On the embedding limits of the discrete cosine transform

Tamer Rabie[1] · Ibrahim Kamel[1]

Abstract This paper investigates the embedding capacity limits of high-capacity data hiding in color images based on a locally Adaptive-Region Discrete Cosine Transform (AR-DCT) frequency domain data hiding scheme, and explores the relationship between hiding capacity and image quality. It also compares the embedding capacities of various steganography schemes which have been recently published in the literature. Experimental results confirm that our proposed scheme successfully enhances hiding capacity while maintaining acceptable image quality and concludes that the capacity for our DCT hiding scheme can achieve extremely high bit rates of 20 bits-per-pixel, which is much higher than other DCT-based approaches, as well as other spatial and frequency domain schemes.

1 Introduction

The proliferation and exchange of multimedia data over the internet and wireless networks has brought with it new prospects for covert communication. Data hiding techniques, commonly known as steganography when dealing with hiding secret messages into a cover medium to form a "stego" medium [27], or watermarking when copyright protection of

---

✉ Dr. Tamer Rabie
trabie@sharjah.ac.ae

Prof. Ibrahim Kamel
kamel@sharjah.ac.ae

[1] Department of Electrical and Computer Engineering,
University of Sharjah, Sharjah, UAE

multimedia data is involved [37], have received a great deal of attention in the past decade [6, 18, 23, 24, 33].

Motivated by growing concern about the protection of intellectual property on the Internet and by the threat of a ban for encryption technology, the interest in Steganographic Information Security (SIS) systems has been increasing over the years [26].

Given a secret message **sm** and a cover medium **f**, an SIS system uses two algorithms for communication between the transmitter and the receiver; an embedding algorithm to produce the stego-data $\hat{\mathbf{f}}$ (by embedding **sm** inside **f**); and an extraction algorithm to recover **sm** from $\hat{\mathbf{f}}$.

Techniques for data hiding inside digital images have been generally confined to three popular approaches, namely the spatial [5, 10], compression [9, 12] and frequency [11, 28–30] domains of the cover images, with variants that try to improve four different aspects; perceptibility, capacity, security, and robustness [13].

Perceptibility deals with the amount of "distortion" in the cover medium due to embedding information and if this information will lead to a visibly (visually or audibly) unacceptable level of the cover medium. Capacity refers to the amount of information that can be hidden in the cover medium relative to the change in perceptibility. For images, capacity is measured in bits per pixel (bpp). Security refers to an eavesdropper's inability to detect and inturn extract or change the hidden information, and robustness to the amount of modification the stego medium can withstand before an adversary can destroy the hidden information.

The main driving force behind data hiding in images is the fact that most images have inter-pixel relations that vary between high correlation and almost no correlation. The idea is to identify the redundancy in the pixel information of the cover image where the correlation is the least and use it to embed the information that we seek to hide.

One important problem related to embedding capacities in digital image media is the problem of automatically choosing an optimal-size cover image medium **f** from a database of K varying candidate cover images, f_i; $i = [1, K]$, that can accommodate the full size of a given secret message **sm** without leaving any unused redundancy in the cover medium, thus minimizing transmission bandwidth overhead. A solution to this problem requires the embedding algorithm to estimate the maximum embedding capacity limits for each of the candidate cover images f_i; $i = [1, K]$ and choosing the smallest size cover image with the optimal capacity that can be used to fully embed the secret message **sm** at hand. This paper presents a possible solution to this problem while investigating the embedding capacity limits of the discrete cosine transform.

Other aspects of importance in the data hiding field are losslessness and reversibility. A lossless steganography scheme is able to extactly retrieve the hidden data without any modification in its bit pattern, while reversibility is concerned with the ability to reconstruct the host cover medium after extraction of the hidden information with bit-by-bit exactness. Steganography and watermarking schemes satisfying this reversibility requirement are called reversible schemes. Reversibility is much more common in watermarking than in steganography. Reversible techniques for watermarking have drawn more and more interest in recent years, especially for some critical applications such as digital watermarking systems in the area of medical and military imaging and remote sensing where the watermarked media is required to be in exact form after watermark extraction [20, 22].

The image hiding scheme described in this paper is lossless but irreversible. The main advantages of lossy-irreversible steganography schemes is increased hiding capacities. For

the purpose of our work where both the hidden information and the host cover medium are color image data, we maintain losslessness of the hidden image while relaxing the reversibility constraint since we are mainly concerned with steganography applications and not watermarking. Nevertheless we show that extremely high capacities can be achieved for reasonable perceptibility taking into account the typically subjective nature of the measure of visual fidelity, which is our own visual perception.

The rest of this paper is organized as follows. In Section 2 we present prior work in the area of high-capacity data hiding. Section 3 briefly reviews the theory of the DCT transform and its wide use in image compression. Our proposed AR-DCT high-capacity data hiding scheme is discussed in Section 4, and Section 5 presents our comparison results and demonstrates the highest-capacity limits that can be achieved based on our approach. Finally, concluding remarks appear in Section 6.

2 Prior work

Embedding capacity for SIS systems has been a major area of research during the past few years [2, 3, 19, 21, 30] due to the fact that embedding capacities have remained relatively small, where researchers had to trade-off between higher capacity and reduced robustness and perceptual quality or higher perceptual quality and more robustness with lower capacities.

Spatial and time domain steganography methods are mainly broken into the following techniques: least significant bit (LSB) methods; Color palette methods [3, 36]; Minimax algebra decomposition; and Spatial delay (echoing in audio signals) methods. Transform domain methods include: Unitary transforms (discrete Fourier transform, discrete cosine transform); Wavelet transform methods; and Mellin-Fourier transform methods. Compression-based techniques include: Singular value decomposition; Use of edge and shape characteristics in images; and Fractal encoding in image data.

In the spatial domain approach, the secret message is embedded directly into the pixels of a cover image. The manipulation of the LSB of an image pixel value and the rearrangement of image colors to create LSB or parity bit patterns, which correspond to the message being hidden [15], are the most commonly used strategies in this approach.

An algorithm by Lee and Chen specifically designed for high capacity embedding is given in [19]. Their scheme modifies the LSB of pixels in an image and then attenuates the result to control the perceptibility. In Chang et al.'s scheme [10], a dynamic programming strategy is used to find the optimal LSB substitution in order to hide images. In addition to LSB-based hiding strategies, several schemes that use different strategies to hide secret messages in the spatial domains of cover images also have been proposed [19, 34]. For example, Chung et al. offered the singular value decomposition (SVD)-based hiding scheme [14], Tsai et al. used the bit plane of each block truncation coding (BTC) block to embed secret messages [35], and Chang et al. used the GA algorithm and absolute moment BTC to embed secret messages into color images [9].

In the frequency domain [11, 17, 20, 22, 28–30], the cover host image must first be transformed using a frequency-oriented mechanism such as the discrete Fourier transform (FFT), the discrete cosine transform (DCT), the discrete wavelet transform (DWT) or similar transformations, after which the secret messages can be combined with the coefficients in the frequency spectrum of the cover image to achieve embedding.

In [28] the author introduced a novel image hiding framework that makes use of the Fourier magnitude of the Luminance cover host image to maintain color composition of the stego image while allowing a relatively large size secret image (to a maximum of half the size of the color cover host image in both dimensions [30]) to be robustly embedded and extracted with acceptable perceptibility of the stego image and minor degradation in the extracted hidden data.

In Chang et al.'s scheme [7], the medium-frequency coefficients of DCT-transformed cover images are used to embed a secret message. The JPEG quantization table is also modified to further protect the embedded secret message. Similarly, Iwata et al. use the boundaries between zero and non-zero DCT coefficients to hide secret data [17]. In 2007, Chang et al. extended Iwata et al.'s idea and presented a lossless steganographic scheme for hiding secret data in each block of quantized DCT coefficients in JPEG images [11]. In Chang et al.'s scheme, the two successive zero coefficients of the medium-frequency components in each block are used to hide secret data. They further modified the quantization table to maintain the quality of the stego-image while concealing a higher payload compared with Iwata et al. scheme. Thus, their scheme achieves reversibility and acceptable image quality of the stego-image simultaneously. However, their scheme can only embed secret bits into the zero coefficients located in the successive zero coefficients in the medium area; non-zero coefficients in the medium area cannot be used.

In 2009, Lin and Shiu combined Chang et al.'s [11] scheme and then designed a 2-layers data hiding scheme for DCT-based images. Lin and Shiu's [20] scheme outperforms Chang et al.'s scheme [11] in hiding capacity but the size of the hidden secret data was still less than 70000 bits on average (equivalent to approximately embedding a 94×94 pixel gray-level image size inside a 512×512 gray-level cover image) because it retains the reversibility function.

Our work in this paper investigates the embedding capacity limits of high-capacity data hiding in color images based on an Adaptive-Region Discrete Cosine Transform (AR-DCT) frequency domain data hiding scheme and explores the relationship between hiding capacity and image quality. A comparison with the embedding capacities of various steganography schemes that have been recently published in the literature [3, 19, 21, 30, 38] is also demonstrated. It will be shown that our scheme takes the embedding capacity of the DCT to its limits where an extremely high embedding capacity of approximately 20 bpp can be achieved with reasonable perceptibility.

3 Mathematical background

In this section we detail the mathematical formulation that our steganography scheme is based on, and provide justification for the use of these formulas.

3.1 The discrete cosine transform

The one-dimensional Discrete Cosine Transform (1D-DCT) is often used in signal and image processing, especially for lossy data compression, because it has a strong "energy compaction" property: most of the signal information tends to be concentrated in a few low-
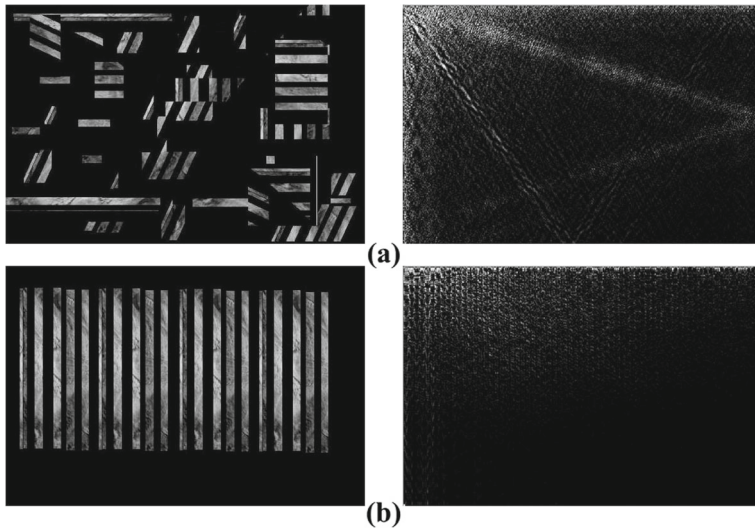
Fig. 1 **a** Uncorrelated image and its DCT; **b** Correlated image and its DCT

frequency coefficients of the DCT [1, 31]. Figure 1 shows an example of an uncorrelated image in (a) which has more sharp intensity variations than the correlated image in (b). Therefore, the former has more high frequency content than the latter. Clearly, the uncorrelated image has its energy spread out evident by the large values of the DCT coefficients well into the high-frequency bottom-right region, whereas the energy of the correlated image is packed into the low-frequency top-left region of its DCT.

The 2-dimensional DCT (2D-DCT) is a widely used image transformation, extended from the 1D-DCT, to compress digital images using the Joint Photographic Experts Group (JPEG) compression standard. The JPEG compression process consists of first transforming an (R,G,B) color image to a (Y,Cb,Cr) image where Y is the Luminance component and (Cb,Cr) are the two chrominance compnents. The transformed image is then subdivided into non-overlapping blocks of 8 × 8 pixels. The 2D-DCT is then performed on each block using (1). This generates 64 coefficients in the DCT domain which are then quantized to reduce their magnitude. The coefficients are then reordered into a one-dimensional array in a zigzag manner before further entropy encoding. The compression is achieved in two stages; the first is during quantization and the second during the entropy coding process. JPEG decoding is the reverse process of coding. An example 8 × 8 DCT basis functions (Left) and the zigzag reordering (Right) are shown in Fig. 2.

The forward 2D-DCT equation which calculates the coefficients $F(u, v)$, for an $N \times N$ block from gray-scale image $f(x, y)$ is:

$$F(u, v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)$$
$$\times \left[\cos\left(\frac{\pi u(2x + 1)}{2N} \right) \cos\left(\frac{\pi v(2y + 1)}{2N} \right) \right] \tag{1}$$
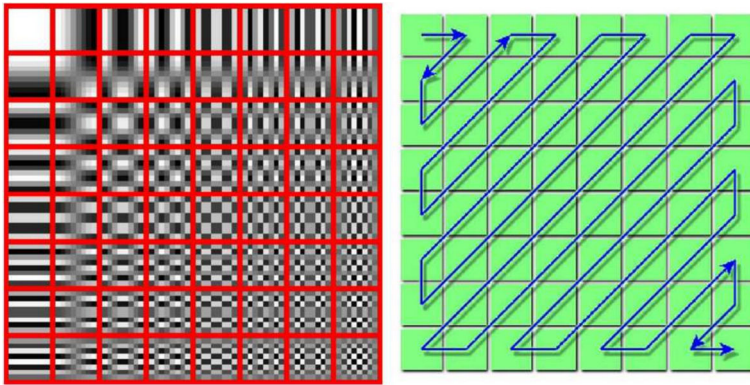
Fig. 2 *Left*: 8 × 8 2D-DCT basis functions. *Right*: zigzag reordering before encoding

and the inverse equation to retrieve the gray-scale $N \times N$ block for image $f(x, y)$ from 2D-DCT-coefficients $F(u, v)$ is obtained, as in (2), by multiplying the coefficients by the DCT basis functions:

$$f(x, y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v)$$
$$\times \left[\cos \left(\frac{\pi u(2x + 1)}{2N} \right) \cos \left(\frac{\pi v(2y + 1)}{2N} \right) \right] \quad (2)$$

where the basis functions are the cosines multiplied by the constant scale factors $C(u)C(v)$, where

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}} & if\ u = 0 \\ 1 & otherwise \end{cases} \quad (3)$$

and similarly for $C(v)$.

Referring to the left part of Fig. 2, each basis function from the 8 × 8 block is multiplied by its coefficient and then this product is added to the previous value to form the final image $f(x, y)$.

3.2 Stego image quality measures

In evaluating the performance of individual SIS systems it is important to take into consideration both the analytical performance of the system as well as the visual quality of the stego images generated by the system in comparison to the ideal cover image.

Detecting an embedded message defeats the primary goal of steganography, that of concealing the existence of a hidden message. As steganography is based on obscurity, the most important tests are related to our human perception, the ultimate measure of visual fidelity, which is very subjective.

The subjective tests are carried out by people who look for visual differences between the images (original and stego image) trying to find which one of them is the original. If the percentage of success reaches 50 %, it can be concluded that the message is invisible. The subjective test's rules and recommendations are defined by the International Telecommunication Union [16, 32].

Unlike the subjective approach which is based on human vision, the well known mean-square-error (MSE) metric calculates the global error variance (power in the difference image) between an ideal cover image $\mathbf{f}$, and the stego image $\hat{\mathbf{f}}$, and has been widely used for measuring the performance of various filters [4]. The only shortcoming in an MSE metric is that it is not ideal for tracking visual quality in the estimated image, because it is sensitive to minor pixel variations between the cover and stego images that do not, in general, affect the perceived visual quality.

A more robust measure of SIS system performance that has been widely used by the signal processing community is the Peak-Signal-to-Noise-Ratio (PSNR) in decibels (dB) [8, 25], given by:

$$\text{PSNR} = 20 \log_{10} \left(\frac{L-1}{\sigma_e} \right) \tag{4}$$

where L is the number of gray levels in an image ($L = 256$ for 8-bit images), and σ_e is the residual standard deviation in the error image given as:

$$\sigma_e = \sqrt{ \frac{1}{S} \sum_{s=0}^{S-1} (f_s - \hat{f}_s)^2 } \tag{5}$$

for S-sized images.

This PSNR metric is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. The PSNR is most commonly used to measure the quality of reconstruction in an image by comparing the stego image with the original image. This measure is less sensitive to minor deviations between images and will be adopted for comparing the various embedding schemes.

It is important to note, however, that objective measures such as the PSNR and MSE metrics are not necessarily correlated to our perception of an image. This is because methods that are least squares based are optimum in terms of MSE values without necessarily producing the best visual results.

4 The proposed scheme

The strong "energy compaction" property of the 2D-DCT for highly correlated image data suggests that we can make use of the high-frequency areas in the DCT domain to hide information, given that most of the signal information tends to be concentrated in a few low-frequency components of the DCT. It should be noted that the upper-left corner of the 2D-DCT represents the lower frequency coefficients with increasingly higher frequency components towards the lower-right corner. The fact that DCT coefficients of the lower-frequency areas can represent the correlated image information to a high degree of accuracy allows us to embed the hidden image into the high-frequency areas of the DCT with minimal effect on the perceptibility of the stego image.

The embedding capacity is nevertheless dependent on the amount of correlation in the cover image regions. As discussed in Section 3 and demonstrated by Fig. 1, an uncorrelated image region has its energy spread out in the DCT domain, whereas the energy of a correlated image region is packed into the low-frequency areas of its DCT. For cover images that represent natural scenes, any given part of the image generally differs sufficiently from the other parts so that the high-correlation assumption over the entire image is not generally valid. Thus, to achieve the highest embedding capacity it is imperative to assure high

correlation of cover image regions in which the secret image data will be embedded. In this work we develop an Adaptive-Region (AR) methodology to assure high correlation of cover image regions and seek to find the maximum possible secret image size that can be embedded into the DCT of the cover host image which will not reduce the perceptibility below the limit of 20 decibels (dB) for the peak-signal-to-noise-ratio (PSNR).

4.1 The embedding process

The Adaptive-Region Discrete Cosine Transform (AR-DCT) embedding approach we follow to demonstrate the hiding capacity limits of DCT-based steganography is influenced by the JPEG compression standard. To assure high correlation of cover image regions a smart-based technique is implemented where the cover host image is first divided into non-overlapping equal-sized $m \times m$ blocks[1] and the 2D-DCT is applied to each block using (1). Next, a maximum contiguous zero-valued square area of variable size $n \times n$; $n < m$ in the high-frequency region (lower-right corner) of each individual DCT block of the cover host image is adaptively estimated by the embedding function. The modal value of these varying $n \times n$ sized regions in the individual DCT blocks is computed, and it is this final modal size $n \times n$ that is used in all DCT blocks for embedding the same-sized $n \times n$ corresponding regions from the secret image.

   The embedding is performed by replacing each $n \times n$ region of the DCT blocks by the corresponding $n \times n$ region from the secret image after first re-scaling this secret image block's gray-level values to the range $[0, k]$. This re-scaling step is necessary to allow the hidden secret image information to blend into the natural range of values of the DCT coefficients. The maximum scale value k is empirically estimated and is found to be in the range $k = [10, 20]$ for typical natural RGB color images. This is repeated for all DCT blocks of the cover host image. Figure 3 shows pseudo code of the embedding algorithm.

   The technique used to adaptively estimate the largest contiguous square area of variable size $n \times n$ in the high-frequency region of the individual DCT blocks of the cover host image is depicted in Fig. 4. The idea is to use a quantization step to help the embedding function to automatically estimate the maximum area in the lower-right (high-frequency) corner of the DCT of the current block in the cover image. The secret image is then directly embedded by replacing the zero DCT coefficients in this square area by the $k-$scaled values in the current block of the secret image information.

   Quantization of each individual $m \times m$ cover image DCT block is computed by dividing the elements of this block by the elements of a quatization matrix of the same size. The quatization matrix values used are shown in Fig. 4 and are simply one of the many standard quatization matrices used for JPEG compression. The purpose of this quantization step is to keep intact the important DCT coefficients which can be used to reconstruct the original cover host image while masking out the coefficients which provide less important (redundant) information which can be removed to compress the image (as in the lossy JPEG standard) or replaced by the hidden information as is the case with steganography.

   The block size we use is not limited to 8×8 as in the JPEG method, but this block size is allowed to vary to the maximum cover image size (512×512 for our experiments) in multiples of the smallest size 8×8 (i.e. 16×16, 32×32, $\cdots$, 256×256, 512×512). In

---

[1]For JPEG standard the block size $m \times m$ is 8×8

```
Algorithm: [f'] = ARdctEmbed (f, sm, m, k)
Input:      f:  color RGB cover host image
            sm: secret message color RGB image to be hidden
            m:  square block size to use: [m x m]
            k:  scale factor empirically chosen between [10,20]
            QM: 8 x 8 quantization matrix values
Output:     f': stego color RGB image containing the secret image

 1: resize(QM, [m x m]) //resize to same size as block size [m x m]
 2: fi ← divide(f, m) //divide cover image into N non-overlapping [m x m] blocks
 3: si ← divide(sm, m) //divide secret image into N non-overlapping [m x m] blocks
 4: for i := 1 to N blocks
 5:   fiDCT ← dct2(fi)    //apply 2D DCT on each [m x m] block
 6:   fiQdct ← fiDCT/QM   //quantize each block by dividing my QM
 7:   [ni x ni] ← find(fiQdct) // find largest block of zero dct coefficients
 8: end for
 9: [n x n] ← mode([ni x ni]) //mode of all blocks of zero-region dct coefficients
10: for i := 1 to N blocks
11:   resize(si, [n x n]) //resize secret image blocks to same block size [n x n]
12:   si' ← rescale(si, [0,k]) //rescale pixel values to be in the range [0,k]
13:   fiDCT' ← replace([n x n], fiDCT, si') //replace lower-right [n x n] with si'
14:   fi' ← idct2(fiDCT') //apply inverse 2D DCT on each [m x m] block
15:   f' ← pack(fi') // combine all [m x m] blocks into one stego image
16: end for
17: return f'  // the final stego image
```

Fig. 3 Pseudo Code of the AR-DCT embedding algorithm

cases where the block size used is larger than 8×8, the standard 8×8 quantization matrix indicated in Fig. 4 is still used after first interpolating its values to the larger $m \times m$ block size. It should be noted that the larger the block size we used the less the JPEG-like blocky artifacts appeared in the final stego image.

The Inverse 2D-DCT given by (2) is then applied to the modified DCT coefficients containing the hidden color image information to produce the color stego image. This is stored in the highest quality (lossless) JPEG image format to maintain the hidden information from being modified.

4.2 The extraction process

Extraction is performed in the reverse order where the 2D-DCT of the stego image is taken and, knowing the hidden block size $n \times n$ apriori, the same $n \times n$ sized lower-right region of the DCT coefficient blocks are extracted and rescaled to the original intensity range ([0, 255] per color channel). For extra security, an encryption step may be introduced before embedding the hidden image during the hiding stage, and a decryption key must then be known apriori to be able to correctly extract the exact hidden information.

5 Experimental comparisons

In this section we present comparative results of our AR-DCT scheme against various other steganography schemes which have been recently published in the literature. We also present detailed demonstrative results of our scheme applied at seven different block size scale levels ($8 \times 8, 16 \times 16, 32 \times 32, \cdots, 256 \times 256, 512 \times 512$), and tested on four different cover host images.

One possible application scenario demonstrating the utility of our technique is described towards the end of this section, and finally experimental results showing effects on the
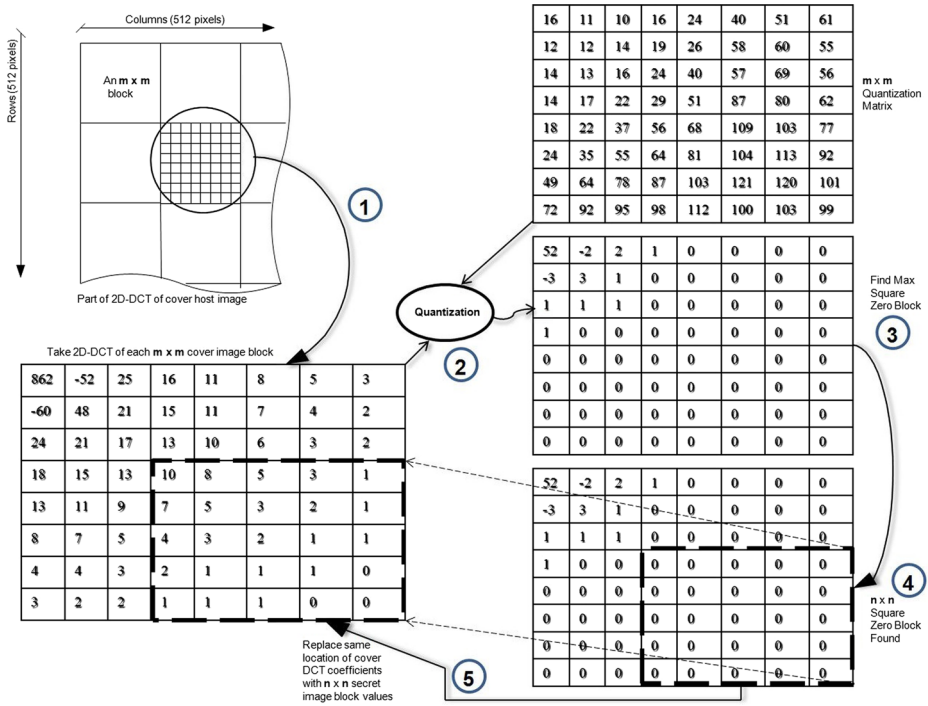
Fig. 4 Example showing quantization of each individual $m \times m$ ($m = 8$) cover image DCT block by dividing the elements of each DCT block of the cover image by the elements of a quatization matrix of the same size and then adaptively selecting the largest square zero region which was found to be of size $n \times n$ (for this example $n = 5$ shown as the shaded area)

quality of the recovered hidden secret image using our scheme when the stego image has been contaminated by different types of noise is also presented.

5.1 Comparative results

Our adaptive-region (AR-DCT) embedding scheme, described in Section 4 above, is able to embed, in a typical cover image of a natural scene, and losslessly extract, from the generated stego image, approximately 6.125 bpp per color channel of a three channel (R,G,B) color cover image for a maximum overall 18.375 bpp embedding capacity with a perceptibility measured at PSNR of 25.8 dB. This corresponds to embedding a color image of size 448 × 448 inside a color cover image of size 512 × 512 with an 87.5 % embedding capacity. When taking our scheme to the limits and forcing the embedding capacity in the DCT of the cover image to a size increase of 32 × 32 above the estimated adaptive region which takes the embedded color hidden image size to 480 × 480, the average PSNR was still 23.2 dB for a total embedding capacity of 7.03 bpp per color channel which is equivalent to 21.1 bpp in the three color channels of an (R,G,B) color cover image.

The algorithm by Lee & Chen specifically designed for high capacity embedding is given in [19]. Their scheme modifies the least significant bits (LSBs) of pixels in an image and then attenuates the result to control the perceptibility. They were able to embed and extract a

maximum capacity of 4.06 bpp from a gray-scale image which can be extrapolated to 12.18 bpp for an (R,G,B) color image (equivalent to 364×364 hidden color image size embedded in a 512×512 color cover image size). The perceptibility for this method was at a PSNR of 34.03 dB.

The maximum capacity that was reached in the work of Brisbane et al. [3] was an average 6 bpp for an (R,G,B) color cover image, with a perceptibility given by a PSNR of 40 dB. This is equivalent to 2 bpp per color channel (i.e. 256×256 hidden color image embedded in 512×512 color cover image size).

The maximum possible capacity reachable using the FFT Magnitude of the Luminance channel approach described in our previous work in [30] was 6 bpp for hiding an (R,G,B) color image in an (R,G,B) color cover host image, with a maximum perceptibility given by a small PSNR of 19.51 dB. This is equivalent to 256×256 hidden color image embedded in 512×512 color cover image size. In comparison the DCT high-capacity steganography scheme described in this work takes the embedding capacity of the DCT to its limits and shows that an extremely high embedding capacity of approximately 20 bpp can be achieved with reasonable perceptibility.

Other DCT-based information hiding techniques include the work by Yang et al. [38] and Lin & Shiu [21]. In [38] they present a high capacity reversible watermarking scheme and use the bit-shift operation of companding technique over integer DCT coefficients of image blocks. The maximum data hiding capacity they reached was 173123 bits for a gray-scale image which can be extrapolated to 1.96 bpp for an (R,G,B) color image at PSNR of 28.16 dB which is equivalent to hiding a color image of size 147×147 inside a 512×512 color host image. While in [21] the maximum achievable hiding capacity was 90112 bit which is equivalent to a hidden image of size 106×106 gray pixels inside a 512×512 gray cover host image which translates to about 0.34 bpp per channel which can be extrapolated to 1.02 bpp for a three-color image at PSNR of 28.22 dB.

Table 1 shows the different capacity limits of various schemes discussed in this section in comparison to our AR-DCT scheme which clearly shows the increased capacity that can be achieved by our scheme with reasonable PSNR perceptibility. It is also clear from the PSNR values of our scheme that the rate of decrease in PSNR is much slower than the rate of increase in the capacity achieved; inceasing the capacity $3\times$ (from **6 bpp** to **18.4 bpp**), only decreases the PSNR from 28.7 dB to 25.8 dB. Similarly, increasing the capacity $3.5\times$ (from **6 bpp** to **21.1 bpp**) only decreases the PSNR from 28.7 dB to 23.2 dB, where the increased capacity outweighs the small decrease in PSNR.

Table 1 Comparative results expressed as maximum Capacity/PSNR values for the various methods. Highest Capacities and PSNR values are emphasized in a bold font

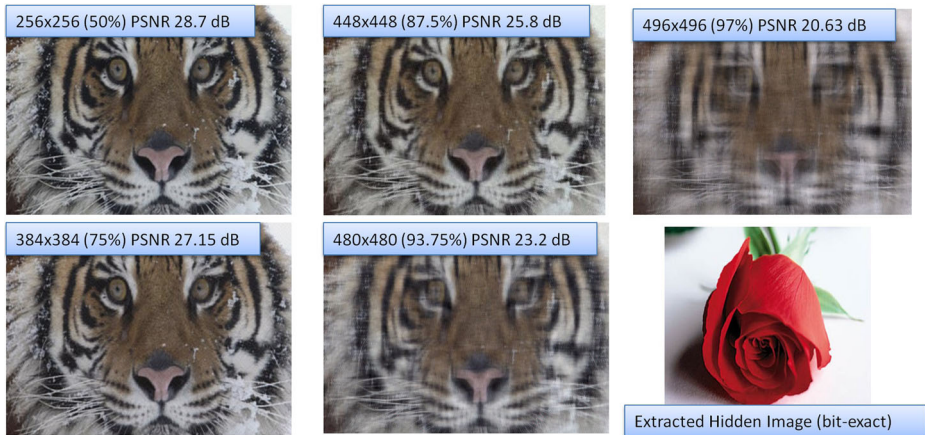| Method	Capacity	PSNR
Lee & Chen [19]	12.18 bpp	34.03 dB
Yang et al. [38]	1.96 bpp	28.16 dB
Brisbane et al. [3]	6 bpp	**40 dB**
Lin & Shiu [21]	1.02 bpp	28.22 dB
Rabie (FFT) [30]	6 bpp	19.51 dB
AR-DCT (Forced-Under)	6 bpp	28.7 dB
AR-DCT (Adaptive-Capacity)	**18.4 bpp**	25.8 dB
AR-DCT (Forced-Over)	21.1 bpp	23.2 dB

Fig. 5 Stego image portion from a 512 × 512 "Tiger Face" image used as a cover host to embed the different sizes of the "Red Rose" image indicated on each image using our AR-DCT scheme

5.2 Results of proposed method

Figure 5 shows the stego image of a "Tiger Face" image used as a cover host to embed the different sizes of the Red Rose image indicated on each image using our technique. These are the values presented in Table 1 which demonstrate the capacity limits of our AR-DCT scheme. The figure clearly shows that the hiding capacity of the DCT continues to produce acceptable perceptibility when working below 90 % (approximately 20 bpp) embedding capacity where PSNR values remain above 25 dB.

Next we show results for block sizes below the full frame size of the cover host image. Figure 6 shows results of capacities and PSNR values estimated by our AR-DCT scheme for four different stego images of size 512 × 512 representing the cover host images ("Peppers", "Zebras", "Balloons", and "Snow Tiger") having different levels of correlation in the scenes that they capture. In Fig. 6-(a) we use 8 × 8 block sizes, and in (b) we use 16 × 16 block sizes for the secret image embedding using our AR-DCT method. Highest embedding capacities estimated by our AR-DCT scheme for each block size are indicated in underlined bold font.

Table 2 presents capacities and PSNR values estimated by our AR-DCT scheme applied at seven different block size scale levels (8 × 8, 16 × 16, 32 × 32, · · · , 256 × 256, 512 × 512), and tested on the four different cover host images represented in Fig. 6. The data in Table 2 can also be interpreted graphically by plotting the capacities and PSNR values of each of these four test images with respect to the block size scale levels as shown in Fig. 7.

It is clear from Table 2, that using different block sizes for our AR-DCT embedding process results in different embedding capacities relative to each cover host image used. This is due to the varying scenes that each cover image represent and the varying scales of correlated objects in each scene. For example, if the size of correlated objects in a given cover image are within the size 64 × 64 scale-level, then a 64 × 64 block size would be the most suitable to be used for the AR-DCT scheme to produce the optimal capacity for this specific cover host image. This can be seen very clearly from the very similar capacities (18 bpp) and the similar PSNR values (25 dB) of the "Peppers" stego image at both the 64 × 64 and the 256 × 256 block sizes, as shown in Table 2, indicating that correlated objects in this
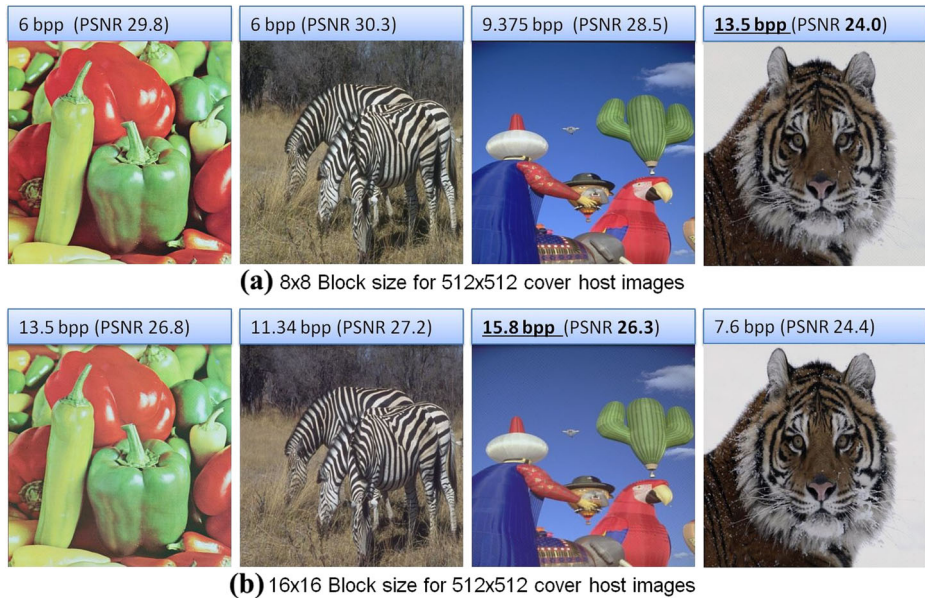
| 6 bpp (PSNR 29.8) | 6 bpp (PSNR 30.3) | 9.375 bpp (PSNR 28.5) | **13.5 bpp** (PSNR **24.0**) |

(a) 8x8 Block size for 512x512 cover host images

| 13.5 bpp (PSNR 26.8) | 11.34 bpp (PSNR 27.2) | **15.8 bpp** (PSNR **26.3**) | 7.6 bpp (PSNR 24.4) |

(b) 16x16 Block size for 512x512 cover host images

Fig. 6 Capacities and PSNR of various Stego images of size 512×512: **a** using 8×8 block size for the AR-DCT method, and **b** using 16×16 block size for the AR-DCT method. Highest embedding capacities estimated by our AR-DCT scheme for each block size are indicated in underlined bold font. (From Left to Right: "Peppers", "Zebras", "Balloons", and "Snow Tiger" Stego Images)

image lie within these two scale-levels (64×64 and 256×256), which can also be verified by visual inspection of the "Peppers" image from Fig. 6.

5.3 Example application scenario

The utility of our AR-DCT steganography scheme can be demonstrated by an example application scenario;

- Given that we have an RGB color secret message image of size 405×405 pixels, which translates to **15 bpp** in total for the three color RGB channels, find all possible cover host images from a database of candidate cover images of varying sizes that can host the given secret message image with a waste tolerance of **1 bpp** in storage bandwidth (i.e. cover images can be chosen with a maximum capacity limit of **16 bpp**).
- Using our scheme for the example database of four images shown in Table 2, the possible cover host images that can accommodate the given secret image are:

    – *Balloons* cover image at a capacity of **15.8 bpp**, with embedding at 16×16 block size, and a PSNR of 26.3 dB
    – *Peppers* cover image at a capacity of **15.8 bpp**, with embedding at 32×32 block size, and a PSNR of 26.36 dB
    – *Zebras* cover image at a capacity of **15.24 bpp**, with embedding at 64×64 block size, and a PSNR of 25.29 dB

Table 2 Application of our AR-DCT steganography scheme showing capacities and PSNR of various Stego images of size 512 × 512

Block Size	Cover Image	Capacity (bpp)	PSNR (dB)
	Peppers	6	29.8
8 × 8	*Zebras*	6	**30.3**
	Balloons	9.38	28.5
	Snow Tiger	**13.5**	24.0
	Peppers	13.5	26.8
16 × 16	*Zebras*	11.34	**27.2**
	Balloons	**15.8**	26.3
	Snow Tiger	7.6	24.4
	Peppers	15.8	26.36
32 × 32	*Zebras*	13.5	25.98
	Balloons	**19.71**	**26.45**
	Snow Tiger	18.375	22.66
	Peppers	**18.375**	25.5
64 × 64	*Zebras*	15.24	25.29
	Balloons	12.4	**27.65**
	Snow Tiger	12.4	23.64
	Peppers	**19.04**	25.18
128 × 128	*Zebras*	18.375	21.74
	Balloons	16.46	**26.55**
	Snow Tiger	16.15	23.39
	Peppers	18.05	**25.79**
256 × 256	*Zebras*	21.62	16.7
	Balloons	19.04	24.84
	Snow Tiger	**22.16**	20.4
	Peppers	20.22	**24.3**
512 × 512	*Zebras*	22.16	15.73
	Balloons	**22.52**	22.26
	Snow Tiger	18.62	22.83

Highest Capacity and PSNR values for each block size level are emphasized in a bold font

- Our scheme would thus choose the *Peppers* cover image as a host for embedding the secret image of size 405 × 405 because of higher PSNR (26.36 dB) compared to the other two candidate cover images.

5.4 Effects of noisy stego image

In this section we discuss the quality of recovering the hidden secret image using our AR-DCT scheme when the stego image has been contaminated by different types of noise. For this test we experiment with contaminating the stego image (generated after hiding the secret image into the cover image) using three different types of noise; Gaussian noise, Salt & Pepper noise, and Speckle noise as shown in Fig. 8.
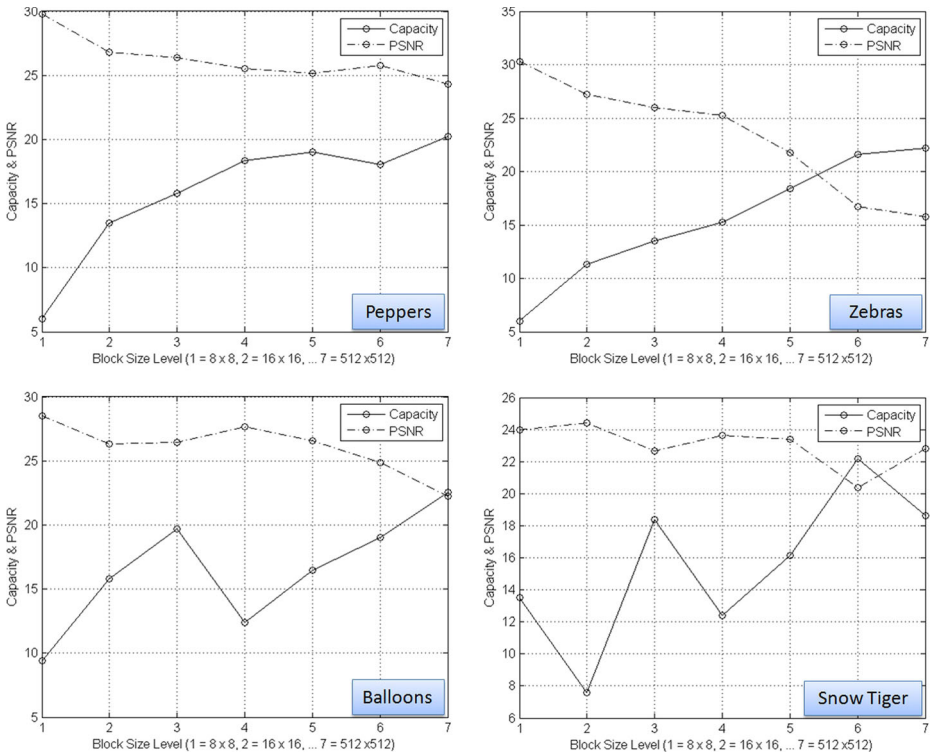
Fig. 7 Variation of Capacity and PSNR with AR-DCT Block Size for each test cover image used

Gaussian noise is additive white zero-mean noise with a Gaussian distribution and noise power given by the variance in its distribution. Salt & Pepper noise which adds sparse "On and Off" random pixels of density $\mathbf{d}$, where $0 < \mathbf{d} < 1$, to an image of size $M \times N$ will affect approximately $\mathbf{d} \times M \times N$ pixels in this image. Speckle noise adds multiplicative noise to the stego image $\hat{\mathbf{f}}$ to produce a noisy stego image $\mathbf{g}$ using the equation $\mathbf{g} = \hat{\mathbf{f}} + \mathbf{n} * \hat{\mathbf{f}}$, where $\mathbf{n}$ is uniformly distributed random noise of the same size as the image with zero-mean and noise power given by the variance in its distribution. Noise variance for Gausian and Speckle noise, and noise density $\mathbf{d}$ for Salt & Pepper noise have been unified at gradually increasing values of 0.0001, 0.0005, and 0.001 for proper comparison.

It should be noted, however, that DCT-based steganography methods are not at all suited to dealing with any kind of noise due to the highly uncorrelated nature of the noise random variable itself. As was discussed previously, and demonstrated by Fig. 1, uncorrelated signals have their energy spread out in the DCT domain with large values of the DCT coefficients well into the high-frequency regions. Since the general idea of our AR-DCT scheme is to embed the secret image into the high-frequency regions of the DCT of the cover image, contaminating this stego image with a high-frequency uncorrelated noise component would have the effect of decorrelating the stego image information which would affect the hidden secret image during the extraction process. Adding noise to the stego image would cause its high-frequency DCT coefficients to have larger values which in turn contaminates the hidden secret image data residing at this high-frequency location in the DCT of the stego image.

Fig. 8 Extraction of the hidden secret "Red Rose" image from a noise contaminated "Peppers" stego image (only a magnified portion is shown) using Gaussian, Salt & Pepper, and Speckle noise. Noise variance for Gausian and Speckle, and noise density for Salt & Pepper have been unified at gradually increasing values of 0.0001, 0.0005, and 0.001 for proper comparison. PSNR values of the extracted secret image are indicated under each image

6 Conclusions

The authors have presented an adaptive-region discrete cosine transform (AR-DCT) high-capacity image embedding framework that exceeds the hiding capacity of other spatial and frequency domain data hiding schemes while maintaining very acceptable perceptibility of the stego image. The idea is to use a quantization step to help the embedding function to adaptively estimate the maximum area in the lower-right (high-frequency) corner of the DCT of the current block in the cover image. The secret image block is then directly embedded in this square area replacing the cover host image's DCT coefficients in the same block location. It was shown that our scheme takes the embedding capacity of the DCT to its limits where an extremely high embedding capacity of approximately 20 bpp can be achieved with reasonable perceptibility. One application of our high-capacity steganography scheme is automatically selecting the optimal-size cover host image from a database of candidate cover images that can accommodate the full size of a given secret message without leaving any unused redundancy in the cover medium, thus minimizing transmission bandwidth

overhead. One possible solution to this problem was demonstrated by an example application scenario. Another potentially attractive application is video privacy protection where an important video sequence may be embedded frame-by-frame in the individual frames of another less important video thus securing its privacy.

References

1. Ahmed N, Natarajan T, Rao K (1974) Discrete cosine transform. IEEE Trans Comput 23(1):90–93
2. Anderson RJ, Petitcolas FA (1998) On the limits of steganography. IEEE Journal on Selected Areas in Communications 16(4):474–481
3. Brisbane G, aini RSN, Ogunbona P (2005) High-capacity steganography using a shared colour palette. IEE Proc, Vis Image Signal Process 152(6):787–792
4. Castleman K (1996) Digital Image Processing. Prentice Hall, Upper Saddle
5. Celik MU, Sharma G, Tekalp AM, Saber E (2005) Lossless generalized-lsb data embedding. IEEE Trans Image Process 14(2):253–266
6. Chan CK, Cheng L (2004) Hiding data in images by simple LSB substitution. Pattern Recog 37:469–474
7. Chang CC, Chen TS, Chung LZ (2002) A steganographic method based upon jpeg and quantization table modification. Inform Sci 141:123–138
8. Chang CC, Chen TS, Chung LZ (2002) A steganographic method based upon jpeg and quantization table modification. Inform Sci 141(1):123–138
9. Chang CC, Chen YH, Lin CC (2008) A data embedding scheme for color images based on genetic algorithm and absolute moment block truncation coding. Soft Comput 13:21–331
10. Chang CC, Hsiao JY, Chan CS (2003) Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy. Pattern Recog 36(7):1595–1683
11. Chang CC, Lin CC, Tseng CS, Tai WL (2007) Reversible hiding in dct-based compressed images. Inform Sci 177:2768–2786
12. Chang CC, Tai WL, Lin CC (2006) A reversible data hiding scheme based on side match vector quantization. IEEE Trans Circuits and Systems for Video Technology 16(10):1301–1308
13. Chen B, Wornell G (2001) Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. IEEE Trans Inf Theory 47(4):1423–1443
14. Chung KL, Shen CH, Chang LC (2001) A novel svd- and vq-based image hiding scheme. Pattern Recogn Lett 22(9):1051–1058
15. Curran K, Bailey K (2003) An evaluation of image based steganography methods. International Journal of Digital Evidence 2(2):1–40
16. IEC I (1994) Information technology-digital compression and coding of continuous-tone still images: Requirements and guidelines. Standard, ISO IEC pp 10,918–1
17. Iwata M, Miyake K, Shiozaki A (2004) Digital steganography utilizing features of jpeg images. IEICE Trans Fundamentals E87-A(4):929–936
18. Jain A, Uludag U, Hsu R (2002) Hiding a face in a fingerprint image. In: Proceedings of the International Conference on Pattern Recognition (ICPR). Quebec City, Canada
19. Lee Y, Chen L (2000) High capacity image steganographic model. IEE Proc, Vis Image Signal Process 147(3):288–294
20. Lin CC, Shiu PF (2009) Dct-based reversible data hiding scheme. In: Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication (ICUIMC09), pp 327–335
21. Lin CC, Shiu PF (2010) High capacity data hiding scheme for dct-based images. Journal of Information Hiding and Multimedia Signal Processing 1(3):220–240
22. Lin CY, Chang CC, Wang YZ (2008) Reversible steganographic method with high payload for jpeg images. IEICE Trans Inf Syst 91-D(3):836–845
23. Marvel LM, Charles G, Boncelet J, Retter CT (1999) Spread spectrum image steganography. IEEE Trans Image Processing 8(8):1075–1083
24. Nozaki K, Niimi M, Eason RO, Kawaguchi E (1998) A large capacity steganography using color bmp images. In: ACCV '98: Proceedings of the Third Asian Conference on Computer Vision-Volume I, pp 112–119. Springer-Verlag, London
25. Pavlidis G, Tsompanopoulos A, Papamarkos N, Chamzas C (2003) Jpeg2000 over noisy communication channels thorough evaluation and cost analysis. Signal Process Image Commun 18(6):497–514
26. Petitcolas FA, Anderson RJ, Kuhn MG (1999) Information hiding-A survey. Proc IEEE 87(7):1062–1078

27. Provos N, Honeyman P (2003) Hide and seek: An introduction to steganography. IEEE Computer Society
28. Rabie T (2007) Frequency-domain data hiding based on the matryoshka principle. Special Issue on Advances in Video Processing and Security Analysis for Multimedia Communications. International Journal of Advanced Media and Communication 1(3):298–312
29. Rabie T (2012) Digital image steganography: An fft approach. In: 4th International Conference on Networked Digital Technologies (NDT), pp 217–230. Springer Verlag
30. Rabie T (2013) High-capacity steganography. In: 6th International Congress on Image and Signal Processing (CISP), vol 2, pp 858–863
31. Rao K, Yip P (1990) Discrete Cosine Transform: Algorithms, Advantages, Applications. Academic Press, ISBN 0-12-580203-X, Boston
32. Rodrigues J, Rios J, Puech W et al (2004) Ssb-4 system of steganography using bit 4. In: 5th International Workshop on Image Analysis for Multimedia Interactive Services
33. Solanki K, Jacobsen N, Madhow U, Manjunath BS, Chandrasekaran S (2004) Robust image-adaptive data hiding using erasure and error correction. IEEE Trans Image Process 13(12):1627–1639
34. Tian J (2003) Reversible data embedding using a difference expansion. IEEE Trans Circuits and Systems for Video Technology 13(8):890–896
35. Tsai P, Hu YC, Chang CC (2002) An image hiding technique using block truncation coding. In: Proceedings of Pacific Rim Workshop on Digital Steganography, pp 54–64
36. Wang X, Yao Z, Li CT (2005) A palette-based image steganographic method using colour quantisation. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), pp II – 1090–3
37. Wu M, Liu B (2003) Data hiding in image and video: Part I - fundamental issues and solutions. IEEE Trans Image Process 12(6):685–695
38. Yang B, Schmucker M, Funk W, Busch C, Sun S (2004) Integer dct-based reversible watermarking for images using companding technique. In: Proceedings of the SPIE 5306, Security, Steganography, and Watermarking of Multimedia Contents. vol 6 (405)

Tamer Rabie (S'89-M'99-SM'03) received the M.Sc. degree in adaptive image restoration in 1993 from the Department of Electrical and Computer Engineering at The University of Calgary in Canada, and the Ph.D. degree in active computer vision applied to dynamic virtual environments in January 1999 from the Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto in Canada, where, in collaboration with world-renowned computer scientist Professor Demetri Terzopoulos, they pioneered the Animat Vision Paradigm, widely cited today in the virtual vision literature. During the period from May 1998 to September 1998 Dr. Rabie held a five-month software engineer contract with the visualization and image processing group at ISG Technologies Inc. in Mississauga, Ontario, Canada, where he was involved in a critical project to develop image quality enhancements for leading edge volume rendering servers for the medical imaging industry. From October 1998 to December 1999 he held a postdoctoral fellow in the Department of Computer Science at the University of Toronto where he was involved in further research and development of the Animat Vision project. From January 2000 to August 2001 he held an assistant professor appointment in the Department of Electrical and Computer Engineering at Ryerson University in Toronto, Canada, and an associate professor position in the Faculty of Information Technology at the UAE University from September 2001 to 2011. He is currently Associate Professor of Computer Engineering at the University of Sharjah's Electrical & Computer Engineering Department. He is also an adjunct professor at the University of Toronto's Intelligent Transportation Systems Centre, conducting collaborative research work in active computer vision-based traffic surveillance and control. His current research interests include digital image processing, computer vision, image/speech watermarking, and intelligent robotic systems. Dr. Rabie has published papers in digital image processing, computer vision, and artificial intelligence.

Prof. Ibrahim Kamel received Ph.D. degree in Computer Science from University of Maryland, College Park, USA in 1994, and B.Sc. in Computer Engineering from University of Alexandria, Egypt in 1984. Currently, Prof. Kamel is a Full Professor and the chair of the Department of Electrical and Computer Engineering at University of Sharjah, UAE. He also holds an adjunct professor position at Concordia University, Canada. Before that, Prof. Kamel was a Lead Scientist at Panasonic Research Laboratory in Princeton, NJ, USA where he was managing several research projects, information security, multimedia information retrieval, and smart appliances. Prof. Kamel taught also at Polytechnic University in Brooklyn, Rutgers University in New Jersey, USA.

Prof. Kamel has 23 US patents in information storage and retrieval. He published more than 130 papers in international journals and conferences. He co-chaired several international conferences and workshops in information retrieval and databases. Prof. Kamel is serving as an associate editor for several international journals. He also participated in many technical program committees. He served in several National Science Foundation (NSF) review and panel committees. Prof. Kamel is a Senior Member of the IEEE. He served in the advisory committee for the Department of Electrical and Computer Engineering, University of Miami, USA and the Center for Advanced Information Processing (CAIP), Rutgers University, USA.