Received: 14 June 2023 | *Revised:* 7 October 2023 | *Accepted:* 20 October 2023
DOI: 10.37917/ijeee.20.2.4 *Early View* | *December 2024*

Iraqi Journal for Electrical and Electronic Engineering
Original Article

Open Access
IJEEE University of Basrah College of Engineering

An Adaptive Steganography Insertion Technique Based on Cosine Transform

Taif Alobaidi*[1], Wasfy Mikhael[2]
[1]Department of Mobile Communications and Computing Engineering, College of Engineering, University of Information Technology and Communications (UOITC), Baghdad, Iraq
[2]Department of Electrical and Computer Engineering, University of Central Florida, Orlando, USA

Correspondence
*Taif Alobaidi
Department of Mobile Communications and Computing Engineering, College of Engineering,
University of Information Technology and Communications (UOITC), Baghdad, Iraq
Email: taif.alobaidi@uoitc.edu.iq

Abstract
In the last couple decades, several successful steganography approaches have been proposed. Least Significant Bit (LSB) Insertion technique has been deployed due to its simplicity in implementation and reasonable payload capacity. The most important design parameter in LSB techniques is the embedding location selection criterion. In this work, LSB insertion technique is proposed which is based on selecting the embedding locations depending on the weights of coefficients in Cosine domain (2D DCT). The cover image is transformed to the Cosine domain (by 2D DCT) and predefined number of coefficients are selected to embed the secret message (which is in the binary form). Those weights are the outputs of an adaptive algorithm that analyses the cover image in two domains (Haar and Cosine). Coefficients, in the Cosine transform domain, with small weights are selected. The proposed approach is tested with samples from the BOSSbase, and a custom-built databases. Two metrics are utilized to show the effectiveness of the technique, namely, Root Mean Squared Error (RMSE), and Peak Signal-to-Noise Ratio (PSNR). In addition, human visual inspection of the result image is also considered. As shown in the results, the proposed approach performs better, in terms of (RMSE, and PSNR) than commonly employed truncation and energy based methods.

Keywords
Steganography, Digital Signal Processing, Cosine Transform, DWT

I. Introduction

Steganography finds its etymological roots in the Greek language. It is derived from the combination of two Greek words: ”steganos,” which signifies ”covered” or ”protected,” and ”graphia,” which means ”writing” or ”drawing.” These two terms come together to form ”steganographia,” a term that can be interpreted as ”covered writing” or ”hidden writing.” As time progressed, the term underwent a transformation, eventually becoming ”steganography” as we now understand it. Today, steganography refers to the technique of hiding information within different forms of media or carriers. Steganography is the art and science of hiding information within seemingly innocuous carriers, such as images, audio files, or text, without arousing suspicion. It has been used throughout history as a means to covertly transmit sensitive information or maintain clandestine communication channels. Steganography’s roots can be traced back to ancient times when secret messages were concealed within wax tablets, tattooed on messengers’ shaved heads, or written using invisible ink. However, one of the earliest recorded instances of steganography dates back to Herodotus, the Greek historian, who described a method where messages were written on a slave’s shaved head, allowing the hair to regrow before the messenger reached the intended recipient.

Digital Steganography techniques include Image [1], Audio [2], Text [3], and video [4]. In image steganography, involves embedding information within the pixels of digi-

tal images. Common methods include Least Significant Bit (LSB) insertion [5], where data bits (the information to be hidden) are stored in the least significant bit(s) of the image pixels (called the cover image that is the image in which you want to hide the secret information), and the use of spread spectrum techniques to distribute the hidden information across the image to obtain the encoded image, or stegoimage. In Audio Steganography, hidden information is concealed within audio files. Techniques like phase coding, echo hiding, and audio masking exploit the characteristics of sound to embed secret messages. In Text Steganography technique, information is hidden within textual content. Methods include employing invisible ink, modifying font styles, or utilizing hidden spaces or punctuation marks to encode data.

On the other hand and as steganography techniques continue to evolve, so do steganalysis [6] methods used to detect and analyze hidden messages. Steganalysis involves the application of statistical analysis [7], machine learning algorithms, and forensic techniques to identify the presence of steganographic content. Some common steganalysis methods include Statistical Analysis in which Steganalysis algorithms analyze statistical properties of carrier files to detect deviations from the expected patterns. These include examining pixel intensity distributions, correlation between neighboring pixels, and frequency domain analysis.

In addition to the first technique, Machine Learning-Based Steganalysis in which algorithms are trained to distinguish between normal and steganographic content by learning patterns from a large database of known carriers. Support Vector Machines (SVMs) [8], Artificial Neural Networks (ANNs) [9], and Random Forests [10] are commonly used in machine learning-based steganalysis. Finally, Visual Inspection where, in some cases, visual inspection by trained experts is employed to identify visual anomalies or irregularities in the carrier files that may indicate the presence of hidden information.

MSE (Mean Squared Error), RMSE (Root Mean Squared Error), SSIM (Structural Similarity Index), PSNR (Peak Signal-to-Noise Ratio) are commonly used metrics in image and video processing to evaluate the quality or fidelity of a reconstructed or compressed signal compared to the original signal [11].

In [12], introduced a novel steganography method employing LSB. The paper also provided information about recent relevant approaches. The process involved flipping and transforming the image and then dividing it into its 3 color channels: red, green, and blue. The blue channel is rearranged using the Magic Matrix, a built-in MATLAB function, to hide a secret message. To enhance security, Multi-Level Encryption (MLEA) algorithm is utilized. The proposed technique is evaluated using 12 color images, 9 grayscale images, 9 texture images, and 9 aerial images. These images were tested with different dimensions, in particular $2^d \times 2^d$ pixels where $d = [7, 8, 9,$ and $10]$. The secret message size varied between 2KB and 16KB. Evaluation metrics included PSNR, MSE, Structural Similarity Index (SSIM), and Normalized Cross-Correlation (NCC). The results demonstrated that this approach outperforms existing techniques. Further details regarding the results can be found in Section IV. in this work.

In [13], an examination of existing approaches, analyze current trends, and address the obstacles encountered in related research. It also included an examination of publicly accessible databases commonly employed in these studies and the evaluation measures utilized. Furthermore, the paper presented a comparative analysis of the performance of different methods and engages in a discussion regarding the identified gaps, advantages, and disadvantages of the approaches utilized in the present research. In [14], an approach is introduced that utilizes k least significant bits (LSB) coding to hide an image. This k-LSB-based method employs a specific number of least significant bits to conceal the image. To decode the hidden image, a region detection operation is performed to identify the blocks that contain the concealed image.

The resolution of the resulting stego image may be impacted, so an image quality enhancement technique is employed to improve the resolution. In order to showcase the effectiveness of this proposed approach, a comparison is made against several state-of-the-art methods. In [15], a robust and secure video steganographic algorithm was proposed in the discrete wavelet transform (DWT) and discrete cosine transform (DCT) domains, based on the multiple object tracking (MOT) algorithm and error correcting codes. The secret message was preprocessed by applying both Hamming and Bose, Chaudhuri, and Hocquenghem codes to encode the secret data. Initially, the motion-based MOT algorithm was implemented on host videos to differentiate the regions of interest in the moving objects. Subsequently, the data hiding process was performed by hiding the secret message within the DWT and DCT coefficients of all motion regions in the video, based on foreground masks. The experimental results demonstrated that the suggested algorithm not only improved the embedding capacity and imperceptibility, but also enhanced its security and robustness by encoding the secret message and withstanding various attacks.

In [16], a work presented a novel technique for image steganography based on Huffman Encoding. Two 8-bit gray level images of size M X N and P X Q were used as the cover image and secret image respectively. Huffman Encoding was performed over the secret image/message before embedding, and each bit of the Huffman code of the secret image/message was embedded inside the cover image by altering the least significant bit (LSB) of each pixel's intensity in the cover image.

The size of the Huffman encoded bit stream and Huffman Table were also embedded inside the cover image, making the Stego-Image a standalone information for the receiver. The experimental results showed that the algorithm had a high capacity and good invisibility. Furthermore, the Peak Signal to Noise Ratio (PSNR) of the stego image with the cover image yielded better results compared to other existing steganography approaches. Additionally, satisfactory security was maintained since the secret message/image could not be extracted without knowing the decoding rules and Huffman table.

In this work, a new image Steganogrphy insertion approach is presented. The approach starts with dividing the cover image into predefined number of non-overlapping blocks. Then, cover image blocks are transformed to the Cosine domain by utilizing the 2D DCT. The adaptive algorithm explained in [17] is employed to find the weights of each coefficient, for each block individually, in the Cosine domain. Blocks with coefficients that have low, compared to the rest of the coefficients, total weights are chosen. The coefficients in that block is converted to binary representation, refereed to as "cover in binary". The secret data, or more precisely its binary form, is embedded in the LSB bit of "cover in binary". The block is converted back to the decimal representation. Then, 2D IDCT is applied to obtain the stego image.

The metrics utilized to evaluate the performance of the proposed technique are RMSE, and PSNR. In addition, human visual inspection is also considered. The proposed system is compared with three other techniques. The first one is the traditional Spatial LSB, the second is energy based DCT insertion (in which total block energy is used as the selection parameter), and the third is the most recent reported work in [12]. The effect of size of cover image blocks is also examined. As shown in the results, the proposed approach performs better than the other techniques when tested with 10 samples from BossBase [18], and a custom-built databases.

The rest of the paper is organized as follows. Details about Stegnaogrpahy techniques are presented in Section II. In Section III. , the proposed technique is explained. The results are presented in Section IV. Section V. contains the discussion. The conclusions are shown in Section VI.

II. Steganography Techniques

In this section, details about Steganography insertion techniques [19] are presented. These techniques are in Spatial domain and in Discrete Cosine Transform (DCT) domain.

A. Spatial Domain LSB Insertion

The LSB (Least Significant Bit) insertion technique is a commonly used method in steganography for hiding information within the time domain of a digital signal, such as an image or audio file. This technique takes advantage of the fact that changing the least significant bit of a pixel or a sample in an audio signal has minimal impact on the overall perception of the signal. In LSB insertion, the binary representation of the secret message is embedded by replacing the least significant bit of selected pixels or audio samples with the corresponding bits from the message. The LSBs are typically modified because they have the least impact on the visual or auditory quality of the carrier signal. For example, in image steganography using LSB insertion, the pixels of an image are represented by three color channels: red, green, and blue (RGB). Each color channel consists of 8 bits per pixel, ranging from 0 to 255. The LSB of each color channel can be modified to store a single bit of the secret message, effectively hiding the information. The process involves the following steps:

1. Convert the secret message into binary representation;
2. Iterate through the pixels of the image or audio samples;
3. Modify the LSB of each selected pixel or sample according to the corresponding bit of the secret message;
4. Repeat the process until all bits of the secret message are embedded.

By modifying only the LSB, the changes introduced to the carrier signal are generally imperceptible to the human eye or ear. However, it is essential to consider the capacity of the carrier signal and ensure that the secret message can be embedded without causing significant distortion or noticeable artifacts. In conclusion, LSB insertion in the time domain provides a simple and straightforward method for steganographic data hiding, but it may be susceptible to detection by steganalysis techniques that analyze statistical properties or deviations from expected patterns in the carrier signal. Therefore, additional techniques such as encryption and more advanced steganographic methods may be employed to enhance the security and robustness of the hidden information.

B. Cosine Domain Insertion

Discrete Cosine Transform (DCT) is a mathematical technique commonly used in signal processing and data compression. It is also utilized in certain forms of steganography to hide information within digital media such as images or videos. The equations, forward and inverse, for calculating such coef-

ficients are [20]:

$$A(m,n) = \frac{2}{\sqrt{M \times N}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) c_m \cos\left(\frac{m(2u+1)\pi}{2M}\right) c_n \cos\left(\frac{n(2v+1)\pi}{2N}\right), \quad (1)$$

where $g(u,v)$ is the signal in the time domain and $G(m,n)$ is the m^{th} row, n^{th} column DCT coefficient for $u = 0,1,\ldots M-1$ and $v = 0,1,\ldots N-1$.

$$g(u,v) = \frac{2}{\sqrt{M \times N}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} A(m,n) c_m \cos\left(\frac{m(2u+1)\pi}{2M}\right) c_n \cos\left(\frac{n(2v+1)\pi}{2N}\right) \quad (2)$$

where c_m, and c_n are:

$$c_m = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } m = 0 \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

In steganography, the DCT is applied to blocks or segments of the cover image. Steganography algorithms that utilize DCT often choose certain frequency coefficients for hiding information. These coefficients are typically selected based on their perceptual importance, meaning coefficients that are less noticeable to the human eye are preferred. The most commonly used coefficients for steganography are usually those corresponding to low-frequency components. The secret data is usually in the form of binary bits. These bits are then embedded by modifying the selected DCT coefficients. The modification can be achieved by adding or subtracting small values to the coefficients, thereby encoding the secret information. After embedding the secret data, the modified DCT coefficients are quantized and compressed. Quantization reduces the precision of the coefficients, making the changes introduced by embedding less noticeable. Compression further helps in reducing the size of the stego image. At the receiver end and in order to retrieve the hidden information, the stego image undergoes the reverse process. The DCT coefficients are inversely transformed to the spatial domain, resulting in the reconstructed image. The hidden data is extracted by examining the modified coefficients. Finally, it's worth noting that the specific techniques and algorithms used in steganography can vary, and there are numerous variations and refinements to the process described earlier.

C. Discrete Haar Transform (DHT)

The process of Wavelets [21] results in 4 frequency bands: LL (Low Pass-Low Pass) $\varphi(i,j)$, LH (Low Pass-High Pass) $\psi^H(i,j)$, HL (High Pass-Low Pass) $\psi^V(i,j)$, and HH (High Pass-High Pass) $\psi^D(i,j)$, all combined within a matrix. When applied to 2D signals like images, a single-level DWT decomposition involves the utilization of a scaling function called $\varphi(i,j)$ and 3 wavelets referred to as $\psi(i,j)$. The computation of these wavelets is performed as follows:

$$\varphi(i,j) = \varphi(i)\varphi(j) \quad (4)$$

$$\psi^H(i,j) = \psi(i)\varphi(j) \quad (5)$$

$$\psi^V(i,j) = \varphi(i)\psi(j) \quad (6)$$

$$\psi^D(i,j) = \psi(i)\psi(j) \quad (7)$$

The 2D-DWT of an image *g(i,j)* of size $M \times M$ is:

$$W_\varphi(t_0,m,m) = \frac{1}{\sqrt{MM}} \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} g(i,j) \varphi_{t_0,m,m}(i,j) \quad (8)$$

$$W_\psi^r(t,m,m) = \frac{1}{\sqrt{MM}} \sum_{i=0}^{M-1} \sum_{t=0}^{M-1} g(i,j) \psi_{t,m,m}^r(i,j) \quad (9)$$
$$r = \{H,V,D\}$$

t_0 is an arbitrary initial scale and the $W_\varphi(t_0,m,m)$ coefficients is the Approximation of the *g(i,j)* at scale t_0. The $W_\psi^r(t,m,m)$ coefficients add horizontal, vertical, and diagonal details for scales $t \geq t_0$. Practically, $t_0 = 0$, $M = 2^J$so that $t = 0,1,2,....T-1$ and $m = 0,1,2,...,2^t - 1$.

D. Adaptive Algorithm

The steps in the adaptive algorithm detailed in [17] can be summarized as follows:

1. the total cover image energy is calculated;
2. The 2D DCT is applied to get the first 2D DCT representation and the energy in this domain is calculated;
3. Predefined number of coefficients are chosen and the rest are transformed back to the spatial domain;
4. The current version of the image is transformed to the Haar domain using 2D DHT. Then, predefined number of coefficients are chosen and the rest are transformed back to the spatial domain;
5. The current total energy is calculated. If the calculated value is less than 0.05% of the value in step 1, go to step 2 ;otherwise the algorithm halts;
6. The final outputs are the weights of each coefficient in each domains (Cosine, and Haar).

The energy residual, $\Phi(\alpha,\beta)$, the distinction lies in minimizing the cost function, which is calculated as the discrepancy between the initial energy and the energies preserved within each domain. In particular, $\Phi(\alpha,\beta)$ is calculated as follows:

$$\Phi(\alpha,\beta) = [C_1]^2 - [T_{2,1}(C_2)]^2 - [T_{3,1}(C_3)]^2 \tag{10}$$

where $[\,]^2$ is the element-wise square. The process begins by utilizing a Steepest Descent Algorithm [22] to decrease the remaining error. Once the iteration concludes, a designated count of coefficients is preserved in two distinct domains: the Cosine and the Haar domains. The resulting feature vector for each signal (here, it is the cover image) is obtained by combining these retained coefficients together.

The parameters for the Training phase are as follows. The weight matrices α is populated with 0.5 while β is initialized with 0.3. The updating equations in every iteration are as follows [23]:

$$\alpha_{x,y}(n+1) = \alpha_{x,y}(n) - \mu_{\alpha_{x,y}} \nabla_{\alpha_{x,y}} \Phi \tag{11}$$

$$\beta x,y(n+1) = \beta x,y(n) - \mu_{\beta x,y} \nabla_{\beta x,y} \Phi \tag{12}$$

where x, and y span the entire domain and depending on $\alpha_{x,y}$ and $\beta_{x,y}$ are elements in $[\alpha]$ and $[\beta]$ respectively, n is the iteration index, and μ is the converging factor. The converging factors, $\mu_{\alpha_{x,y}}$ and $\mu_{\beta x,y}$, are calculated in the following fashion:

$$\mu_\alpha = \frac{\Phi(n)}{\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}[\nabla_{\alpha_{x,y}}\Phi]^2} \tag{13}$$

$$\mu_\beta = \frac{\Phi(n)}{\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}[\nabla_{\beta_{x,y}}\Phi]^2} \tag{14}$$

E. Performance Metrics

Performance metrics [24] are measurements used to evaluate the effectiveness, efficiency, accuracy, or quality of a system, process, algorithm, or model. The choice of performance metrics depends on the specific task or application.

1) Structural Similarity Index (SSIM)

To measure similarity between parts of same or different images, SSIM is employed. Post-processing quantitative judgment of the change of the structure of these parts is measured by SSIM. Structure, contrast, and luminance are the segment ofSSIM. SSIM $\in [-1,1]$, and the maximm limit is reached when image parts are identical. The calculation of SSIM is:

$$SSIM(a,b) = [s(a,b) * c(a,b) * l(a,b)] \tag{15}$$

where a and b are input images (or blocks) under comparison, s(a, b) is structure component, c(a, b) equals contrast, and l(a, b) is luminance. These factors are calculated in the following manner:

$$\begin{aligned} s(a,b) &= \frac{\sigma_{ab} + C3)}{\sigma_a \sigma_b + C3} \\ c(a,b) &= \frac{2\sigma_a\sigma_b + C2}{\sigma_a^2 + \sigma_b^2 + C2} \\ C2 &= (K2 * L)^2 \\ l(a,b) &= \frac{2\mu_a\mu_b + C1}{\mu_a^2 + \mu_b^2 + C1} \\ C1 &= (K1 * L)^2 \end{aligned} \tag{16}$$

where σ depicts standard deviation, μ represents mean, σ^2 represents variance, K1 = 0.01, L equals to one, C3 is a small constant, and K2 = 0.03.

2) Root Mean Squared Error (RMSE)

Mean Squared Error, one of the Regression metrics, measures the average squared difference between the pixel values of the original signal and the reconstructed/compressed signal. It provides a quantitative measure of the overall distortion between the two signals. The Root MSE (RMSE), the square root of the MSE, is calculated as follows:

$$RMSE = \sqrt{\frac{1}{m \times n}\sum_m \sum_n (I(x,y) - K(x,y))^2} \tag{17}$$

where I(x, y) represents the pixel value of the original signal at position (x, y), K(x, y) represents the pixel value of the reconstructed/compressed signal at the same position, and (m * n) is the total number of pixels in the image. A lower RMSE value indicates a smaller average difference and, therefore, better reconstruction or compression quality. However, RMSE alone may not provide a perceptually meaningful measure of quality, as it does not consider the human visual system's sensitivity to different image characteristics.

3) Peak Signal-to-Noise Ratio (PSNR)

PSNR is a logarithmic measure that relates the maximum possible power of a signal (in this case, the maximum possible pixel value) to the power of the noise (the difference between the original and reconstructed/compressed signals). It is expressed in decibels (dB). The formula for PSNR is:

$$PSNR = 10 * \log_{10}(MAX^2/MSE) \tag{18}$$

where MAX is the maximum pixel value (e.g., 255 for an 8-bit grayscale image). PSNR provides a more perceptually relevant measure of quality because it takes into account the dynamic range of the pixel values and is logarithmic. A higher PSNR value indicates better quality, as it indicates a smaller ratio of noise to the maximum signal power.

4) Comments on The Performance Metrics

It is important to note that RMSE, SSIM, and PSNR have limitations. They do not capture all aspects of image quality, such as human visual perception, and may not always correlate well with subjective evaluations. Therefore, it's recommended to use these metrics in combination with other quality assessment methods and consider the specific requirements and characteristics of the application or task at hand.

III. Proposed Technique

Fig. 1 shows the proposed technique. The method initiates by dividing the original image into distinct blocks that do not overlap. To determine the weights of each coefficient within the Cosine domain, an adaptive algorithm described in D.is applied individually to each block. The block with coefficients that exhibits lower total weights compared to the remaining blocks is selected. These chosen coefficients are converted into a binary representation called "cover in binary." The secret data, specifically its binary form, is then embedded in the least significant bit (LSB) of the cover in binary. Afterward, the block is converted back to its decimal representation. To obtain the stegoimage, 2D IDCT is applied. At the receiver end, the recipient can extract the secret data through partitioning the image into blocks. The block dimensions have to be exactly as the ones utilized in encoding process. Also, the index(es) of the chosen block(s) has(have) to be securely sent to the receiver.

The encoding part of the algorithm of the proposed technique/Encoding Process is as follows:

1. Inputs:Cover Image, Secret Message
2. Divide the cover image into non-overlapping blocks
3. Get weights of Cosine and DHT coefficients for each block by Applying D.
4. Sum the weights of Cosine coefficients in each block
5. Convert chosen block(s) to Binary
6. Convert Message to Binary
7. Insert bits of the message in LSB of the chosen blocks
8. Convert blocks to Decimal, and transform to Spatial domain via 2
9. Output: Stegoimage

The performance of the proposed technique is evaluated using the following metrics:

- Root Mean Squared Error (RMSE)
- Structural Similarity Index (SSIM)
- Peak Signal-to-Noise Ratio (PSNR)
- Human Visual Observation

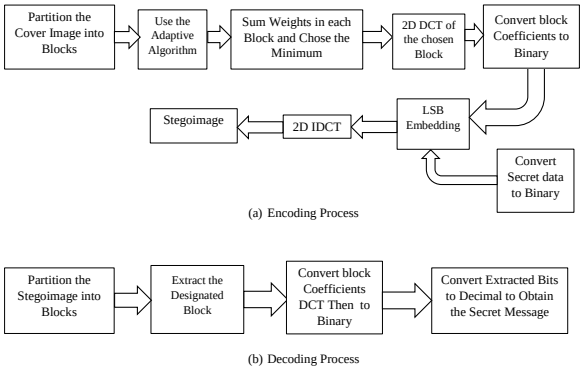

Fig. 1. The proposed technique utilized in steganography system. the two modules of the system are shown.

TABLE I.
PSNRs for Spatial/ DCT Blocks/Proposed Techniques for BOSSBase Database(Proposed Blocks Size is 4×4)

Image Index	Spatial	DCT Blocks	Proposed
1	85.5	50.16	86.3
2	86.75	50.22	87.26
3	86.3	50.25	87.26
4	86.3	50.19	88.51
5	87.84	50.23	86.3
6	90.28	50.16	87.26
7	88.51	50.22	87.84
8	86.75	50.22	87.26
9	87.84	50.17	86.3
10	87.84	50.28	86.75

The proposed system is compared against three other techniques: the traditional Spatial LSB, energy-based DCT insertion (which employs total block energy as the selection parameter), and LSB insertion technique in [12].

IV. Results

The evaluation of the proposed technique is implemented as shown in the following categories:

1. BossBase 10 samples database (Break Our Steganographic System Base) [18].
   The database consists of collection containing $10,000$

TABLE II.
PSNRS FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 8×8)

Image Index	Spatial	DCT Blocks	Proposed
1	81.11	43.13	82.15
2	81.98	43.11	81.38
3	81.98	43.11	82.15
4	80.61	43.12	81.24
5	80.61	43.09	81.24
6	80.61	43.06	82.15
7	81.11	43.05	81.11
8	81.67	43.12	80.98
9	82.32	43.16	81.52
10	80.61	43.12	81.52

TABLE III.
PSNRS FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 16×16)

Image Index	Spatial	DCT Blocks	Proposed
1	75.4	37.39	75.73
2	75.76	37.38	75.65
3	75.4	37.37	74.96
4	75.54	37.37	75.06
5	75.43	37.38	75.19
6	75.76	37.4	75.47
7	75.19	37.37	74.9
8	74.83	37.38	75.26
9	75.36	37.39	75.02
10	74.8	37.41	74.99

TABLE IV.
PSNRS FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 32×32)

Image Index	Spatial	DCT Blocks	Proposed
1	69.29	30.92	69.31
2	69.15	30.92	69.65
3	69.14	30.91	69.35
4	69.19	30.92	69.47
5	69.46	30.92	69.25
6	69.17	30.92	69.19
7	69.39	30.92	69.09
8	69.06	30.91	69.04
9	69.22	30.92	69.15
10	69.17	30.93	69.23

TABLE V.
PSNRS FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABAS(PROPOSED BLOCKS SIZE IS 64×64)

Image Index	Spatial	DCT Blocks	Proposed
1	63.16	24.94	63.27
2	63.23	24.94	63.97
3	63.2	24.94	63.34
4	63.06	24.94	63.18
5	63.33	24.94	63.42
6	63.15	24.92	63.17
7	63.31	24.94	63.28
8	63.2	24.93	63.22
9	63.15	24.94	63.19
10	63.21	24.94	63.32

TABLE VI.
PSNRS FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 128×128)

Image Index	Spatial	DCT Blocks	Proposed
1	57.18	18.92	57.29
2	57.15	18.94	58.09
3	57.15	18.94	57.22
4	57.2	18.94	56.76
5	57.22	18.93	57.76
6	57.16	18.91	57.1
7	57.19	18.92	56.9
8	57.17	18.92	57.15
9	57.17	18.93	57.46
10	57.18	18.94	57.42

TABLE VII.
RMSES FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 4×4)

Image Index	Spatial	DCT Blocks	Proposed
1	0.000183	0.626158	0.000153
2	0.000137	0.618507	0.000122
3	0.000153	0.613738	0.000122
4	0.000153	0.621732	0.000092
5	0.000107	0.616230	0.000153
6	0.000061	0.627000	0.000122
7	0.000092	0.618000	0.000107
8	0.000137	0.618704	0.000122
9	0.000107	0.624807	0.000153
10	0.000107	0.609347	0.000137

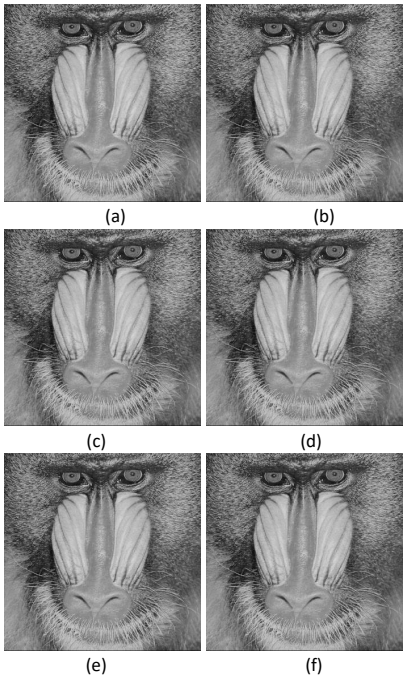

Fig. 2. Sample 1 image from custom-built database/proposed LSB technique. (a) Original cover image, (b) Stegoimage with message size of 6KB, (c) Stegoimage with message size of 8KB, (d) Stegoimage with message size of 10KB, (e) Stegoimage with message size of 14KB, (f) Stegoimage with message size of 16KB.

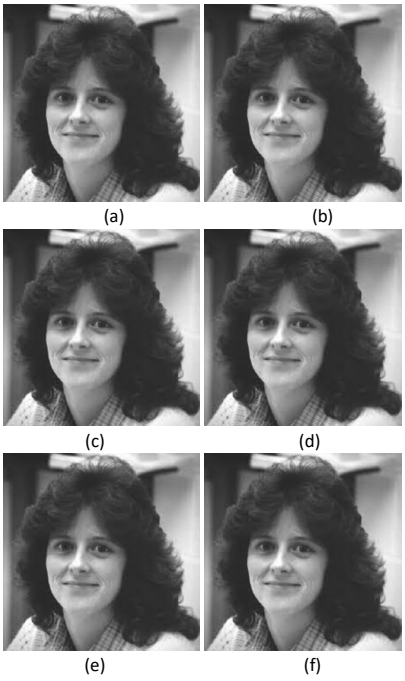

Fig. 3. Sample 2 image from custom-built database/proposed LSB technique. (a) Original cover image, (b) Stegoimage with message size of 6KB, (c) Stegoimage with message size of 8KB, (d) Stegoimage with message size of 10KB, (e) Stegoimage with message size of 14KB, (f) Stegoimage with message size of 16KB.

black and white images for experiments with detecting steganographically hidden data in JPEG images. It contains discrete cosine transform residuals (DCTR), Gabor filter residuals (GFR) and PHase Aware pRojection Model (PHARM) features extracted out of clean images, images with random data hidden using the JPEG Universal Wavelet Relative Distortion, images with random data hidden using the *nsF*5 method, images with random data hidden using the Uniform Embedding Revisited Distortion (UERD) algorithm.
The results (in terms of PSNRs)for this category are shown in Tables I through VI. On the other hand, Tables VII through XII are populated with RMSE. The SSIM for most of the results were close to unity.

2. Custom-built image database appeared in [12]
Four grey-scale image, shown in Figs. 2 through 5, form the second tested database. The results (in terms of PSNRs)for this category are shown in Table XIII. On the other hand, Table XIV is populated with RMSE. The SSIM for most of the results were close to unity.

To examine the effect of the block size, or dimensions, on the performance of the proposed technique, different block sizes are considered. As shown in Tables I through XII, the block size of 4 means 4×4 which is 16 pixels, or coefficients. First, 1 sample from the database is shown in Fig. 6 that shows the original image besides the stegoimages for different block sizes for the LSB Spatial case. Secondly, DCT block insertion case outputs are shown in Fig. 7. Finally, the proposed technique outputs are shown in Fig. 8.

V. Discussion

A. BOSSBase Results

As shown in the presented results, the proposed technique performed better than the other two techniques under comparison. The RMSE maintained at lower levels while higher PSNRs are achieved. The visual human inspection illustrate that the proposed technique does not alter the visual properties of the cover image. In terms of utilized performance metrics, the best block size (window dimensions) is 4×4 (16 coefficients).

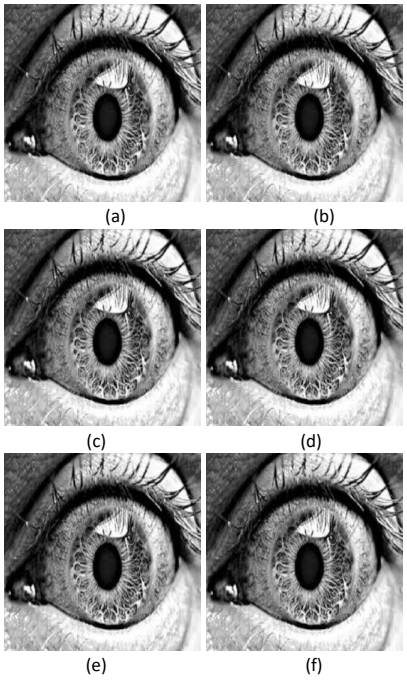

Fig. 4. Sample 3 image from custom-built database/proposed LSB technique. (a) Original cover image, (b) Stegoimage with message size of 6KB, (c) Stegoimage with message size of 8KB, (d) Stegoimage with message size of 10KB, (e) Stegoimage with message size of 14KB, (f) Stegoimage with message size of 16KB.

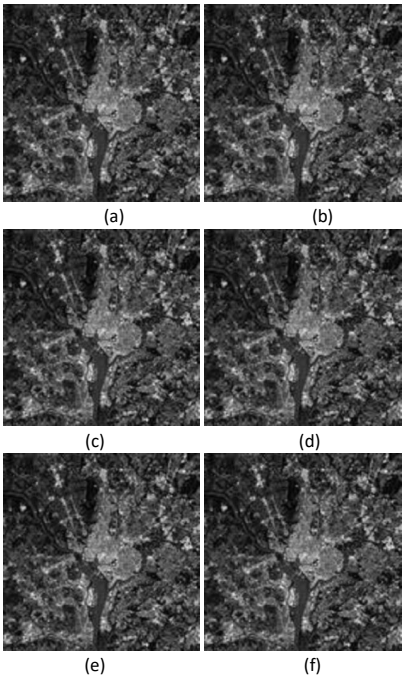

Fig. 5. Sample 4 image from custom-built database/proposed LSB technique. (a) Original cover image, (b) Stegoimage with message size of 6KB, (c) Stegoimage with message size of 8KB, (d) Stegoimage with message size of 10KB, (e) Stegoimage with message size of 14KB, (f) Stegoimage with message size of 16KB.

Nevertheless, the smaller window dimensions acquire more processing resources.

B. Custom-Built Results

As shown in the presented results, the proposed technique performed better than the other technique in [12]. The RMSE maintained at lower levels while higher PSNRs are achieved. The visual human inspection illustrate that the proposed technique does not alter the visual properties of the cover image. In terms of utilized performance metrics, the best block size (window dimensions) is 4×4 (16 coefficients). Nevertheless, the smaller window dimensions acquire more processing resources.

C. Steganalysis Results

As shown in Fig. 9, the histogram of the proposed technique has not been altered and thus the proposed technique is immune against first order attacks like Chi-Square [25].

VI. Conclusions

A technique is proposed for inserting a secret message into an image, which is based on the two-dimensional Cosine Transform (2D DCT). In this method, the image was converted to the Cosine domain using 2D DCT, and a predetermined number of coefficients are chosen to hide the binary secret message. The selection process involves analyzing the image in two different domains: 2D DCT and 2D Haar Transform. This analysis was performed to minimize any distortions in the original cover image. The adaptive algorithm yields weights for each coefficient in its domain, and Cosine coefficients with lower weights are selected for embedding the secret message. To evaluate the effectiveness of the technique, samples from the BOSSbase, and custom-built databases were used, and three metrics were employed: Root Mean Squared Error (RMSE), Sturcutral Similarity Index (SSIM), and Peak Signal-to-Noise Ratio (PSNR). Additionally, a visual inspection of the resulting image is also taken into account. The results demonstrated that the proposed technique outperformed commonly used truncation, energy-based methods, and most recently reported

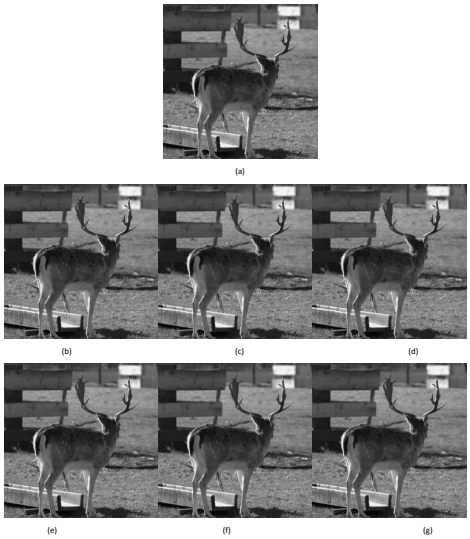

Fig. 6. Samples from BOSSBase database/spatial LSB. (a) Original cover image, (b) Stegoimage with block size of 4, (c) Stegoimage with block size of 8, (d) Stegoimage with block size of 16, (e) Stegoimage with block size of 32, (f) Stegoimage with block size of 64, (g) Stegoimage block size of 128.

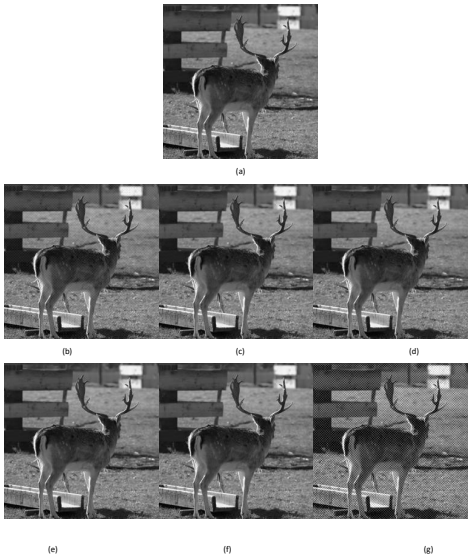

Fig. 7. Samples from BOSSBase database/DCT Blocks. (a) Original cover image, (b) Stegoimage with block size of 4, (c) Stegoimage with block size of 8, (d) Stegoimage with block size of 16, (e) Stegoimage with block size of 32, (f) Stegoimage with block size of 64, (g) Stegoimage block size of 128.

technique in terms of RMSE, SSIM, PSNR, and visual quality. In addition, the histogram of the stegoimages rendered by the proposed technique was not altered which indicate the immunity against attacks.

Conflict of Interest

The authors have no conflict of relevant interest to this article.

References

[1] P. C. Mandal, I. Mukherjee, G. Paul, and B. Chatterji, "Digital image steganography: A literature survey," *Information sciences*, 2022.

[2] A. A. AlSabhany, A. H. Ali, F. Ridzuan, A. Azni, and M. R. Mokhtar, "Digital audio steganography: Systematic review, classification, and analysis of the current state of the art," *Computer Science Review*, vol. 38, p. 100316, 2020.

[3] M. A. Majeed, R. Sulaiman, Z. Shukur, and M. K. Hasan, "A review on text steganography techniques," *Mathematics*, vol. 9, no. 21, p. 2829, 2021.

[4] Y. Liu, S. Liu, Y. Wang, H. Zhao, and S. Liu, "Video steganography: A review," *Neurocomputing*, vol. 335, pp. 238–250, 2019.

[5] S. Dhawan and R. Gupta, "Analysis of various data security techniques of steganography: A survey," *Information Security Journal: A Global Perspective*, vol. 30, no. 2, pp. 63–87, 2021.

[6] J. Fridrich and M. Goljan, "Practical steganalysis of digital images: state of the art," *security and Watermarking of Multimedia Contents IV*, vol. 4675, pp. 1–13, 2002.

[7] R. Sonar and G. Swain, "Steganography based on quotient value differencing and pixel value correlation," *CAAI Transactions on Intelligence Technology*, vol. 6, no. 4, pp. 504–519, 2021.

[8] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Transactions on information forensics and security*, vol. 7, no. 2, pp. 432–444, 2011.

[9] J. Davidson, C. Bergman, and E. Bartlett, "An artificial neural network for wavelet steganalysis," in *Mathemati-*

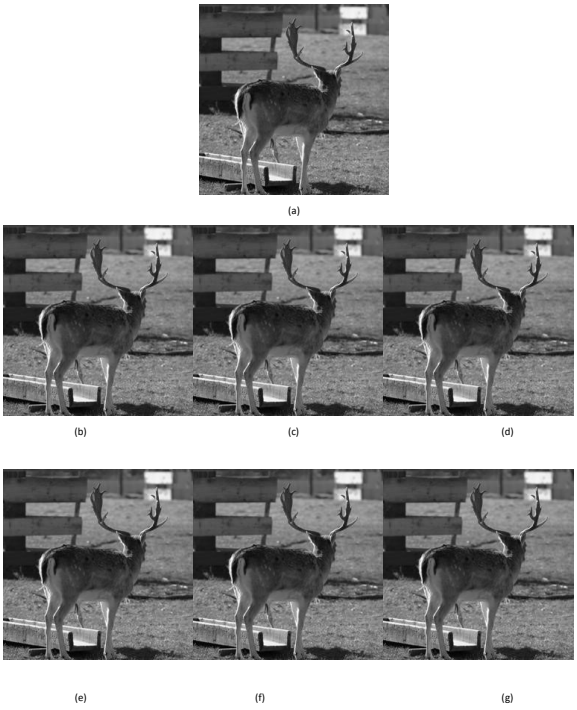

Fig. 8. Samples from BOSSBase database/proposed LSB technique. (a) Original cover image, (b) Stegoimage with block size of 4, (c) Stegoimage with block size of 8, (d) Stegoimage with block size of 16, (e) Stegoimage with block size of 32, (f) Stegoimage with block size of 64, (g) Stegoimage block size of 128.

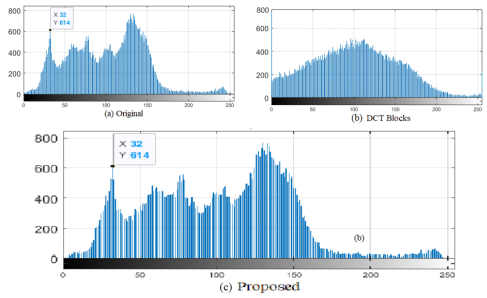

Fig. 9. Histogram of images. (a) Cover image histogram (image 1/custom-built database), (b) DCT blocks insertion stegoimage histogram, (c) Proposed technique stegoimage histogram.

TABLE VIII.
RMSEs FOR SPATIAL/ DCT BLOCKS/PROPOSED TECHNIQUES FOR BOSSBASE DATABASE(PROPOSED BLOCKS SIZE IS 8×8)

Image Index	Spatial	DCT Blocks	Proposed
1	0.022450	1.778500	0.019925
2	0.020298	1.781846	0.021749
3	0.020298	1.781899	0.019925
4	0.023770	1.780727	0.022091
5	0.023770	1.786850	0.022091
6	0.023770	1.792158	0.019925
7	0.022450	1.794135	0.022450
8	0.021048	1.780192	0.022782
9	0.019519	1.772875	0.021401
10	0.023770	1.781059	0.021401

cal Methods in Pattern and Image Analysis, vol. 5916, pp. 138–147, SPIE, 2005.

[10] A. Dehdar, A. Keshavarz, and N. Parhizgar, "Image steganalysis using modified graph clustering based ant colony optimization and random forest," *Multimedia Tools and Applications*, vol. 82, no. 5, pp. 7401–7418, 2023.

[11] A. K. Sahu and M. Sahu, "Digital image steganography and steganalysis: A journey of the past three decades," *Open Computer Science*, vol. 10, no. 1, pp. 296–342, 2020.

[12] S. Rahman, J. Uddin, H. U. Khan, H. Hussain, A. A. Khan, and M. Zakarya, "A novel steganography technique for digital images using the least significant bit substitution method," *IEEE Access*, vol. 10, pp. 124053–124075, 2022.

[13] N. Subramanian, O. Elharrouss, S. Al-Maadeed, and A. Bouridane, "Image steganography: A review of the recent advances," *IEEE access*, vol. 9, pp. 23409–23423, 2021.

[14] O. Elharrouss, N. Almaadeed, and S. Al-Maadeed, "An image steganography approach based on k-least significant bits (k-lsb)," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, pp. 131–135, IEEE, 2020.

[15] R. J. Mstafa, K. M. Elleithy, and E. Abdelfattah, "A robust and secure video steganography method in dwt-dct domains based on multiple object tracking and ecc," *IEEE access*, vol. 5, pp. 5354–5365, 2017.

[16] R. Das and T. Tuithung, "A novel steganography method for image based on huffman encoding," in *2012 3rd National Conference on Emerging Trends and Applications in Computer Science*, pp. 14–18, IEEE, 2012.

TABLE IX.
RMSEs for Spatial/ DCT Blocks/Proposed Techniques for BOSSBase Database(Proposed Blocks Size is 16×16)

Image Index	Spatial	DCT Blocks	Proposed
1	0.043324	3.443646	0.041713
2	0.041521	3.446317	0.042071
3	0.043324	3.450433	0.045552
4	0.042615	3.453333	0.045044
5	0.043151	3.447460	0.044362
6	0.041521	3.439689	0.042965
7	0.044362	3.451074	0.045891
8	0.046217	3.448123	0.044023
9	0.043497	3.445105	0.045222
10	0.046379	3.436676	0.045387

TABLE X.
RMSEs for Spatial/ DCT Blocks/Proposed Techniques for BOSSBase Database(Proposed Blocks Size is 32×32)

Image Index	Spatial	DCT Blocks	Proposed
1	0.087521	7.256624	0.087258
2	0.088904	7.251974	0.083964
3	0.088989	7.261676	0.086908
4	0.088476	7.253137	0.085761
5	0.085849	7.252547	0.087955
6	0.088736	7.254067	0.088561
7	0.086470	7.256357	0.089588
8	0.089844	7.259191	0.090011
9	0.088216	7.253918	0.088904
10	0.088736	7.244622	0.088131

TABLE XI.
RMSEs for Spatial/ DCT Blocks/Proposed Techniques for BOSSBase Database(Proposed Blocks Size is 64×44)

Image Index	Spatial	DCT Blocks	Proposed
1	0.177251	14.444816	0.174997
2	0.175738	14.437805	0.161437
3	0.176516	14.446083	0.173554
4	0.179262	14.443265	0.176777
5	0.173862	14.446958	0.171919
6	0.177423	14.465452	0.176992
7	0.174213	14.441371	0.174780
8	0.176474	14.447889	0.175955
9	0.177336	14.443338	0.176562
10	0.176216	14.442746	0.173948

TABLE XII.
RMSEs for Spatial/ DCT Blocks/Proposed Techniques for BOSSBase Database(Proposed Blocks Size is 128×128)

Image Index	Spatial	DCT Blocks	Proposed
1	0.352732	28.881922	0.348182
2	0.353963	28.819823	0.156946
3	0.353833	28.806846	0.167929
4	0.351909	28.815145	0.170294
5	0.351063	28.841189	0.166724
6	0.353791	28.893205	0.171172
7	0.352580	28.881184	0.169706
8	0.353381	28.880956	0.170605
9	0.353143	28.842339	0.170860
10	0.352624	28.815215	0.168526

[17] T. Alobaidi and W. B. Mikhael, "Mixed nonorthogonal transforms representation for face recognition," *Circuits, Systems, and Signal Processing*, vol. 38, pp. 1684–1694, 2019.

[18] M. Płachta, M. Krzemień, K. Szczypiorski, and A. Janicki, "Detection of image steganography using deep learning and ensemble classifiers," *Electronics*, vol. 11, no. 10, 2022.

[19] A. S. Ansari, M. S. Mohammadi, and M. T. Parvez, "A multiple-format steganography algorithm for color images," *IEEE Access*, vol. 8, pp. 83926–83939, 2020.

[20] W. Burger and M. J. Burge, *Digital image processing: an algorithmic introduction using Java*. Springer Science & Business Media, 2009.

[21] M. Stéphane, *A wavelet tour of signal processing*. Academic press, 1999.

[22] R. Araya, "Enriching elementary school mathematical learning with the steepest descent algorithm," *Mathematics*, vol. 9, no. 11, p. 1197, 2021.

[23] R. Arun and M. Wasfy B., "Multitransform/multidimensional signal representation," in *Circuits and Systems, 1993., Proceedings of the 36th Midwest Symposium on*, pp. 1255–1258, IEEE, 1993.

[24] U. Sara, M. Akter, and M. S. Uddin, "Image quality assessment through fsim, ssim, mse and psnr—a comparative study," *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8–18, 2019.

[25] I.-H. Pan, K.-C. Liu, and C.-L. Liu, "Chi-square detection for pvd steganography," in *2020 International*

TABLE XIII.
PSNRS FOR TWO TECHNIQUES/ CUSTOM-BUILT DATABASE(PROPOSED BLOCKS SIZE IS 4×4)

Image Index	Message Size (KB)	PSNR /Work in [12]	PSNR /Proposed
1	6	81.112	94.535
1	8	77.042	92.316
1	10	77.043	93.285
1	14	73.013	94.535
1	16	77.098	92.316
2	6	89.033	97.545
2	8	87.961	93.285
2	10	81.812	93.865
2	14	79.643	95.327
2	16	77.89	93.285
3	6	90.033	92.316
3	8	84.941	93.865
3	10	80.412	92.774
3	14	79.643	92.774
3	16	78.678	93.285
4	6	86.033	96.296
4	8	85.941	94.535
4	10	81.412	92.774
4	14	77.622	93.865
4	16	77.099	94.535

TABLE XIV.
RMSES FOR TWO TECHNIQUES/ CUSTOM-BUILT DATABASE(PROPOSED BLOCKS SIZE IS 4×4)

Image Index	Message Size (KB)	PSNR /Work in [12]	PSNR /Proposed
1	6	0.02244	0.00478
1	8	0.03585	0.00618
1	10	0.03584	0.00552
1	14	0.05700	0.00478
1	16	0.03562	0.00618
2	6	0.00901	0.00338
2	8	0.01020	0.00552
2	10	0.02070	0.00517
2	14	0.02657	0.00437
2	16	0.03251	0.00552
3	6	0.00803	0.00618
3	8	0.01444	0.00517
3	10	0.02432	0.00586
3	14	0.02657	0.00586
3	16	0.02969	0.00552
4	6	0.01273	0.00391
4	8	0.01287	0.00478
4	10	0.02167	0.00586
4	14	0.03353	0.00517
4	16	0.03561	

Symposium on Computer, Consumer and Control (IS3C), pp. 30–33, IEEE, 2020.