Alobaidi and Mikhael
Journal of Engineering and Applied Science (2023) 70:144
https://doi.org/10.1186/s44147-023-00300-x

Journal of Engineering and Applied Science

RESEARCH Open Access

An adaptive steganography insertion technique based on wavelet transform

Taif Alobaidi[1*] and Wasfy Mikhael[2]

*Correspondence:
taif.alobaidi@uoitc.edu.iq

[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 32816, FL, USA

Abstract

Over the past few decades, there have been several successful methods developed for steganography. One popular technique is the insertion method, which is favored for its simplicity and ability to hold a reasonable amount of hidden data. This study introduces an adaptive insertion technique based on the two-dimensional discrete Haar filter (2D DHF). The technique involves transforming the cover image into the wavelet domain using 2D DWT and selecting a predetermined number of coefficients to embed the binary secret message. The selection process is carried out by analyzing the cover image in two non-orthogonal domains: 2D discrete cosine transform and 2D DHF. An adaptive algorithm is employed to minimize the impact on the unrepresented parts of the cover image. The algorithm determines the weights of each coefficient in each domain, and coefficients with low weights are chosen for embedding. To evaluate the effectiveness of the proposed approach, samples from the BOSSbase and custom databases are used. The technique's performance is measured using three metrics: mean squared error (MSE), peak signal-to-noise ratio (PSNR), and structural similarity index (SSIM). Additionally, a visual inspection by humans is conducted to assess the resulting image. The results demonstrate that the proposed approach outperforms recently reported methods in terms of MSE, PSNR, SSIM, and visual quality.

Keywords: Steganography, Digital signal processing, DCT, DHF

Introduction

The concept of steganography has its origins in the Greek language, specifically from two Greek words: "steganos," meaning "covered" or "protected," and "graphia," meaning "writing" or "drawing." These words combined to form "steganographia," which can be interpreted as "covered writing" or "hidden writing." Over time, the term evolved into "steganography" as we know it today. Steganography refers to the technique of concealing information within various forms of media or carriers. It is considered an art and science of hiding information within seemingly innocent carriers like images, audio files, or text, without arousing suspicion. Throughout history, steganography has been used to covertly transmit sensitive information or maintain secret communication channels. Ancient methods included hiding messages in wax tablets, tattooing them on messengers' shaved heads, or using invisible ink. One of the earliest recorded instances of

steganography dates back to the Greek historian Herodotus, who described a method where messages were written on a slave's shaved head, allowing the hair to regrow before reaching the intended recipient [1]. Steganography techniques involve hiding information in different types of media. Image steganography involves embedding information within the pixels of digital images, using methods like least significant bit (LSB) insertion or spread spectrum techniques. Audio steganography hides information within audio files, utilizing characteristics of sound such as phase coding or audio masking. Text steganography conceals information within textual content, employing methods like invisible ink, modifying font styles, or utilizing hidden spaces or punctuation marks. As steganography techniques evolve, methods of steganalysis also develop to detect and analyze hidden messages. Steganalysis [2] involves statistical analysis, machine learning algorithms, and forensic techniques to identify the presence of steganographic content. Common steganalysis methods include statistical analysis of carrier files, machine learning-based approaches using algorithms like support vector machines (SVMs) [3] or artificial neural networks (ANNs) [4], and visual inspection by trained experts to identify anomalies [5]. Peak signal-to-noise ratio (PSNR) [6], structural similarity index (SSIM) [7], and mean squared error (MSE) [8] are metrics commonly used in image and video processing to assess the quality or fidelity of a reconstructed or compressed signal compared to the original signal. The work presented in [9] proposed a steganography technique using pixel swapping-quantum Hilbert image scrambling and discrete wavelet transformation (DWT), followed by stego image smoothing operation. Initial steps of proposed image steganography include best suitable cover image selection for secret image. This process compares the secret image with the different cover image presents in senders cover image database. Before embedding (encoding), the secret image is scrambled, and then performs the DWT transformation on this scrambled image and cover image. Next steps embed both resultant images and generate a new image named as stego image. The quality of stegoimage is increased using pixel swapping based operation. The decoding process of proposed steganography scheme is just reverse of the former encoding process followed by application of convolutional neural networks (CNN) to improve the extracted secret image quality. Two metrics were employed to show the effectiveness of the proposed technique, namely, PSNR, and normalized cross-correlation (NCC). Experimental results included three publicly available photos, and the results showed improvement (around 2dB) over the existing approaches before and after the smoothing application. In [10], a novel least significant bit substitution (LSB) steganography approach is presented. Details about other recently-published relevant approaches were also given. The image is flipped, transformed, and partitioned into the three-color channels. The red, green, and blue (this channel is shuffled using Magic Matrix, a MATLAB built-in function) channels are employed to embed the secret message. Multi level encryption algorithm (MLEA) is deployed to increase the level of robustness of the proposed approach. Twelve colored images, nine gray-scale, nine texture images, and nine aerial images were utilized to test the performance of the proposed technique. All images in the first set were tested with varying the dimensions (e.g., $128 \times 128, 256 \times 256, 512 \times 512$, and 1024×1024). The size of the secret message was one of the following values: 2, 4, 6, 8, 10, 12, 14, and 16 KB. The metrics employed to assess the proposed technique were PSNR, MSE, structural similarity index (SSIM), and

normalized cross-correlation (NCC). The results showed that the proposed approach was better than the rest of the existing techniques. More details about the results in that paper is given in the "Results" section . In one research paper [11], existing approaches, current trends, and obstacles in steganography research were examined. The paper analyzed publicly accessible databases and evaluation measures commonly used in these studies. It also conducted a comparative analysis of different methods and discussed their identified gaps, advantages, and disadvantages. Another paper [12] introduced an image steganography approach based on k least significant bits (LSB) coding. The proposed method concealed an image by utilizing a specific number of least significant bits. The paper compared this approach with other state-of-the-art methods. In another related work [13], a robust and secure video steganographic algorithm was proposed, utilizing discrete wavelet transform (DWT) and discrete cosine transform (DCT) domains, motion-based tracking, and error correcting codes. The paper demonstrated improved embedding capacity, imperceptibility, security, and robustness against various attacks. A novel technique for image steganography based on Huffman Encoding was presented in [14], showcasing high capacity, good invisibility, and satisfactory security.

In this paper, a new image steganography insertion approach is introduced. The approach divides the cover image into non-overlapping blocks and transforms them using 2D discrete wavelet transform (DWT). An adaptive algorithm [15–18] determines the weights of each coefficient in the Wavelet domain for each block. The block with coefficients having lower total weights compared to others is selected. The secret data is embedded in the least significant bit (LSB) of the binary representation of the selected block. The block is converted back to the decimal representation, and inverse DWT is applied to obtain the stego image. The performance of the proposed technique is evaluated using MSE, PSNR, and human visual inspection. A comparison is made with two other techniques: Spatial LSB and energy-based DCT insertion. The impact of the size of the cover image block is also examined. The experimental results demonstrate that the proposed approach outperforms the other techniques when tested with a samples from the BOSSBase [19], and a customized databases. The remaining sections of the paper are organized as follows: "Methods" presents details about steganography techniques, "Proposed technique" explains the proposed technique, "Results" presents the results, "Results discussion" contains the discussion, and "Conclusions" summarizes the conclusions.

Methods

This section provides an overview of different techniques used in steganography. These techniques can be categorized into two domains: spatial domain and discrete wavelet transform (DWT) domain.

Spatial domain LSB insertion

The spatial domain LSB insertion technique is a commonly used method in steganography. It involves hiding information within the least significant bit of a pixel or a sample in an image or audio signal. The least significant bit is chosen because altering it has minimal impact on the overall perception of the signal. In this technique, 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. This process is

repeated 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 humans. However, it is important to consider the capacity of the carrier signal to ensure that the secret message can be embedded without causing noticeable artifacts. While LSB insertion provides a simple method for steganographic data hiding, it may be vulnerable to detection by steganalysis techniques that analyze statistical properties or deviations from expected patterns in the carrier signal. To enhance security, additional techniques like encryption and more advanced steganographic methods can be employed.

Discrete wavelet transform (DWT)

The wavelet process, as described in [20, 21], produces four frequency bands: LL (low pass–low pass), LH (low pass–high pass), HL (high pass–low pass), and HH (high pass–high pass), which are combined together in a matrix. When this process is applied to 2D signals such as images, a single-level discrete wavelet transform (DWT) decomposition involves the use of a scaling function called $\varphi(x, y)$ and three wavelets denoted as $\psi(x, y)$. The computation of these wavelets can be outlined as follows:

$$\varphi(x, y) = \varphi(x)\varphi(y) \tag{1}$$

$$\psi^H(x, y) = \psi(x)\varphi(y) \tag{2}$$

$$\psi^V(x, y) = \varphi(x)\psi(y) \tag{3}$$

$$\psi^D(x, y) = \psi(x)\psi(y) \tag{4}$$

In this context, the scaling function $\varphi(x, y)$ represents the low-frequency component or the LL band, which captures overall variations in the image. The column variations, denoted by the LH band, are obtained by measuring changes along the columns. Similarly, the row variations, referred to as the HL band, are detected by the sensitivity of the wavelet function $\psi^V(x, y)$ to changes along the rows. Finally, diagonal variations, corresponding to the HH band, are simulated by the wavelet function $\psi^D(x, y)$ to replicate variations along the diagonal direction. Therefore, the 2D-DWT (discrete wavelet transform) of an image represented by *g(x,y)* with dimensions $M \times M$ is:

$$W_\varphi(j_0, m, m) = \frac{1}{\sqrt{MM}} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} g(x, y)\varphi_{j_0,m,m}(x, y) \tag{5}$$

$$W_\psi^i(j, m, m) = \frac{1}{\sqrt{MM}} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} g(x, y)\psi_{j,m,m}^i(x, y) \qquad i = \{H, V, D\} \tag{6}$$

j_0 represents an arbitrary initial scale, and the coefficients $W_\varphi(j_0, m, m)$ serve as an approximation of the function $g(x, y)$ at scale j_0. On the other hand, the coefficients $W_\psi^i(j, m, m)$ contribute additional details in the horizontal, vertical, and diagonal

directions for scales $j \geq j_0$. Typically, j_0 is set to 0, and $M = M = 2^J$ is chosen, where $j = 0, 1, 2, ...J - 1$ and $m = m = 0, 1, 2, ..., 2^j - 1$.

Cosine domain insertion

The discrete cosine transform (DCT) is a mathematical method widely employed in signal processing and data compression. It is also utilized in specific types of steganography to conceal data within digital media like images or videos. The mathematical equations, both forward and inverse, used to compute these coefficients are[14]:

$$G(m,n) = \frac{2}{\sqrt{M \times N}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) c_m \cos\left(\frac{m(2u+1)\pi}{2M}\right) c_n \cos\left(\frac{n(2v+1)\pi}{2N}\right), \tag{7}$$

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 $v = 0, 1, \ldots N - 1$.

$$g(u,v) = \frac{2}{\sqrt{M \times N}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} G(m,n) c_m \cos\left(\frac{m(2u+1)\pi}{2M}\right) c_n \cos\left(\frac{n(2v+1)\pi}{2N}\right) \tag{8}$$

where c_m, and c_n are:

$$c_m = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } m = 0 \\ 1 & \text{otherwise} \end{cases} \tag{9}$$

In steganography, the DCT is employed on blocks or segments of the original image. Steganography techniques that utilize DCT often select specific frequency coefficients for hiding information. The selection of coefficients is typically based on their perceptual significance, favoring those that are less noticeable to the human eye. Usually, low-frequency components are commonly chosen for steganography purposes. The confidential data is typically represented as binary bits, which are then embedded by modifying the selected DCT coefficients. This modification involves adding or subtracting small values to the coefficients to encode the secret information. After embedding the data, the modified DCT coefficients are quantized and compressed. Quantization reduces the precision of the coefficients, making the changes caused by embedding less noticeable, while compression further reduces the size of the resulting stego image.

When the stego image reaches the recipient, the reverse process is applied to retrieve the hidden information. The DCT coefficients are inverted to the spatial domain through an inverse transformation, resulting in the reconstructed image. The hidden data is extracted by analyzing the modified coefficients. It's important to note that the specific techniques and algorithms used in steganography can vary, and there are numerous variations and refinements to the process just described.

Adaptive algorithm

The adaptive algorithm is a technique used in steganography to optimize the selection of coefficients for embedding secret information. The algorithm involves several steps:

1. Calculation of the total energy of the cover image;
2. Application of 2D DCT [22] to obtain the first representation of DCT coefficients;
3. Selection of a predefined number of coefficients, with the rest transformed back to the spatial domain;
4. Transformation of the current version of the image to the wavelet domain using 2D DWT. 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 the cosine and the wavelet domains.

The difference in minimizing the cost function, which measures the inconsistency between the initial energy and the energies retained in each domain, lies in the energy leftover, denoted as $\Phi(\alpha, \beta)$. Specifically, $\Phi(\alpha, \beta)$ is computed in the following manner:

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

where $[\ ]^2$ is the square of each element separately. The procedure starts by employing a steepest descent algorithm [23] to reduce the remaining error. After the iteration is complete, a specific number of coefficients are saved in two separate domains: the 2D-DCT and the 2D-DWT domains. These preserved coefficients are then combined to form the resulting feature vector for each pose. The parameters used in the training phase are as stated below. The weight matrices α and β are initially set with elements of 0.5 and 0.3, respectively. The updating equations for each iteration are described in [24]:

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

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

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

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

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

Performance metrics

Performance metrics are utilized to assess how well a system, process, algorithm, or model performs in terms of its effectiveness, efficiency, accuracy, or quality. The selection of performance metrics relies on the particular task or application at hand.

Mean squared error (MSE)

Mean squared error (MSE) is a metric commonly used in regression to gauge the average squared disparity between the pixel values of the initial signal and the reconstructed or compressed signal. It offers a numerical assessment of the overall distortion existing between these two signals. The mathematical expression for MSE is as follows:

$$MSE = \frac{1}{m \times n} \sum_{m} \sum_{n} (I(x, y) - K(x, y))^2 \tag{15}$$

In the formula, the variable $I(x, y)$ represents the pixel value of the original signal at a specific position (x, y), $K(x, y)$ represents the pixel value of the reconstructed or compressed signal at the same position, and $(m * n)$ represents the total number of pixels in the image. A reduced MSE value indicates a smaller average difference, suggesting a higher quality of reconstruction or compression. However, MSE alone may not offer a perception-based measure of quality since it does not account for the human visual system's sensitivity to various image characteristics.

Peak signal-to-noise ratio (PSNR)

Peak signal-to-noise ratio (PSNR) is a logarithmic metric that establishes a connection between the maximum achievable power of a signal (such as the highest pixel value) and the power of the noise (i.e., the dissimilarity between the original and reconstructed/compressed signals). PSNR is typically measured in decibels (dB). The mathematical representation for PSNR is as follows:

$$PSNR = 10 * \log_{10}(MAX^2/MSE) \tag{16}$$

The maximum pixel value, represented as MAX (such as 255 for an 8-bit grayscale image), is used in calculating the PSNR (peak signal-to-noise ratio). PSNR is a measure of quality that considers the range of pixel values and follows a logarithmic scale, making it more relevant in terms of human perception. A higher PSNR value signifies better quality since it represents a lower ratio of noise to the maximum strength of the signal.

Structural similarity index (SSIM)

The SSIM reflects the similarity between two images, whole or parts, by providing a quantitative assessment of how well the perceived structural information of an image is preserved after undergoing various processing. The SSIM takes into account 3 important components, namely, luminance, contrast, and structure. The SSIM index falls between $[-1, 1]$, with 1 indicating perfect similarity between the images. This index is calculated as follows:

$$SSIM(x, y) = [l(x, y) * c(x, y) * s(x, y)] \tag{17}$$

where x and y are the two input images (or windows) being compared, $l(x, y)$ represents the luminance comparison, $c(x, y)$ represents the contrast comparison, and $s(x, y)$ represents the structure comparison. Each of these coefficients is calculated as follows:

$$\begin{aligned} l(x,y) &= \frac{2\mu_x\mu_y + C1}{\mu_x^2 + \mu_y^2 + C1} \\ C1 &= (K1 * L)^2 \\ c(x,y) &= \frac{2\sigma_x\sigma_y + C2}{\sigma_x^2 + \sigma_y^2 + C2} \\ C2 &= (K2 * L)^2 \\ s(x,y) &= \frac{\sigma_{xy} + C3)}{\sigma_x\sigma_y + C3} \end{aligned} \tag{18}$$

where μ is mean, σ is standard deviation, σ^2 is the variance, $K1 = 0.01$, $L = 1$ (the dynamic range of pixel values for gray-scale images, $L = 1$), $K2 = 0.03$, and C3 is a small constant.

General comments on the performance metrics

It should be emphasized that MSE (mean squared error), PSNR (peak signal-to-noise ratio), and structural similarity index (SSIM) have their own limitations. They do not perfectly encompass all aspects of image quality, such as the way humans perceive visual information, and may not consistently align with subjective evaluations. Consequently, it is advisable to utilize these metrics in conjunction with other methods for assessing quality and to take into account the unique requirements and characteristics of the particular application or task.

Proposed technique

In Fig. 1, the proposed technique is depicted. The process begins by dividing the original image into non-overlapping blocks. These blocks are then transformed into the wavelet domain using the 2D DWT (2D discrete wavelet transform). To determine the weights of each coefficient within the wavelet domain, an adaptive algorithm described in "Adaptive Algorithm" is applied to each block individually. The block(s) with coefficients exhibiting lower total weights compared to the other blocks is(are) selected. These selected coefficients are converted into a binary representation called "cover in binary." The secret data, in its binary form, is then embedded in the least significant bit (LSB) of the cover in binary. Afterwards, the block is converted back to its decimal representation. To obtain the stegoimage, a 2D IDWT (2D inverse discrete Haar transform) is applied. At the receiver's end, the recipient can extract the secret data by partitioning the image into blocks with the same dimensions as those used in the encoding process. Additionally, the index of the chosen block must be securely transmitted to the receiver. The effectiveness of the suggested method is assessed by employing evaluation metrics like mean squared error (MSE), peak signal-to-noise ratio (PSNR), and structured similarity index (SSIM) in addition to human visual examination. A comparative analysis is performed between the proposed system and recently reported approaches.

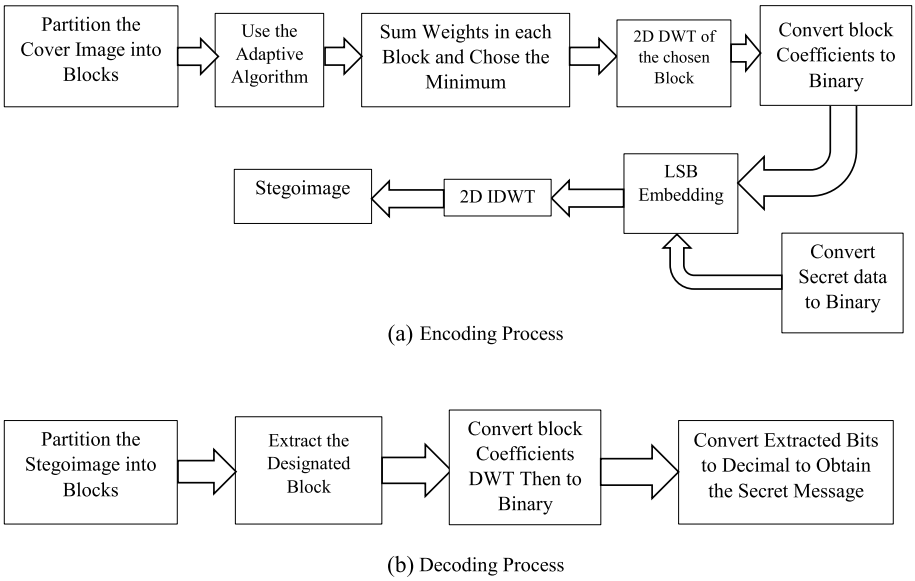


Fig. 1 The proposed technique utilized in steganography system. The two modules of the system are shown

Results

The results are presented as two scenarios, namely, BOSS database, and customized database.

First scenario: BOSS database

In this part of the results, the proposed method is tested using a set of 10 gray-scale samples obtained from the BOSSBase database (Break Our Steganographic System Base) [19]. This database comprises 10,000 black and white images designed for experiments related to detecting hidden data in JPEG images. It includes various features extracted from clean images, as well as images with hidden random data using different techniques such as the JPEG universal wavelet relative distortion, the *nsF5* method, and the uniform embedding revisited distortion (UERD) algorithm. The features include discrete cosine transform residuals (DCTR), Gabor filter residuals (GFR), and the phase-aware projection model (PHARM). To study the effect of the block size on the PSNR, the following non-overlapping block sizes were chosen: $4 \times 4, 8 \times 8, 16 \times 16, 32 \times 32, 64 \times 64$, and 128×128. Figure 2 shows samples from The BOSS database (original cover and stego images with different block sizes).

To study the effect of the secret message size on the PSNR of the stegoimage, the following message sizes were chosen: 4, 6, 8, 10, 12, 14, and 16 KB (kilobytes $= 1024 \times 8$ bits). Figure shows PSNRs for all images with different message size.

Second scenario: customized database

In this part of the results, the proposed method is tested using a set of 5 RGB images that appeared in [10]. The unprocessed samples of these images are shown in Fig. 3. The proposed approach is compared with 7 other approaches explained in [10].

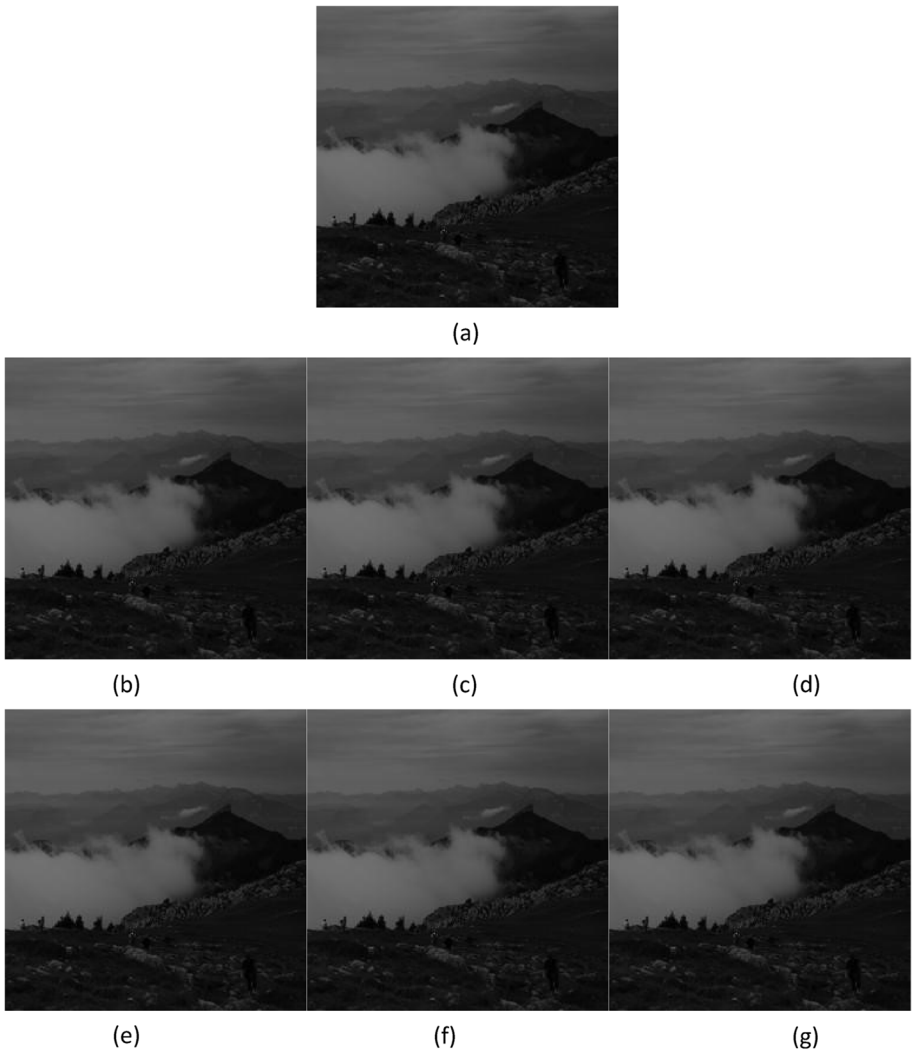


Fig. 2 Samples from the BOSS database/proposed LSB technique. **a** Original cover image, **b** stegoimage with block size of 4, **c** stegoimage with block size of 8, **d** stegoimage with block size of 16, **e** stegoimage with block size of 32, **f** stegoimage with block size of 64, and **g s**tegoimage block size of 128

These are labeled in the results as follows: 1*st* Modified LSB RGB [25], 2*nd* Improved LSB for RGB [26], 3*rd* LSB Replacement through XOR [27], 4*th* Multi-Stego for Grey Scale [28], 5*th* Value Difference using adjacent pixel LSB [29], 6*th* Grey Level Modification and Multi Level Encryption [30], and 7^th Novel Least Significant Bit Technique [10].

Each cover image (the red layer only) is partitioned into non-overlapping blocks of 4×4 and a 16 KB message is embedded in it. No pre-processing step is utilized except the image resizing to obtain the required dimensions. Two image dimensions, 512×512 and 1024×1024, were chosen to have a fair comparison with other recently reported results and to fit the Haar transform requirement.

To study the effect of the block size on the PSNR, the following non-overlapping block sizes were chosen: $4 \times 4, 8 \times 8, 16 \times 16, 32 \times 32, 64 \times 64$, and 128×128.

Image 1 Image 2

Image 3 Image 4

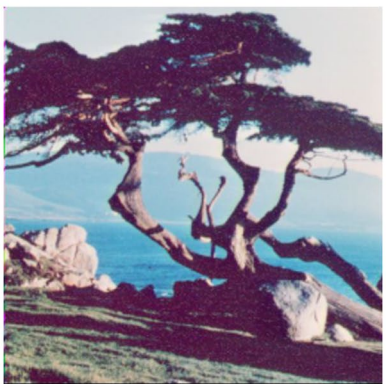


Image 5

Fig. 3 Unprocessed samples of the customized database

Results discussion

Based on the results displayed, the suggested method outperformed the other techniques being compared. It consistently maintained lower levels of mean squared error (MSE) while achieving higher peak signal-to-noise ratios (PSNRs) and SSIMs that are close to 1 in the majority of cases. Moreover, upon visual examination by humans, it was observed that the proposed technique did not alter the visual characteristics of the original image.

First database

The results of this database, in terms of PSNRs, are shown in Fig. 4 for both Spatial and proposed techniques. Figure 2 contains samples, before and after processing step, from The BOSS database/proposed LSB techniques. Original cover image and stego images with different message sizes are shown in this figure. As shown in the results, the proposed technique outperformed, or at least was at the same performance level except for two cases. Figure 5 shows the MSEs for different message sizes. The parts of

Fig. 4 PSNRs for the BOSS database/proposed LSB technique with different message sizes with 4 × 4 block size. **a** Message size = 4 KB, **b** message size = 6 KB, **c** message size = 8 KB, **d** message size = 10 KB, **e** message size = 12 KB, **f** message size = 14 KB, and **g** message size = 16 KB

Fig. 5 MSEs for the BOSS database/proposed LSB technique with different message sizes with 4 × 4 block size. **a** Message size = 4 KB, **b** message size = 6 KB, **c** message size = 8 KB, **d** message size = 10 KB, **e** message size = 12 KB, **f** message size = 14 KB, and **g** message size = 16 KB

the charts that are barely seen refer to low error values. The SSIMs were equal to, or very close to, 1 for all cases.

Second database

Figure 3 shows processed samples of the customized database with 4 × 4 blocks, and 16 KB secret message size. Figure 6 shows the results, in terms of PSNR, obtained from the proposed approach and the other 7 approaches in comparison when the cover images' dimensions are 1024 × 1024. As shown in this figure, the proposed approach is outperformed the rest of the approaches by several dBs. In addition, the

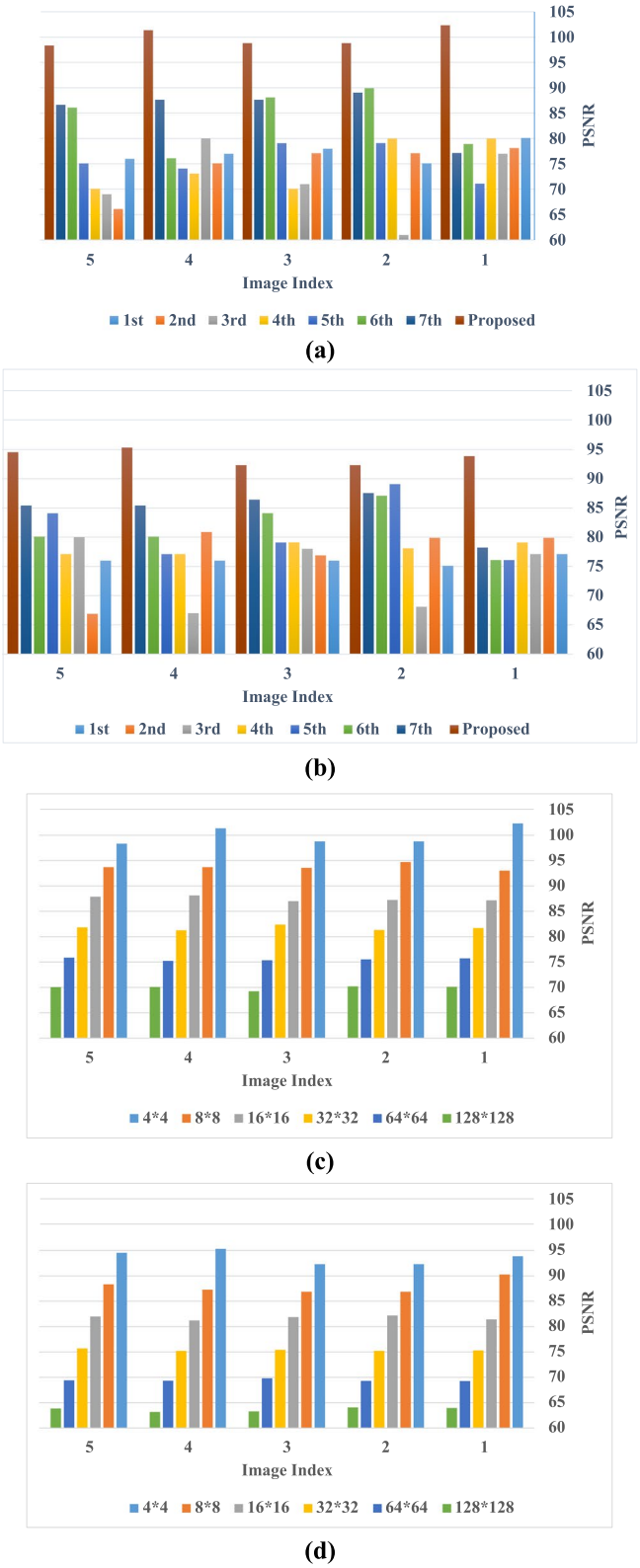
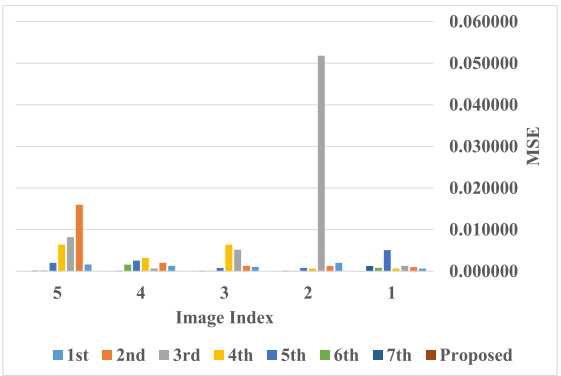
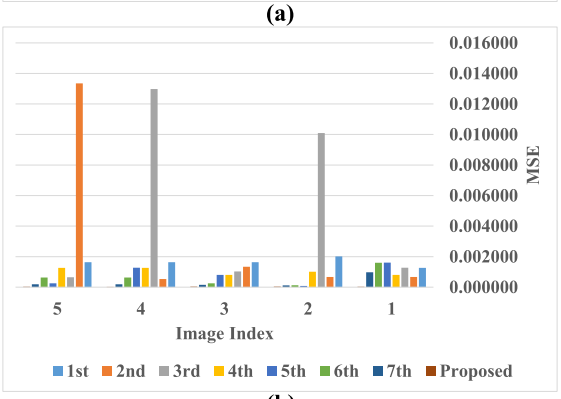


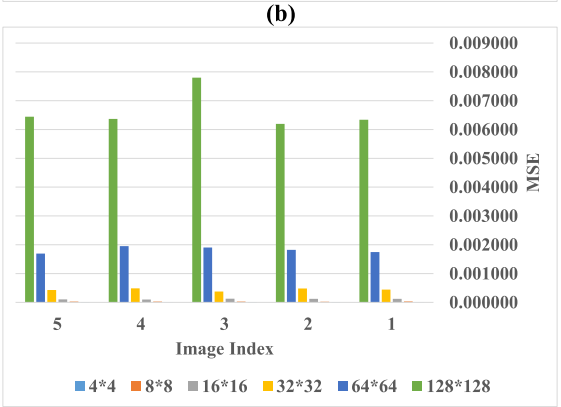
Fig. 6 **a** PSNRs for proposed approach and other recently published results/customized database/image dimensions are 1024×1024, **b** image dimensions are 512×512, **c** different block size/customized database/image dimensions are 1024×1024, and **d** different block size/customized database/image dimensions are 512×512



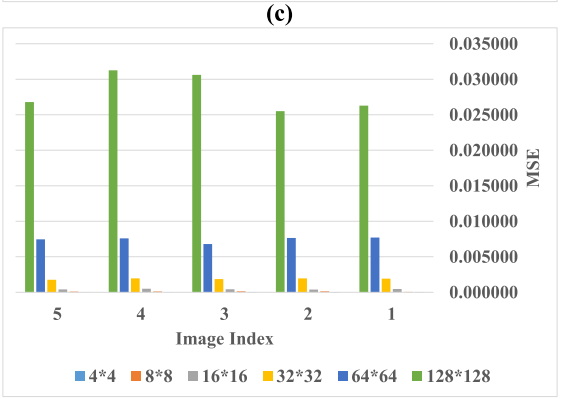
(a)



(b)



(c)



(d)

Fig. 7 **a** MSEs for the proposed approach and other recently published results/customized database/image dimensions are 1024×1024, **b** image dimensions are 512×512, **c** different block size/customized database/image dimensions are 1024×1024, and **d** different block size/customized database/image dimensions are 512×512

figure shows the results, in terms of PSNR, obtained from the proposed approach and the other 7 approaches in comparison when the cover images' dimensions are 512×512. As shown in this figure, the proposed approach is also outperformed the rest of the approaches. The SSIMs were equal to, or very close to, 1 for all cases. Figure 6 shows the PSNRs of the proposed approach when block size is changed to values mentioned above for image sizes 1024×1024 and 512×512, receptively. As shown in these subfigures, the less the block size (i.e., less coefficients are selected and processed individually), the higher PSNR obtained. Nevertheless, small block size leads to increase the number of total blocks that have to be processed by the system and hence more processing time is required. On the other hand, Fig. 7 show the MSEs obtained from the proposed approach and the other 7 approaches in comparison when the cover images' dimensions are 1024×1024, 512×512, different block size/customized database/image dimensions are 1024×1024, and different block size/customized database/image dimensions are 512×512. The results clearly show the better performance of the proposed approach compared with the rest of the reported approaches.

Conclusions

A new approach is presented for concealing a hidden message within an image, utilizing the two-dimensional discrete wavelet transform (2D DWT). The method involves transforming the image into the Wavelet domain using 2D DWT, followed by the selection of a specific number of coefficients to embed the binary secret message. This selection process incorporates an analysis of the image in two distinct domains: 2D DCT and 2D discrete wavelet transform. The analysis is adaptively executed to minimize any potential alterations to the original image. An adaptive algorithm is employed to assign weights to each coefficient in both domains, with lower-weighted coefficients chosen for embedding the secret message. To assess the efficacy of the technique, Grey scale samples from the BOSSbase and a RGB costumed databases were utilized, and three metrics, mean squared error (MSE), peak signal-to-noise ratio (PSNR), and structural similarity index (SSIM) were employed. Additionally, a visual examination of the resulting image by human observers was taken into consideration. The results clearly indicated that the proposed technique outperformed other recently reported techniques MSE, PSNR, SSIM, and visual quality.

Acknowledgements
Not applicable.

Authors' contributions
Alobaidi: conceptualization, methodology, modeling, reviewing, interpretation, writing, and editing. Mikhael: supervising. All authors have read and approved the manuscript.

Funding
No funding was obtained for this study.

Availability of data and materials
The BOSSbase database analyzed during the current study are available in the following website: https://www.kaggle.com/datasets/heyongchong/bossbase-256-4.

Code availability
The program codes (written in MATLAB) during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 15 June 2023 Accepted: 5 October 2023
Published online: 23 November 2023

References

1. Hussain H, Khan AA, Ahmed A, Haleem M (2023) A comprehensive study of digital image steganographic techniques. IEEE Access https://doi.org/10.1109/ACCESS.2023.3237393
2. Eid WM, Alotaibi SS, Alqahtani HM, Saleh SQ (2022) Digital image steganalysis: current methodologies and future challenges. IEEE Access 10:92321–92336
3. Halboos EHJ, Albakry AM (2022) Hiding text using the least significant bit technique to improve cover image in the steganography system. Bull Electr Eng Inform 11(6):3258–3271
4. Chang CC (2022) Bayesian neural networks for reversible steganography. IEEE Access 10:36327–36334
5. Shehab DA, Alhaddad MJ (2022) Comprehensive survey of multimedia steganalysis: Techniques, evaluations, and trends in future research. Symmetry 14(1):117
6. Peter G, Sherine A, Teekaraman Y, Kuppusamy R, Radhakrishnan A, et al (2022) Histogram shifting-based quick response steganography method for secure communication. Wirel Commun Mob Comput 2022. https://doi.org/10.1155/2022/1505133
7. Rustad S, Syukur A, Andono PN et al (2022) Inverted lsb image steganography using adaptive pattern to improve imperceptibility. J King Saud Univ-Comput Inf Sci 34(6):3559–3568
8. Aslam MA, Rashid M, Azam F, Abbas M, Rasheed Y, Alotaibi SS, Anwar MW (2022) Image steganography using least significant bit (lsb)-a systematic literature review. In: 2022 2nd International Conference on Computing and Information Technology (ICCIT). IEEE pp 32–38
9. Sharma VK, Kumar P, Singhal S, Soni BP, Shukla PK (2023) Secret image scrambling and dwt-based image steganography using smoothing operation and convolution neural networks. J Discrete Math Sci Cryptogr 26(3):695–705. https://doi.org/10.47974/JDMSC-1742
10. Rahman S, Uddin J, Khan HU, Hussain H, Khan AA, Zakarya M (2022) A novel steganography technique for digital images using the least significant bit substitution method. IEEE Access 10:124053–124075
11. Subramanian N, Elharrouss O, Al-Maadeed S, Bouridane A (2021) Image steganography: A review of the recent advances. IEEE Access 9:23409–23423
12. Elharrouss O, Almaadeed N, Al-Maadeed S (2020) An image steganography approach based on k-least significant bits (k-lsb). In: 2020 IEEE international conference on informatics, IoT, and enabling technologies (ICIoT). IEEE pp 131–135
13. Mstafa RJ, Elleithy KM, Abdelfattah E (2017) A robust and secure video steganography method in DWT-DCT domains based on multiple object tracking and ECC. IEEE Access 5:5354–5365
14. Das R, Tuithung T (2012) A novel steganography method for image based on huffman encoding. In: 2012 3rd National Conference on Emerging Trends and Applications in Computer Science. IEEE pp 14–18
15. Alobaidi T, Mikhael WB (2019) Mixed nonorthogonal transforms representation for face recognition. Circ Syst Signal Process 38:1684–1694
16. Alobaidi T, Mikhael WB (2018) A modified discriminant sparse representation method for face recognition. In: 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC). IEEE pp 727–730
17. Alobaidi T, Mikhael WB (2019) A transform domain implementation of sparse representation method for robust face recognition. Circ Syst Signal Process 38:4302–4313
18. Alobaidi T, Mikhael WB (2018) A wavelet domain implementation of sparse representation method for face recognition. In: 2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE pp 214–217
19. Płachta M, Krzemień M, Szczypiorski K, Janicki A (2022) Detection of image steganography using deep learning and ensemble classifiers. Electronics 11(10):1565
20. Sturm BL (2007) Stéphane mallat: A wavelet tour of signal processing. 2nd Ed. Comput Music J 31(3):83–85. https://doi.org/10.1162/comj.2007.31.3.83
21. Jähne B (2005) Digital image processing. Springer Science & Business Media, NY
22. Sayood K (2017) Introduction to data compression. Morgan Kaufmann
23. Widrow B, McCool J (1976) A comparison of adaptive algorithms based on the methods of steepest descent and random search. IEEE Trans Antennas Propag 24(5):615–637
24. Ramaswamy A, Mikhael WB (1993) Multitransform/multidimensional signal representation. In: Proceedings of 36th Midwest Symposium on Circuits and Systems. IEEE pp 1255–1258
25. Almazaydeh L (2020) Secure RGB image steganography based on modified LSB substitution. Int J Embed Syst 12(4):453–457
26. Singh A, Singh H (2015) An improved lsb based image steganography technique for rgb images. In: 2015 IEEE International Conference on electrical, computer and communication technologies (ICECCT). IEEE pp 1–4

27. Bhuiyan T, Sarower AH, Karim R, Hassan M (2019) An image steganography algorithm using lsb replacement through xor substitution. In: 2019 International Conference on Information and Communications Technology (ICOIACT). IEEE pp 44–49
28. Sahu AK, Swain G (2019) Dual stego-imaging based reversible data hiding using improved LSB matching. Int J Intell Eng Syst 12(5):63–73
29. Kalita M, Tuithung T, Majumder S (2019) An adaptive color image steganography method using adjacent pixel value differencing and LSB substitution technique. Cryptologia 43(5):414–437
30. Muhammad K, Ahmad J, Farman H, Jan Z, Sajjad M, Baik SW (2015) A secure method for color image steganography using gray-level modification and multi-level encryption. KSII Trans Internet Inf Syst (TIIS) 9(5):1938–1962

Publisher's Note