technology
New Jersey Center for Multimedia Research
University Heights, Newark, NJ 07102

ABSTRACT

We present a data hiding scheme for still images, in which only the magnitude of the DFT coefficients are altered to embed the hidden information bits. We begin by identifying various *components* of a typical data hiding scheme - like the *decomposition* used, *signature design*, *self-noise suppression* and *signaling*. The final choice of the components of the proposed data hiding scheme are well tailored by utilizing theoretical reasonings and experimental observations. Performance results show good robustness of the data hiding scheme for both JPEG and SPIHT image compression.

1. INTRODUCTION

Data hiding in raw multimedia data is a relatively new field which developed mainly out of growing concerns of protection of ownership rights of digital multimedia data. However, multimedia content providers envisage many other emerging applications utilizing data hiding. Some of these are [1] content description, tracking of use for pay-per-use applications, e-commerce, customized multimedia delivery, tamper detection *etc.*. These applications would involve a communication scenario through a *subliminal channel* between *content providers* and *generic multimedia players*.

Data hiding in still images is achieved by *imperceptibly* modifying the original image to add some information-bearing *message signal* **S** (often in data hiding literature, **S** is also referred to as *signature*, especially in the context of watermarking applications). In Figure 1 (a), **I** is the original image. The modified image

$$\hat{\mathbf{I}} = \mathbf{I} + \mathbf{S} \tag{1}$$

should be *visually indistinguishable* from I. The image $\hat{I}$ may be typically subjected to a lossy compression,

$$\tilde{\mathbf{I}} = \mathcal{C}(\hat{\mathbf{I}}), \tag{2}$$

where $\mathcal{C}(.)$ denotes compression / decompression operations. We would like to extract the hidden information signal **S** from $\tilde{\mathbf{I}}$.

In Refs. [2, 3], we modeled data hiding schemes for still images as a communication channel (Figure 1 (b)) with two sources of noise - noise due to the original image (otherwise referred to as image noise or self-noise or host-signal noise) $\boldsymbol{I}$, and noise due to processing (also referred to as channel noise), $\boldsymbol{P}$. Further, we argued how a decomposition of an original image or cover image using some transform (or in other words, decomposing the data hiding channel into multiple channels as shown in Figure 1 (c)) could help in increasing the data hiding capacity. We obtained information theoretic estimates of data hiding in still images for various decompositions, for JPEG and SPIHT compression. We also extended these results to video frames undergoing MPEG-2 compression in Ref. [4].

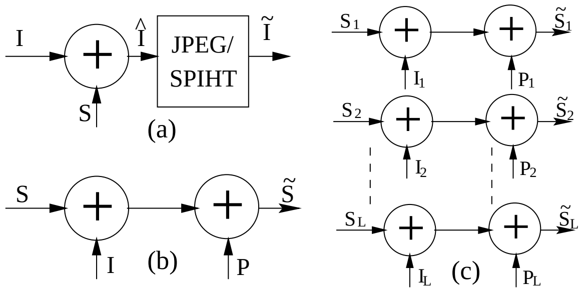

Figure 1: (a) Block diagram of Data Hiding (b) The Data Hiding Channel (c) Decomposition of the Data Hiding Channel

2. PROBLEM STATEMENT

Let **I** be the original image. A *decomposition* $\mathcal{T}$ of the original image results in $\mathbf{c} = \mathcal{T}(\mathbf{I})$, a vector of coefficients of the original image. To embed a sequence of bits **b** in the image, the sequence of bits is *mapped* to a *signature sequence*

$$\mathbf{s} = \mathcal{S}(\mathbf{b}). \tag{3}$$

We then add **s** to **c** to obtain

$$\hat{\mathbf{c}} = \mathbf{c} + \mathbf{s} \quad \hat{\mathbf{I}} = \mathcal{T}^{-1}(\hat{\mathbf{c}}), \tag{4}$$

where $\hat{\mathbf{I}}$ is the image with the signature (or hidden bits). The image $\hat{\mathbf{I}}$ is then transmitted through a channel (containing the compressor/ decompressor $\mathcal{C}$), the output of which is $\tilde{\mathbf{I}}$. The corresponding coefficients $\tilde{\mathbf{c}} = \mathcal{T}(\tilde{\mathbf{I}})$ are extracted, and then the hidden bits are extracted as

$$\tilde{\mathbf{b}} = \mathcal{S}^{-1}(\tilde{\mathbf{c}}) = \mathcal{S}^{-1}(\mathbf{c} + \mathbf{s} + \mathbf{n}), \tag{5}$$

where **n** is noise introduced in the channel (by the lossy compressor / decompressor). Note that for detection of the signature **s**, the coefficients of the original (cover) image **c** is noise.

From the above model of the data hiding scheme, there are two obvious issues to be addressed before an efficient overall scheme can be developed:

1. Choice of the decomposition $\mathcal{T}$. This issue is addressed in Section III.
2. Choice of the signaling scheme $\mathcal{S}$. This issue is addressed in Section V.

But there are also *other issues* which are not immediately obvious from the above model. The first among them is the *choice* *of the signature sequence.* To appreciate that, one should realize that the channel (the compressor / decompressor) is designed for efficient transmission of the *host-signal* - and *not* for the transmission of hidden message signal. If the channel (compression scheme) were *ideal*, it would *completely eliminate* the signature component of the composite signal $\hat{\mathbf{I}}$. An ideal compressor would suppress any component of $\hat{\mathbf{I}}$ that is not needed for effective perception of the image $\mathbf{I}$. As the signature component $\mathbf{s}$ does nothing to *enhance* the visual quality of $\mathbf{I}$, it will be completely eliminated by the compressor. Thus no data hiding would be possible through an ideal channel (compressor). However, practical compressors are not ideal. So an efficient data hiding scheme should utilize the "holes" in practical compression schemes. The issue of *choice* of signature sequences might imply some *constraints* in the choice of the signature sequence so that the hidden signal $\mathbf{s}$ can *sneak through* the channel. It should be noted here that placing constraints on the degrees of freedom of the signature will reduce the *entropy* of the signature, and therefore the *capacity* of the data hiding channel.

The second issue that is not immediately obvious, is *self-noise suppression.* In Eq (5), $\mathbf{c}$ is noise for the purpose of detection of the signature. If the original image is available at the receiver, then it is obvious that this component of the noise can be *completely* eliminated. In Section IV, we shall show that a significant portion of the self noise can be eliminated *even if the original image is not available at the receiver.*

3. THE DECOMPOSITION $\mathcal{T}$

The *purpose of a decomposition* is primarily to distribute the image and processing noise in the data hiding channel in an *intelligent* way. Typically any Fourier based transform would decompose the image into frequency bands. Low frequency bands have very high self noise and low processing noise. In other words, coefficients of low frequency bands have high variances and are relatively unaffected by processing. High frequency bands on the other hand, have high processing noise, but low self noise. It was observed in [2, 3, 4] that the choice of the the decomposition should depend on the nature of the application - especially the robustness requirements;

- For low robustness (*e.g.* robustness to high quality compression) high energy compaction or GTC (Transform Coding Gain) [5], transforms (transforms suitable for compression applications - like DCT, Wavelets etc.) are more suitable.
- For increased robustness, (*e.g.* robustness to poorer quality compression) lower GTC transforms like Hadamard or Hartley transforms are more suitable.
- An exception to above rules is the *magnitude DFT decomposition* (using only the DFT magnitude coefficients and ignoring the phase), which yields superior performance for both low and high robustness requirements.

Obviously, the choice of the decomposition would also depend on the *type of processing* the image is most likely to undergo. *Matching* the decomposition with the compression transform may result in a better performance of the data hiding scheme. For example a DCT based data hiding scheme designed intelligently (say, by taking JPEG quantization tables into account) can be very robust to JPEG compression. But the same scheme may not be robust to SPIHT compression [6]. With a widely accepted wavelet based compression scheme for the imminent JPEG 2000 standard, it may be crucial to design data hiding schemes that are robust to both DCT and Wavelet based compression schemes. In addition, it would be very difficult to design wavelet / DCT based schemes *intelligently* for video compression.

For the magnitude DFT decomposition, we obtain block based DFT of the original image. For example, 2-D DFT of 8×8 blocks of the image, results in 64 DFT coefficients. Out of the 64 coefficients, 4 are real, and 60 complex. Only 30 of the 60 complex coefficients however, have unique magnitudes. Therefore, a total of 34 coefficients per block have unique magnitudes. For adding a signature, we modify only the DFT magnitudes - the phase is untouched. As no signature information is buried in the DFT phase, the phase information can be completely ignored when the signature is detected. The superiority of the magnitude DFT decomposition can be attributed to the following reasons:

- It is well known that the DFT magnitude coefficients convey very little information about any typical image - most of the information is contained in the *phase.* Very little information content of the DFT magnitude coefficients translates to very little *entropy* which in turn translates to very little *self-noise* of the magnitude DFT coefficients. Thus ignoring the phase of the DFT coefficients can result in *substantial reduction of the self-noise.*
- As the magnitude DFT coefficients have very little importance for the purpose of perception of an image, one would expect good compressors to preserve the DFT phase well and give little importance to preserving DFT magnitudes. However, our simulations show that all major compression schemes (JPEG, SPIHT and MPEG), surprisingly, *preserve the DFT magnitude coefficients* as well as they preserve the DFT phase (this 'hole' in compression schemes probably arises out of the fact that most compression schemes tend to maximize the PSNR). So using the magnitude DFT decomposition is a way of exploiting a hole in practical compression schemes.

However, to take advantage of the *reduced entropy* of the magnitude DFT coefficients, some *additional processing* is required after obtaining the magnitude DFT coefficients. We begin by removing the correlation between the coefficients. To remove the correlation between the 34 coefficients of each block, we obtain the Karhunen-Loeve Transform (KLT) of the 34 magnitude DFT coefficients. Let $\mathbf{x}$ represent the vector of 34 magnitude DFT coefficients of a 8×8 image block. The 34×34 KLT matrix $\mathbf{T}$ is obtained as follows. Let $\mathbf{y} = \mathbf{T}\mathbf{x}$. Complete decorrelation of the coefficients of $\mathbf{y}$ implies that $\mathbf{R}_y = E[\mathbf{y}\mathbf{y}^{\mathrm{T}}]$ is a diagonal matrix;

$$\mathbf{R}_y = E[\mathbf{y}\mathbf{y}^{\mathrm{T}}] = \mathbf{T}E[\mathbf{x}\mathbf{x}^{\mathrm{T}}]\mathbf{T}^{\mathrm{T}} = \mathbf{T}\mathbf{R}_x\mathbf{T}^{\mathrm{T}} = \mathbf{\Lambda} \tag{6}$$

where $\mathbf{\Lambda}$ is a diagonal matrix. The correlation matrix $\mathbf{R}_x$ is estimated from samples of $\mathbf{x}$ obtained from test images. The KLT matrix $\mathbf{T}$ is the 34×34 transform basis that diagonalizes the covariance matrix $\mathbf{R}_x$, and therefore the matrix of eigen vectors of $\mathbf{R}_x$. (The KLT we use is not *strictly* KLT; it is obtained *once* from many sample images. We obtain the 'KLT' from 15×1024 samples of $\mathbf{x}$ obtained from 15 standard test images of size 256×256).

3.1. Signature Design

We now consider the second issue of signature design. It is not very easy to *design* arbitrary signatures that can escape compression process. To detect a random sequence embedded on an image with a reasonable degree of certainty, the random sequence should be as close to orthogonal as possible to the *space* of images (or the space of the image coefficients in some transform domain). However, the job of a compression scheme is precisely to remove those components. So as explained earlier, with an ideal data compression scheme, no data hiding would be possible. An ideal data compressor would retain as much of the information content of the image as possible while chopping off the added information (corresponding to the hidden data) that does not enhance the visual quality of the image. So intelligent design of signatures would involve exploiting *holes* in practical compression tools.

It is well known that the magnitude DFT coefficients can be altered significantly without affecting the visual quality of the image. This implies that significant signature energy can be *squeezed* into the image while still maintaining its perceptual quality. Therefore, one can hope that the significant signature energy, which can remain *invisible* to the human visual system, would also *remain invisible to the compressor*, and thus *sneak* through the compressor. If this is true, then this is the second 'hole' in practical compression schemes that is utilized by magnitude DFT based data hiding schemes! Simulations indicate that this is indeed true. Recall that in Section II, we had indicated that intelligent design of signature sequences may involve a reduction in *degrees of freedom* for the choice of the signature. In the proposed scheme the reduction in the degrees of freedom is a direct result of the decomposition used. Note that addition of the signature *does not change the DFT phase.*

4. SELF NOISE SUPPRESSION

From Eq (5) it is obvious that it is very important to suppress as much as possible of the self-noise of the image. The decomposition $\mathcal{T}$ of the channels helps in solving the the problem to a certain extent by concentrating a significant portion of the self-noise into few coefficients of $\boldsymbol{c}$, and leaving a large number of coefficients with tolerable levels of self-noise. However, we can do better than that. We shall show, how *preprocessing* the coefficients $\tilde{\boldsymbol{c}}$ prior to detection of the signature $\boldsymbol{s}$ can substantially reduce the self-noise, thereby, improving the performance of the steganographic channel.

To understand the situation better, consider the following scenario. Let the value of an arbitrary coefficient of $\tilde{\boldsymbol{c}}$ be, say, 350. Further, it is known that the image could not have undergone drastic modification in the channel (for example, we know that the image could have only undergone lossy compression of reasonably good quality). We can now say with a high degree of certainty that the corresponding coefficient in the original image had a value between $350-\delta$ and $350+\delta$ ($\delta = 30$, for example). Though at first sight, the self-noise of the image appears to be directly related to the variance of the image coefficients, it should be appreciated that 'noise' is actually a measure of the *lack of information.* In other words, the entropy of the self-noise is equal to the entropy of the original image, *given the received image.* Mathematically, $\mathcal{H}$, the entropy of the self-noise is given by $\mathcal{H} = h(\mathbf{I} \mid \tilde{\mathbf{I}})$ bits, where $h(.)$ denotes the entropy [7]. Self noise suppression schemes should utilize the fact that the self-noise entropy $\mathcal{H}$ is substantially smaller than $h(\mathbf{I})$.

A self-noise suppression scheme, $D(.)$ uses an estimate of the original image (or its coefficients), obtained from the received image or its coefficients $\tilde{\mathbf{c}}$. $D(.)$ operates on $\tilde{\mathbf{c}}$ to produce an output

$$\tilde{\mathbf{c}}_e \quad = \quad D(\tilde{\mathbf{c}}) = \tilde{\mathbf{c}} - \mathbf{c}_s. \tag{7}$$

In the above equation, $\mathbf{c}_s$ is the *estimate* of $\mathbf{c}$. The buried information bits $\boldsymbol{b}$ can therefore be detected from $\tilde{\mathbf{c}}_e$ of Eq (7)

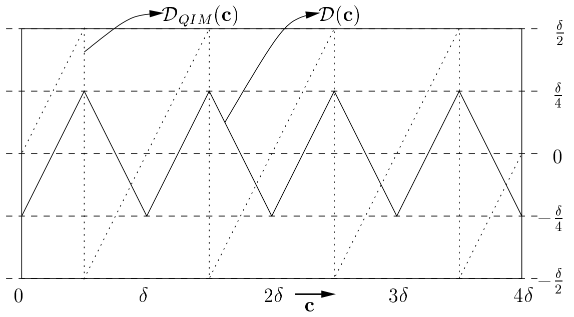

Figure 2: The mapping $\tilde{\boldsymbol{s}} = D(\tilde{\boldsymbol{c}})$ for our scheme (continuous line) and QIM (dotted line).

rather than $\tilde{\mathbf{c}}$. Therefore Eq (5) can be rewritten as

$$\tilde{\boldsymbol{b}} = \mathcal{S}^{-1}(\tilde{\mathbf{c}}_e) = \mathcal{S}^{-1}(\boldsymbol{c} - \mathbf{c}_s + \boldsymbol{s} + \boldsymbol{n}), \tag{8}$$

Probably the best *prediction* of the $\mathbf{I}$, under the given circumstances, is the received image $\tilde{\mathbf{I}}$ itself. But this does not help us in any way in terms of detecting $\mathbf{s}$. Subtracting $\tilde{\mathbf{I}}$ from itself will leave us with nothing to detect $\mathbf{s}$ from! What we actually need is the best possible prediction of $\mathbf{I}$, that is also *orthogonal* to $\mathbf{s}$.

A safe way out is to make no prediction at all. This guarantees that the predicted value (a vector of zeroes!) is trivially orthogonal to $\boldsymbol{s}$. But of course, we do not achieve any image noise suppression. So a compromise is called for here. If the prediction is coarse, we can expect a significant portion of the signature to remain in the prediction error. But the prediction error, and hence the self-noise is high. As the prediction gets less and less coarse, the self-noise reduces, but so does the component of the signature $\mathbf{s}$ in the prediction error. It is intuitive that some sort of quantization of the received signal $\tilde{\mathbf{I}}$ can be used as the predicted value. The step size of the quantizer used will depend on the level of distortion (or processing noise) the image has undergone. Higher the processing noise, larger should be the step-size of the quantizer used.

A *quantization index modulation* (QIM) scheme to reduce the image noise using a quantizer, was proposed by Chen *et. al.* in Ref. [8]. In this method, if $\boldsymbol{c}$ are the coefficients of the original image, and $\boldsymbol{s}$ is the signature to be added, $\hat{\boldsymbol{c}}$, the coefficients with the signature, is obtained as

$$\hat{\boldsymbol{c}} = Q(\boldsymbol{c} + \boldsymbol{s}) - \boldsymbol{s}, \tag{9}$$

where Q is a uniform quantizer with step size δ. Detection of the signature can then be performed as

$$\tilde{\boldsymbol{s}} = Q(\tilde{\boldsymbol{c}}) - \tilde{\boldsymbol{c}}. \tag{10}$$

The mapping performed by the detector of the self-noise suppression scheme can be represented by the dotted line in Figure 2.

In the self-noise suppression scheme employed in this paper, (bold line in Figure 2), the mapping $\tilde{s} = D(\tilde{c})$ is piece-wise continuous. Let $\mathbf{c}, \mathbf{s} \in \Re^K$. The algorithm for the mapping $D(.)$ is as follows:

$$\begin{aligned} q(k) &= \operatorname{rem}(\frac{|\tilde{c}(k)|}{\delta}), \quad k = 1 \cdots K \\ \tilde{s}(k) &= (q(k) \geq \frac{\delta}{2}) \quad ? \quad (\frac{3\delta}{4} - q(k)) \; : \; (q(k) - \frac{\delta}{4}) \end{aligned} \tag{11}$$

The quantizer step size δ is chosen based on the robustness needed. More severe the compression scenario (lower bit rate), higher the value of δ chosen. Let $\mathbf{p} = F(\mathbf{c})$. To introduce the signature $\mathbf{s}$, we need to modify $\mathbf{c}$ to obtain $\hat{\mathbf{c}}$ such that $\mathbf{s} = F(\hat{\mathbf{c}})$. To achieve this, the distortion $e(k)$ introduced in coefficient $c(k), k = 1 \cdots K$ is equal to

$$e(k) = |\hat{c}(k) - c(k)| = |s(k) - p(k)|. \tag{12}$$

However, the distortion introduced may not always be acceptable. For this reason we introduce a *visual threshold* $v(k)$ for each coefficient depending on the "activity" of a particular block (which in our case is just a measure of standard deviation of the pixels of a block). The visual threshold $v(k)$ limits the distortion that can be introduced to a coefficient.

The algorithm for embedding the signature sequence $\mathbf{s}$ in $\mathbf{c}$ is as follows:

$$\begin{aligned}
\text{if } e(k) &= (|s(k) - p(k)| < v(k)) \\
\text{then } e(k) &= s(k) - p(k) \\
\text{else } e(k) &= v(k)\text{sign}(s(k) - p(k)) \\
e(k) &= (p(k) > \delta/2) \quad ? \quad -e(k) \quad : \quad e(k) \\
\hat{c}(k) &= (c(k) \geq 0) \quad ? \quad c(k) + e(k) \quad : \quad c(k) - e(k)
\end{aligned} \tag{13}$$

5. SIGNALING SCHEMES

The problem of the choice of the signaling scheme can be stated as one of hiding n bits $b_1 \cdots b_n$ in K image data coefficients $\mathbf{c}$, where $K >> n$. As a simple approach which is referred to as bit-by-bit signaling, we have $\mathbf{c} = [\mathbf{c}_1\, \mathbf{c}_2 \cdots \mathbf{c}_n]$ and $\hat{\mathbf{c}} = [\hat{\mathbf{c}}_1\, \hat{\mathbf{c}}_2 \cdots \hat{\mathbf{c}}_n]$, where $\hat{\mathbf{c}}_l = \mathbf{c}_l + \text{sign}(b_l)\mathbf{s}$, and $\mathbf{s}$ is a random sequence of the same length as $\mathbf{c}_l$. The stego image with the signature is obtained as $\hat{\mathbf{I}} = \mathcal{T}^{-1}(\hat{\mathbf{c}})$, and is transmitted through the channel to obtain $\tilde{\mathbf{I}} = \mathcal{C}(\hat{\mathbf{I}})$. The coefficients $\tilde{\mathbf{s}} = F(\tilde{\mathbf{c}})$ (after self-noise suppression) are obtained, and the extracted bit sequence is calculated as

$$\tilde{b}_l = \begin{cases} 1 & \langle \tilde{\mathbf{s}}_l, \mathbf{s} \rangle > 0 \\ 0 & \langle \tilde{\mathbf{s}}_l, \mathbf{s} \rangle \leq 0 \end{cases} \tag{14}$$

On the other hand, we could generate 2^n sequences $\mathbf{s}_k, k = 1 \cdots 2^n$ of length K, such that the sequences $\mathbf{s}_k$ are *maximally separable*. Geometrically, the sequences $\mathbf{s}_k$ can be represented by a set of 2^n points in a K-dimensional hyper-sphere of a fixed radius (corresponding to the maximum energy of the signature permissible). If the sequences are maximally separable the minimum distance between any two of 2^n points should be as high as possible, under the given constraint of the hyper-sphere radius.

The binary sequence $[b_1\, b_2 \cdots b_n]$ is interpreted as a decimal number between 0 to $2^n - 1$. To transmit a particular sequence of the bits, whose decimal equivalent is say D, then the signature indexed by D, viz. $\mathbf{s}_D$ is added to the data coefficients $\mathbf{c}$; $\hat{\mathbf{c}} = \mathbf{c} + \mathbf{s}_D$. For detecting the hidden bits, the inner-product of the coefficients $\tilde{\mathbf{c}}$, of the received image are performed with each of the 2^n signature sequences.

$$\tilde{D} = \arg \max_{i=0 \cdots 2^n - 1} \langle \tilde{\mathbf{s}}, \mathbf{s}_i \rangle \tag{15}$$

While it is assured that the latter scheme, will approach the *channel capacity* closer than the former, in practice, implementation of the second scheme may be prohibitively expensive, especially for large K. A reasonable compromise would be to choose an alphabet size between 2 of the former (bit-by-bit signaling) scheme, and 2^n of the latter scheme. For example, if the alphabet size is chosen as $2^{\frac{n}{k}}$, then a single member of the alphabet is detected from each of the k sequences of length $\frac{K}{k}$.

In the proposed data hiding scheme we use a novel FFT based signaling scheme. in which the maximally separable sequences are constrained to be orthogonal

5.1. FFT Based Signaling Scheme

Let $\mathbf{c}_k$ of length $L_k = 2^{p-1}$ be a segment of the vector of coefficients $\mathbf{c}$. This segment will be used for hiding p bits. The maximally separable signature sequences $\mathbf{s}_k^l, l = 1 \cdots 2^p$ are obtained as 2^{p-1} orthogonal sequences and their negatives. *Random signature spaces* are generated from a seed. This is achieved by constraining the signatures to be *cyclic all-pass sequences.*

Let $\mathbf{h} \in \Re^N$, and $\mathbf{H} = \mathcal{F}(\mathbf{h})$ where, $\mathcal{F}(.)$ stands for the Discrete Fourier Transform (DFT). If $\mathbf{h}$ is such that $\mid H(n) \mid = 1$ for $n = 0, 1, \cdots, N-1$, then $\mathbf{h}$ and all its cyclic shifts are mutually orthogonal. In the proposed scheme, the maximally separated signature sequences are obtained from a random seed as follows:

1. From a random seed, generate a binary (± 1) sequence $\mathbf{b}_k$ of length L_k.
2. Obtain the length-L_k DFT $\mathbf{B}_k$ of the binary sequence.
3. Obtain $\mathbf{S}_k$ from $\mathbf{B}_k$ such that $|S_k(l)| = 1, l = 1 \cdots L_k$ and $\angle S_k(l) = \angle B_k(l), l = 1 \cdots L_k$.
4. Take the length-L_k IDFT of $\mathbf{S}_k$ to obtain $\mathbf{s}_k$. $\mathbf{s}_k$ is a *cyclic all-pass* function. All $L_k = 2^{p-1}$ cyclic shifts of $\mathbf{h}_k$ are orthogonal.
5. $\mathbf{s}_k$ and the other $2^{p-1} - 1$ cyclic shifts of $\mathbf{s}_k$, and their negatives are the 2^p maximally separable sequences.

Note that the inner product of the sequence $\mathbf{c}_k$ of length 2^{p-1} with each of the $2L_k = 2^p$ maximally separable sequences can be obtained by one length-L_k cyclic correlation efficiently implemented using the FFT. The index of the maximum absolute value of the cyclic correlation coefficients gives then detected sequence of p bits. Let $0 \leq D_k \leq 2^p - 1$ be the decimal representation of the binary sequence to be hidden in segment $\mathbf{c}_k$. The signature sequence $\mathbf{s}_{k_D}$ added to $\mathbf{c}_k$ is obtained as

$$\mathbf{s}_{k_D} = \begin{cases} \alpha \mathcal{C}(\mathbf{s}_k, D_k) & \text{if } D_k < 2^{p-1} \\ -\alpha \mathcal{C}(\mathbf{s}_k, D_k - 2^{p-1}) & \text{if } D_k \geq 2^{p-1} \end{cases} \tag{16}$$

where $\mathcal{C}(\mathbf{x}, a)$ stands for cyclic shift of the vector $\mathbf{x}$ by a (counterclockwise) positions, and α depends on the permissible distortion of the host signal due to addition of the signature. For detection,

$$\mathbf{R}_k = \mathcal{F}(\mathbf{s}_k)\mathcal{F}(\tilde{\mathbf{s}}_k) \quad \mathbf{r}_k = \mathcal{F}^{-1}(\mathbf{R}_k) \tag{17}$$

where $\mathcal{F}$ denotes the DFT, and,

$$\tilde{D}_k = \begin{cases} \arg\max_{i=0 \cdots 2^{p-1}-1} |r_k(i)| & \text{if } r_k(i) > 0 \\ \arg\max_{i=0 \cdots 2^{p-1}-1} |r_k(i)| + 2^{p-1} & \text{if } r_k(i) \leq 0. \end{cases}$$

To take advantage of the FFT based signaling we restrict ourselves to segment lengths which are powers of 2. If the total number of coefficients in $\mathbf{c}$ is say, 100000, we could use 97 segments of length 1024, which would result in 11 bits per segment, totaling to 1067 bits. If the processing the image undergoes is expected to be more severe, we would instead use 48 length-2048 segments, amounting to $12 \times 48 = 576$ bits.

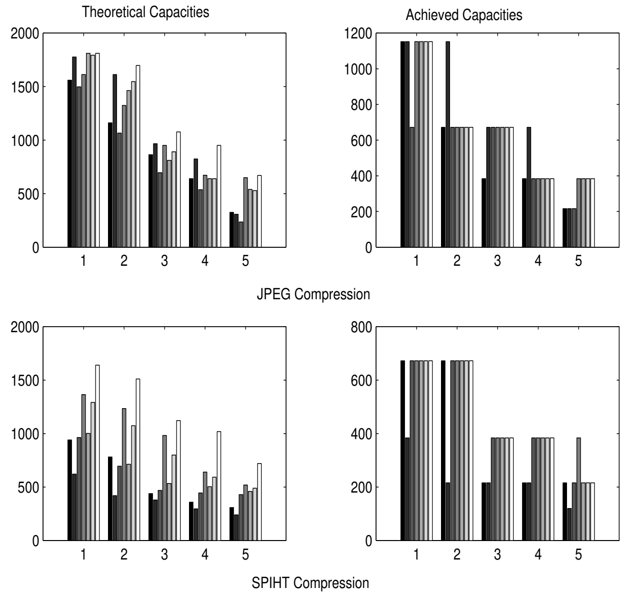

Figure 3: Plots of Theoretical (Left) and Achieved (Right) Capacities for JPEG (Top Row) and SPIHT (Bottom Row) Compression for 7 256×256 test images (Airplane, Barbara,Boats, Girl, Goldhill, Lenna and Peppers). JPEG compression scenarios 1 - 5 correspond to quality factors 75, 65, 55, 50 and 40 respectively. SPIHT compression scenarios 1 - 5 correspond to 1.35, 1.25, 1.15, 1.10 and 1.0 bpp respectively.

Smooth trade-offs between bit-rate and probability of error, can be achieved by redundant signaling, or signaling with parity bits. For example, if we use length 1024 segments with 2 parity bits for each segment, we reduce the bit rate from $97 \times 11 = 1024$ to $97 \times 9 = 873$. Introduction of 2 parity bits would render only one fourth of the points in the constellation as valid. Therefore, if the detected signal or point is not one of the permitted ones, we search for the next highest correlation value, till a permissible sequence (or member of the alphabet) is detected. This would reduce the probability of erroneous detection by a factor of 4. In general, using q parity bits would reduce the probability of error by a factor of 2^q.

There exists an efficient way of introducing parity in this scheme. For q parity bits, the valid points in the constellation are given by

$$D = \begin{cases} m2^q - 1 & D < L_k \\ m2^q & L_k \le D < 2L_k \end{cases} \quad m = 0, 1, \ldots, \frac{L_k}{2^q} \tag{18}$$

In this case, only coefficients of $\mathbf{r}_k$, with indices which are multiples of 2^q are needed. The required coefficients can be efficiently obtained by taking a $\frac{L_k}{2^q}$-length IFFT (instead of the L_k length IFFT) [9];

$$R_{q_k}(l) = \sum_{i=0}^{2^q-1} R_k(l + i\tfrac{L_k}{2^q}) \quad \mathbf{r}_{q_k} = \mathcal{F}^{-1}_{L_k/2^q}(\mathbf{R}_{q_k}) \quad \text{for } l = 0 \cdots \frac{L_k}{2^q} - 1. \tag{19}$$

6. RESULTS AND CONCLUSIONS

Experimental data hiding results obtained for 7 standard 256×256 test images are shown in Figure 3. The theoretical capacities in the left column of Figure 3 is calculated assuming a binary symmetric channel model. If p is the average probability of bit error and N is the total number of embedded bits, then the data hiding capacity is given by $N\mathcal{H}$, where

$$\mathcal{H} = 1 + p\log_2 p + (1-p)\log_2(1-p). \tag{20}$$

The achieved capacity in the right column of Figure 3 is the number of bits embedded in the image yielding *no bit error* over 25 simulations (different random seeds and hence different signal constellations were used for different simulations). We can expect to achieve capacities somewhere in between the *theoretical* and *achieved* capacities by using some error correction schemes. For this case, as erroneous detection of a member of an alphabet may result in many erroneous bits, Reed-Solomon codes may be the best alternative for error corection. The 7 vertical bars for each compression scenario correspond to test images Airplane, Barbara,Boats, Girl, Goldhill, Lenna and Peppers respectively.

The choice of processing scenarios 1 - 5 for JPEG compression and the corresponding choices for SPIHT compression have been made on the basis of effective bit-rates. On an average (obtained over 15 test images) JPEG compression at qualities 75, 65, 55, 50 and 50 resulted in bit rates of approximately 1.35, 1.25, 1.15 1.10 and 1.00 bpp respectively.

The results are extremely promising, and further refinement of the scheme for better utilization of the visual thresholds, and extensions of the scheme for data hiding in video frames, are topics of our current research.

7. REFERENCES

[1] A. H. Tewfik, White Paper on Data Embedding, Media Annotation and Copyright Protection Product Line, available for download from `http://www.cognicity.com`.

[2] M. Ramkumar, A.N. Akansu, "Information Theoretic Bounds for Data Hiding in Compressed Images", 1998 IEEE Second Workshop on Multimedia Signal Processing, Dec 7-9, Redondo Beach, California, USA, pp 267-272.

[3] M.Ramkumar, A.N Akansu, "Capacity Estimates for Data Hiding in Compressed Images", Submitted to IEEE Trans. of Image Processing, 1998.

[4] M.Ramkumar, A.N. Akansu, A. Alatan, "On the Choice of Transforms for Data Hiding in Compressed Video", IEEE-ICASSP, Mar 15 - 19, 1999, Phoenix, Arizona, Vol **VI**, pp 3049 - 3052.

[5] A. N. Akansu, R. A. Haddad, *Multiresolution Signal Decomposition: Transforms, Subbands and Wavelets.* Academic Press Inc., 1992.

[6] A.Said and W.A.Pearlman, "A New Fast and Efficient Implementation of an Image Codec Based on Set Partitioning in Hierarchical Trees", IEEE Transactions on Circuits and Systems for Video Technology, Volume **6**, pp. 243-250, June 1996.

[7] T. M. Cover, J. A. Thomas, *Elements of Information Theory*, Second Edition, John-Wiley and Sons Inc, 1991.

[8] B. Chen, G.W. Wornell, "Digital Watermarking and Information Embedding Using Dither Modulation", 1998 IEEE Second Workshop on Multimedia Signal Processing, Dec 7-9, Redondo Beach, California, USA, pp 273-278.

[9] M.Ramkumar, G.V. Anand and A. N. Akansu, "On the Implementation of 2-Band Cyclic Filterbanks", IEEE ISCAS-99, May 30 - Jun 02 1999, Orlando, Florida, Vol **III**, pp 520 - 523.