

# A Blind High-Capacity Wavelet-Based Steganography Technique for Hiding Images into other Images

Safwat HAMAD<sup>1</sup>, Amal KHALIFA<sup>1,2</sup>, Ahmed ELHADAD<sup>3,4</sup>

<sup>1</sup>*Department of Scientific Computing, Faculty of Computer & Information Sciences, Faculty of Computer & Information Sciences, Ain Shams University, Cairo, 11566, Egypt*

<sup>2</sup>*College of Computer & Information Sciences, Princess Nora University, Riyadh - KSA*

<sup>3</sup>*Departamento de Matemática, Instituto Superior Técnico, Lisbon, 1049, Portugal*

<sup>4</sup>*Faculty of Science, South Valley University, Qena, 83523, Egypt*

*ahmed.elhadad@tecnico.ulisboa.pt*

**Abstract**—The flourishing field of Steganography is providing effective techniques to hide data into different types of digital media. In this paper, a novel technique is proposed to hide large amounts of image data into true colored images. The proposed method employs wavelet transforms to decompose images in a way similar to the Human Visual System (HVS) for more secure and effective data hiding. The designed model can blindly extract the embedded message without the need to refer to the original cover image. Experimental results showed that the proposed method outperformed all of the existing techniques not only imperceptibility but also in terms of capacity. In fact, the proposed technique showed an outstanding performance on hiding a secret image whose size equals 100% of the cover image while maintaining excellent visual quality of the resultant stego-images.

**Index Terms**—blindness, digital images, discrete wavelet transforms, information security, payloads.

## I. INTRODUCTION

The increasing dependency on digital media has created an urgent request for particular techniques to protect communication as well as materials from illegal usage. Popular sites and social networks allow anyone to easily upload, download, and exchange photos all over the world, at any time and free of charge. A question might be raised about the content of those posts, photos and videos; are they truly innocent as they appear to be or are they hiding some other information a normal eye can't notice?

Information hiding techniques provide the right computational tools to emulate camouflage in nature. That is, it uses a host (cover) media to hide or embed a piece of information (message) in such a way that it is imperceptible to a human observer but can be detected/extracted easily with a computer. These covers can take the form of any digital media such as audio tracks, Videos [1], images [2], File systems [3], networks [4], 3D objects [5], and even DNA sequences [6].

In fact, information hiding is an ancient science rooted back to 440 B.C. where the Greek used to send their messages hidden under wax or tattooed on a messenger head especially during war times. When compared to cryptography, the main advantage of information hiding is that the content is inseparable of the hidden message and

hence protects both the message and the parties involved in the communication.

In this paper, we propose a novel wavelet-based technique to enhance the hiding capacity of the existing image-steganography methods. The proposed method can hide images into true colored ones. The rest of the paper is organized as follows: the next section provides a quick overview on the literature of information hiding in images. Section three describes the hiding/extraction model of the proposed technique. Experimental results are presented and analyzed in section 4, where a performance comparison was held between the proposed technique and other methods highlighting weaknesses and strengths of each one of them over the others. Finally, section 5 summarizes the findings and conclusions.

## II. RELATED WORK

Although modern information hiding methods can be applied on various digital media, this review is going to focus on techniques for digital images. In fact, all information hiding techniques attempt to find adequate answers to three main questions: “What”, “How”, and “Where”. Firstly, “What to hide?” In other words, what is the format of the secret message? Some techniques are used to embed any kind of binary data as long as they can be converted into a stream of bits [7], [8]. Another group of techniques were developed mainly to hide certain types of messages such as images [9], logos [10], or even text images [11]. In dual watermarking; instead of embedding one image message, two images are embedded for increased protection and security [12].

Secondly, “How to carry out the hiding process?” The answer to this question can be used to categorize various techniques into a number of classes based on different criteria. One categorization is based on the domain of embedding. In this context, Spatial-domain Techniques embed the secret information directly in the pixel illumination values of the image [13], [14]. On the other hand, Transform-domain Techniques hide the message by modulating coefficients in some transform domain, such as the Fourier Transform (FT) [15], Discrete-Cosine Transform (DCT) [16], and Discrete Wavelet transform (DWT) [7], [10], [11], [17-20]. Some methods may combine more than

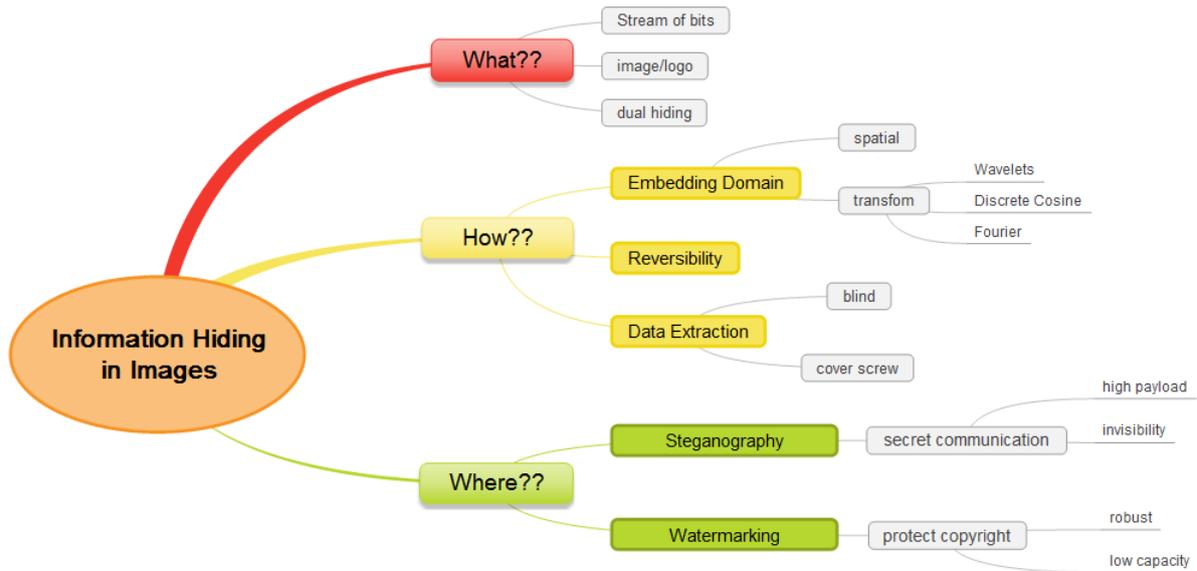


Figure 1. Three main questions categorize different information hiding techniques in images.

a transform to implement their hiding process [21], [22]. Another orthogonal categorization differentiates between blind and non-blind (cover-Screw) schemes. In blind, or oblivious, schemes allows the hidden data to be extracted directly from the modified cover without knowledge of the original image [11], [17], [18], while in non-blind schemes the original cover is needed to reveal the hidden information [9], [21], [23]. Obviously, blind techniques are preferred over the non-blind ones since they are more practical and reliable.

A new concept of dual embedding mixes both blind and non-blind algorithms where a sign image is embedded into a logo in a non-blind fashion to create a signed-logo which is then embedded into the cover image in a blind fashion [12]. Furthermore, some critical applications add another requirement to the hiding process: reversibility. In other words, a reversible embedding technique must guarantee the lossless recovery of the original host image when required [19]. Since most of the conventional transforms are irreversible, some hiding techniques employed the integer-to-integer wavelet transform to prevent coefficients from being potentially lost through forward and inverse transforms due to any truncation or rounding errors [8], [24].

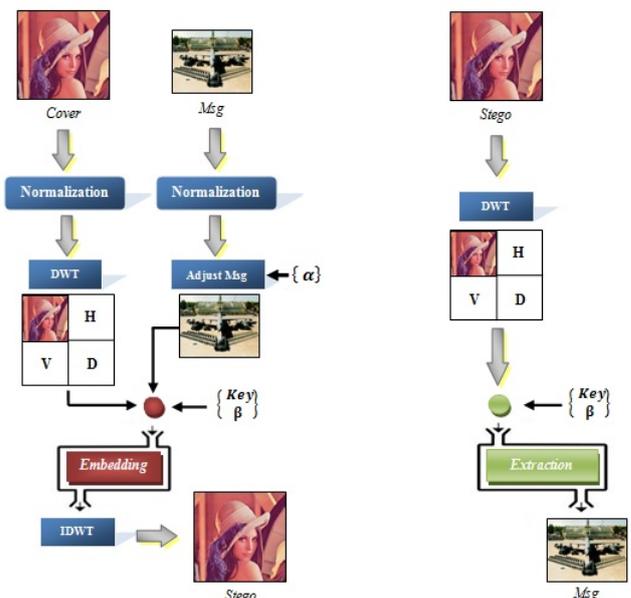
Finally, the “Where” question is concerned with finding the parts of the cover image that should be utilized for embedding. In fact, the answer to this question should be regarded in the light of an application. For example, the objective of Steganography applications is to provide a means of secret communication that is imperceptible to the human visual system [9], [14], [25]. On the other hand, watermarking algorithms must be robust to protect copyright information against certain attacks such as filtering, noise addition, and compression attacks [7], [11], [18], [20], [23]. These diverse goals create a tradeoff between invisibility and robustness. That is, embedding data in significant parts of the image will probably make it survive against attacks. However, this can strongly influence their hiding capacity as well as the perceptual quality of the embedded images.

Figure 1 shows a mind map that summarizes the above discussion. It clearly implies that a perfect hiding technique

can't be found. Instead, designing a practical one can be a tough task trying to achieve the right balance among three conflicting requirements: invisibility, robustness, and capacity.

### III. THE PROPOSED METHOD

Research into human perception indicates that the retina of the eye splits an image similar to the multi-resolution decomposition of the DWT [19], [22]. With this intrinsic similarity to the Human Visual System (HVS) perception, DWT is expected to make the process of imperceptible embedding more effective. Therefore, the proposed steganographic method takes advantage of the properties of 2D wavelet transforms in order to hide a secret image into another one. The cover and the secret images are both assumed to be true colored images. However, the same technique can be applied on grayscale images. The overall scheme is depicted in Figure 2.



(a) The hiding process

(b) The extraction process

Figure 2. The proposed steganographic model

A. Multi-resolution Wavelet Transforms

The wavelet transform is identical to a hierarchical sub-band system, where the sub-bands are logarithmically spaced in frequency. As shown in Figure 3, in a one dimensional discrete wavelet transform (DWT), the input signal (s) is convolved with a low pass filter to produce a smoothed version of the input (A) and a high pass filter to capture the detail coefficients (D).

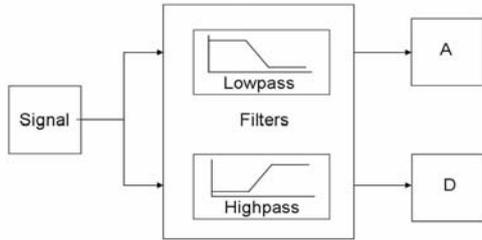


Figure 3. One-dimensional discrete wavelet transform

Images are obviously two dimensional signals that can be transformed by applying the one dimensional transform to the rows and columns of the image successively. The result is shown in Figure 4 and is decomposed into four quadrants with different interpretations: the upper left quadrant (LL) represents the approximated version of the image at half the resolution. The lower left and the upper right blocks (HL and LH) reflect vertical and horizontal details respectively. Finally, the (HH) block represents the diagonal features of the image.

The same two dimensional wavelet transform can be recursively applied on the (LL) quadrant to generate more detail coefficients at different scales, as shown in Figure 5. In this case, the sub-bands in the *l*'th transform level can be denoted by LL<sup>*l*</sup>, LH<sup>*l*</sup>, HL<sup>*l*</sup> and HH<sup>*l*</sup>. That's why this type of image decomposition is known as multi-resolution scheme or multi-scale representation. Afterwards, the inverse of the DWT is called the reconstruction process in which the original image can be synthesized from the coefficients belonging to different sub-bands.

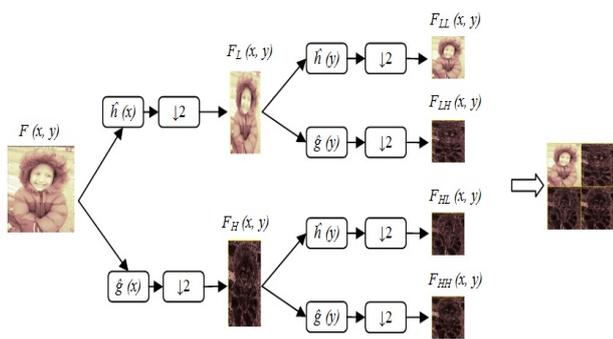


Figure 4. One dimensional DWT transform are applied to the rows and columns of the image successively

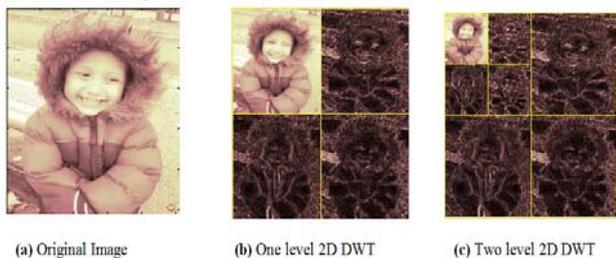


Figure 5. A multi-scale resolution of an image using 2D wavelet transforms

B. The Embedding Module

As mentioned before, the embedding process actually takes place in wavelet domain of the cover image. Therefore, the cover image is normalized in order to get float pixel values that ranges between 0.0 and 1.0 instead of the integer range of (0 - 255). Similarly, the secret image is normalized as well. As shown in Figure 2, once a 2D DWT is applied on the normalized cover image, the next step carries out the hiding process on the normalized coefficients of the cover. In this paper, we propose a novel technique that replaces these coefficients by the pixel values of the secret image using Equation 1.

$$Stego_{x,y}^s = \frac{2}{\beta}(Msg + i), \quad \frac{2i}{\beta} \leq Cover_{x,y}^s < \frac{2(i+1)}{\beta} \quad (1)$$

$$i = \begin{cases} 0, 1, 2, 3, \dots, (\beta-1) & S = A \\ -2, -1, 0, 1, \dots, (\beta-3) & S = H, V, D \end{cases}$$

Where, Msg refers to an adjusted normalized pixel of the secret image, Stego<sub>x,y</sub><sup>s</sup> refers to the resultant coefficient of the stego-image and Cover<sub>x,y</sub><sup>s</sup> is the corresponding cover coefficient. The (x,y) values specify the coordinates of the selected coefficients at a certain sub-band (s). The four sub-bands are utilized for embedding, where A, H, V, D stand for approximation, horizontal, vertical, and diagonal sub-bands respectively. Notice that; for more secured hiding, the order by which the coefficients in different sub-bands are selected for embedding can be made determined using a pseudo random permutation function based on the secret key. In this way, only the one who has the key will be able to correctly extract the secret image.

Equation 1 also specifies an additional parameter (β) that will be used for embedding. β indicates the number of intervals that will be used to divide the range of the cover coefficients where the range is actually determined by the minimum and maximum coefficient values in each specific sub-band. For example, when β = 4 and s represents the approximation sub-band, stego coefficients will be computed according to the following rules:

$$Stego = \begin{cases} \frac{1}{2}Msg, & 0 \leq Cover < \frac{1}{2} \\ \frac{1}{2}Msg + \frac{1}{2}, & \frac{1}{2} \leq Cover < 1 \\ \frac{1}{2}Msg + 1, & 1 \leq Cover \leq 1\frac{1}{2} \\ \frac{1}{2}Msg + 1\frac{1}{2}, & 1\frac{1}{2} \leq Cover \leq 2 \end{cases} \quad (2)$$

That is, the Equation 2 utilized for embedding will actually depend on value of the cover coefficient. To illustrate this point, Figure 6 shows an example in which the cover coefficient lies in the second region that implies that the corresponding stego coefficient will be computed as follows:  $\frac{1}{2}(0.6) + \frac{1}{2} = 0.8$

Notice that the computed values introduce slight changes on the original ones. This decreases the possibility of the degradation that will be caused due to the hiding process.

In addition, an adjustment step takes place on the normalized pixel values of secret image using (α). This step assures that saturated pixel values would not eventually result in an overflow in the embedded stego coefficients. Finally, after the embedding process is done, the stego-

image is obtained by an inverse wavelet (IDWT). Since the pixel values were actually normalized, they need to be de-normalized to convert the pixel values back to their original integer domain. The detailed steps of the embedding process are listed in Algorithm 1.

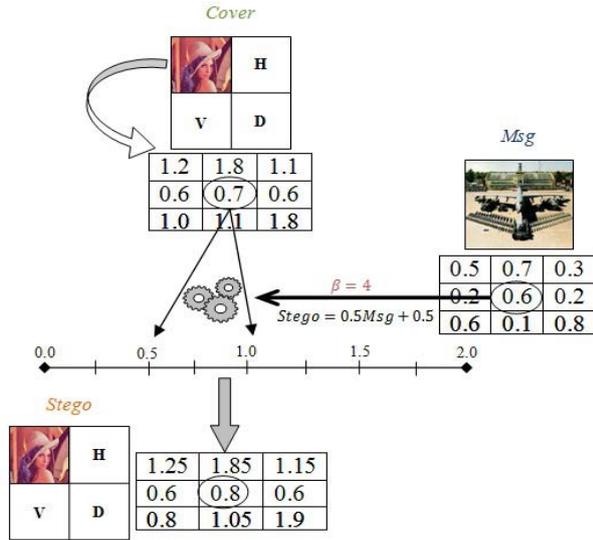


Figure 6. An example of the hiding process using  $\beta = 4$

**Algorithm 1: The Embedding Process**

*Input:* Cover, Secret images,  $\alpha$ ,  $\beta$  and Key

*Output:* Stego Image

- 1) Read Cover image  $\rightarrow$  Cover.
- 2) Read Secret image  $\rightarrow$  Msg.
- 3) Normalize both Cover and Msg.
- 4) Adjust extreme pixel values in Msg using the function:

$$Msg = \begin{cases} 1 - \alpha, & Msg = 1 \\ \alpha, & Msg = 0 \end{cases}$$

- 5) Transform Cover into wavelet domain to produce  $Cover^A$ ,  $Cover^H$ ,  $Cover^V$  and  $Cover^D$  representing Approximation, Horizontal, Vertical and Diagonal sub-bands respectively.
- 6) Use Key to randomly permute coefficients and sub-bands
- 7) Embed Msg pixels into coefficients of the selected sub-band (s) using the following parameterization function:

$$Stego_{x,y}^s = \frac{2}{\beta}(Msg + i), \quad \frac{2i}{\beta} \leq Cover_{x,y}^s < \frac{2(i+1)}{\beta}$$

- 8) Apply inverse wavelet transform on resultant Stego coefficients.
- 9) Return the reconstructed Stego image.

**C. The Extraction Module**

The steps of extraction process are exactly the inverse of those followed during the embedding phase. That is, the process starts by computing the DWT decomposition of the stego image. Next, the key is required to identify the locations at which the secret pixels were embedded. So, for each coefficient location (x, y); that was utilized for embedding, the range is subdivided according to  $\beta$  and the Msg pixel can be extracted according to the following rule (Equation 3):

$$Msg = \frac{\beta}{2} \left( Stego_{x,y}^s - \frac{2i}{\beta} \right), \quad \frac{2i}{\beta} \leq Stego_{x,y}^s \leq \frac{2(i+1)}{\beta} \quad (3)$$

$$i = \begin{cases} 0, 1, 2, 3, \dots, (\beta-1), & S = A \\ -2, -1, 0, 1, \dots, (\beta-3), & S = H, V, D \end{cases}$$

Where, Msg is the extracted pixel value of the secret image from the Stego coefficient located at coordinates (x,y) of the sub-band (s).

Figure 7 shows the inverse of the embedding process illustrated in Figure 6. In this case, the value of the stego coefficient determines the Equation that will be used for extraction. Notice that the extraction process is done blindly where Msg pixels can be extracted correctly from the stego image using only the values of  $\beta$  and the secret key. The detailed steps of the extraction process are listed in Algorithm 2.

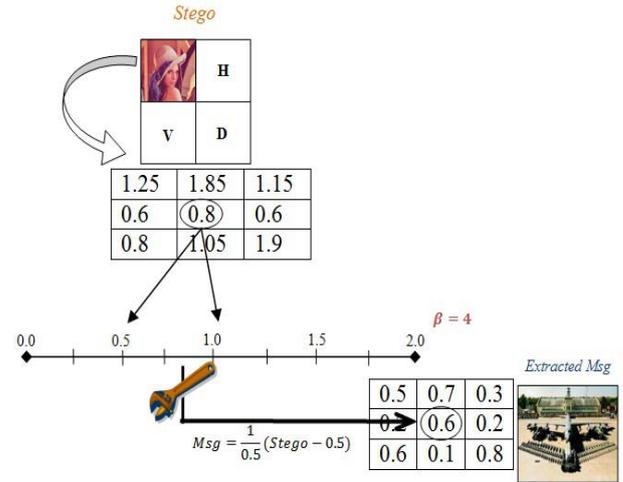


Figure 7. An example of the extraction process using  $\beta = 4$

**Algorithm 2: The Extraction Process**

*Input:* Stego Image,  $\beta$ , and Key

*Output:* Secret image

- 1) Read Stego image  $\rightarrow$  Stego.
- 2) Transform Stego into wavelet domain to produce  $Cover^A$ ,  $Cover^H$ ,  $Cover^V$  and  $Cover^D$  representing Approximation, Horizontal, Vertical and Diagonal sub-bands respectively.
- 3) Use Key to randomly permute Stego coefficients and sub-bands
- 4) Apply the inverse of the parameterization function to extract Msg pixels from coefficients of the selected sub-band (s):

$$Msg = \frac{\beta}{2} \left( Stego_{x,y}^s - \frac{2i}{\beta} \right), \quad \frac{2i}{\beta} \leq Stego_{x,y}^s \leq \frac{2(i+1)}{\beta}$$

- 5) De-normalize the extracted Msg pixels
- 6) Return the Secret image

**D. The Computational Complexity**

The counter stone of the proposed algorithm is the wavelet decomposition/reconstruction phases. These represent the most computationally intensive steps in the watermarking process. In fact, the complexity of the wavelet transform actually depends on their respective implementation. A classical analysis was provided in [26] for a number of sequential and parallel wavelet computations. Recently, a number of studies have been published to describe a derivation for the computational complexity of a number of wavelet families such as [27] and [28].

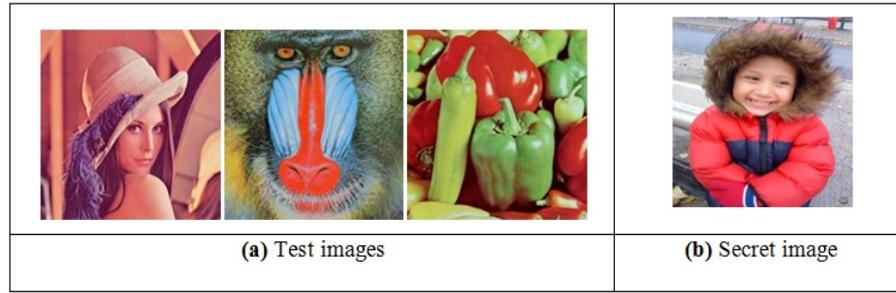


Figure 8. Images used for testing the performance of the proposed algorithm

Generally speaking, a classical 1D Haar wavelet decomposition can be as low as  $O(n)$  [29]. However, the 2D transform is just a generalization of the 1D as it is applied on rows and columns respectively. Therefore, the computational complexity of the 2D wavelet transform can't exceed  $O(n^2)$ . The same is true for the rest of the steps making up the proposed algorithm. The normalization, the permutation, and the embedding steps can be computed in  $O(n^2)$  steps as well. In conclusion, the computational complexity of the algorithm described here is  $O(n^2)$  where  $n$  represents the dimension of the cover image.

#### IV. EXPERIMENTAL RESULTS

##### A. Measures and Metrics

In this section we are going to describe the metrics used to evaluate the invisibility performance of the proposed algorithm. It is essential to have a measure by which one can judge how an image is degraded after embedding. Usually the invisibility of the hidden message is measured in terms of the Peak Signal-to-Noise Ratio (PSNR). PSNR is measured in decibels (dB) and can be computed as in Equation 4, where  $p(x, y)$  represents the shade level of a pixel, whose coordinates are  $(x, y)$  in the original image, and  $\tilde{p}(x, y)$  represents the same pixel in the distorted image.

$$PSNR = 10 \times \log\left(\frac{(\max p(x, y))^2}{MSE}\right) \quad (4)$$

$$MSE = \frac{1}{X \times Y} \sum_{x,y} (p(x, y) - \tilde{p}(x, y))^2$$

Obviously, high value of PSNR indicates that the image is less distorted. Usually, a good steganographic technique should keep the hidden data imperceptible by maintaining the PSNR at 40dB or above.

Furthermore, since the extracted message is only an estimate of the original one, we need some measure to quantify similarity between the original secret message and the extracted one. Here, we employ the normalized correlation (NC) coefficient to indicate how much of the original message was successfully extracted. It can be computed as follows (Equation 5):

$$Sim(x, x^*) = \frac{(x \times x^*) \div (\sqrt{x \times x^*})}{(x \times x) \div (\sqrt{x \times x})} \times 100 \quad (5)$$

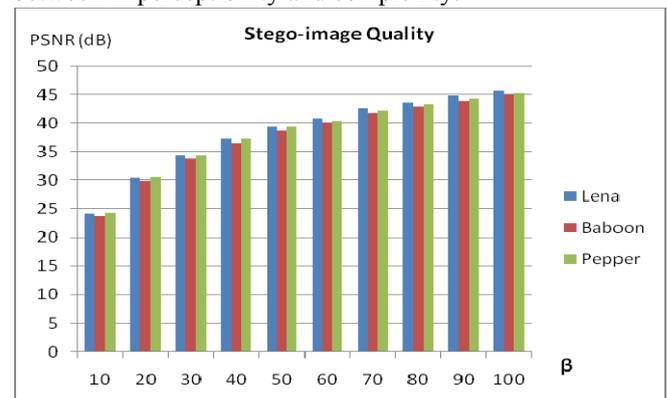
Where  $X$  is the original message components organized as a vector, and  $X^*$  is the recovered vector. Obviously, the higher similarity is the better quality of the retrieved watermark.

##### B. Invisibility Performance

Invisibility; or sometimes called imperceptibility, refers to the degree by which the embedded message doesn't introduce visible distortions to the cover image. In this section, we are going to analyze the invisibility performance of the proposed algorithm keeping into consideration a number of parameters such as  $\beta$  and the data payload.

Throughout the following sets of experiments, three  $512 \times 512$  standard test images (Lena, Baboon, and Pepper) were used as covers. The experiments were conducted on two versions of the covers: 8-bit and 16-bit formats. Furthermore, an 8-bit  $512 \times 512$  colored image was used as the secret message. These images are shown in Figure 8. The implementations of the hiding and extracting algorithms were developed using the image processing tool box in Matlab. Furthermore, all of the conducted experiments were carried out based on the Haar transform.

First, we need to investigate the effect of the parameter  $\beta$  on the fidelity of the stego-images using the maximum payload. In fact, these results can't be useful without simultaneously studying their effect on the correctness of the extracted image. Figures 9 and 10 show the computed PSNR of the stego-images as well as the similarity of the extracted messages for 8-bit cover images. In this case,  $\beta$  ranges from 10 to 100. On the other hand, Figures 11 and 12 show the same results for 16-bit cover images with  $\beta$  values ranging from 100 to 1000. These different ranges are customized to decrease the effect of the truncation errors that would result from the decimal representation of images. In both cases, the results show that higher values of  $\beta$  provides better visual quality of the stego image with minor impact on the accuracy of the extracted message. Therefore, we recommend using  $\beta = 40$  for 8-bit images and  $\beta = 500$  for 16-bit images since they provide a good compromise between imperceptibility and complexity.

Figure 9. The invisibility performance of the proposed method for 8-bit images at different values of  $\beta$

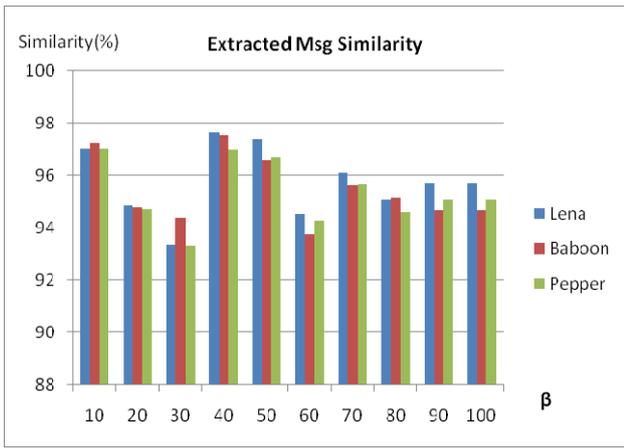


Figure 10. The accuracy of the extracted message for 8-bit images at different values of  $\beta$

Figure 13 gives a closer look on the results of the stego-images embedded with secret images of the same size. The experiment was using  $\beta = 40$  for 8-bit images and  $\beta = 500$  for 16-bit images. Simple visual inspection of the results show that in case of 16-bit images, the quality of the stego-image maintains very high (over 50 dB) while maintain almost perfect retrieval of the secret message. On the other hand, 8-bit cover images succeeded to hide a message that is as large as itself while maintaining an acceptable visual quality. That is, in conclusion, the proposed hiding technique doesn't introduce significant distortions on the embedded images even with very high data payloads.

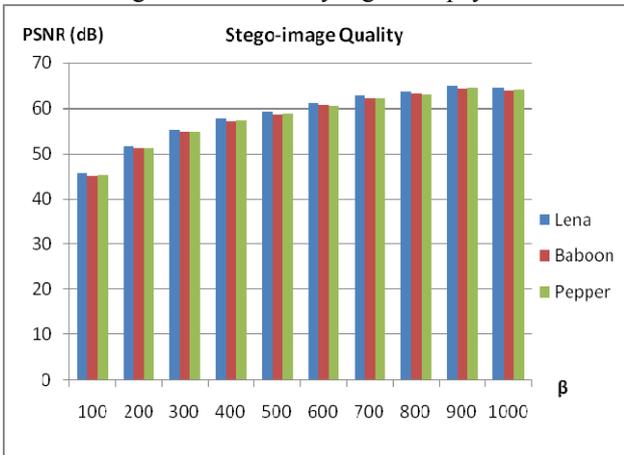


Figure 11. The invisibility performance of the proposed method for 16-bit images at different values of  $\beta$

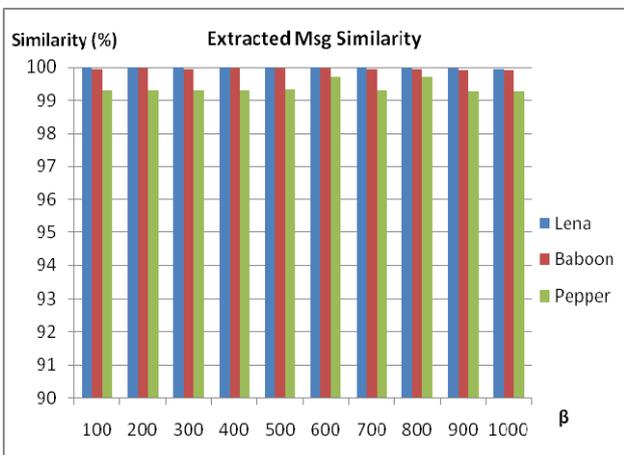


Figure 12. The accuracy of the extracted message for 16-bit images at different values of  $\beta$

### C. Comparisons with other methods

To further evaluate the performance of the proposed algorithm, several simulations have been performed and the results were compared with other existing steganography schemes. For the sake of standardization, this set of experiments used the  $512 \times 512$  colored Lena as the test image. Table 1 lists the PSNR values caused by utilizing the maximum embedding capacity provided by each algorithm. Note that those values are based on the original experimental results demonstrated in their respective publications. The table also shows the domain of embedding utilized by each method, the maximum hiding capacity offered, and whether the method is blind or not. Fundamentally, the hiding capacity offered by an algorithm is defined as the maximum amount of information that can be hidden within an image. It is usually measured in bits per pixel. However, for the sake of easier comparisons, we computed the capacity as a percentage of the cover image size as shown in Equation 6. Notice that the size here is measured in bytes.

$$Data\ Payload = \frac{Max\ size\ of\ hidden\ data}{size\ of\ image} \times 100 \quad (6)$$

Now, since the proposed algorithm can hide one byte in each cover coefficient. Equation 4 evaluates to 100% in case of 8-bit cover images and. However, in the case of 16-bit cover images the capacity falls to 50% because the secret image is still in 8-bit format. As the results in table 1 implies, a trade-off is always present between capacity and invisibility.

The results listed in Table 1 show that the proposed algorithm not only outperforms the other existing schemes in terms of invisibility, but also in capacity. More interestingly, although Tolba's [9] scheme provides the same hiding capacity, it still cannot extract the hidden data without referring to the original cover image. This is considered a great achievement of the proposed method over the existing ones.

TABLE 1. PERFORMANCE COMPARISONS WITH OTHER TECHNIQUES

Method	Domain of embedding	PSNR (dB)	Hiding Capacity	blind?	
Kawaguchi and Eason, 1998 [30]	Spatial (BPCS)	NA	30%-50%	√	
Chang et al., 2007 [31]	DCT	30.34	1.76%	√	
Lin and Shiu, 2009 [32]	DCT	34.30	2.75%	√	
Lin et al., 2010 [16]	DCT	35.28	4.30%	√	
Tolba et al., 2005 [33]	Integer WLT (N=4)	39.36	50%	√	
Jinna and Ganesan, 2010 [8]	Integer WLT	37.89	5%	√	
Spaulding et al., 2002 [34]	DWT	30	25%	√	
Tolba et al., 2004 [9]	DWT	58.40	100%	×	
Proposed	8-bit image	DWT ( $\beta=40$ )	37.41	100%	√
	16-bit image	DWT ( $\beta=500$ )	59.38	50%	√



Figure 13. Resultant stego and the extracted images where both are 512×512

## V. CONCLUSION

This paper presented a novel technique for hiding images into other images. The hiding process is carried out in the wavelet domain of the cover image where all of the four transform sub-bands are considered for embedding. The hiding function is based on a parameter  $\beta$ , which divides the range of coefficients values into non-overlapping regions. The extraction process can be done in a blind fashion where the cover image is not needed to retrieve the hidden message. Extensive experiments highlighted the outstanding performance of the algorithm which succeeded to hide an image into another one that is as large as itself without introducing any noticeable distortions in the resultant image. Furthermore, when compared with a number of existing techniques, the proposed algorithm continued its impressive performance not only in imperceptibility but also with respect to the hiding capacity it offers.

## REFERENCES

- [1] P. Davern and M. Scott, "Steganography: Its history and its application to computer based data files," Dublin City University, School of Computer Applications 1995.
- [2] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, pp. 727-752, 2010.
- [3] R. Anderson, R. Needham, and A. Shamir, "The steganographic file system," in *Proceedings of the Second International Workshop on Information Hiding*, 1998, pp. 74-84.
- [4] S. Murdoch and S. Lewis, "Embedding Covert Channels into TCP/IP," in *Information Hiding*, vol. 3727, M. Barni, J. Herrera-Joancomartí, S. Katzenbeisser, and F. Pérez-González, Eds., ed: Springer Berlin Heidelberg, 2005, pp. 247-261.
- [5] W. Kai, G. Lavoue, F. Denis, and A. Baskurt, "A Comprehensive Survey on Three-Dimensional Mesh Watermarking," *Multimedia, IEEE Transactions on*, vol. 10, pp. 1513-1527, 2008.
- [6] A. Khalifa and A. Atito, "High-capacity DNA-based steganography," in *Informatics and Systems (INFOS), 2012 8th International Conference on*, 2012, pp. BIO-76-BIO-80.
- [7] G. Thirugnanam and S. Arulsevi, "Wavelet Packet based Robust Digital Image Watermarking and Extraction using Independent Component Analysis," *International Journal of Signal & Image Processing*, vol. 1, pp. 80 – 87, 2010.
- [8] S. K. Jinna and L. Ganesan, "Reversible Image data Hiding using Lifting wavelet Transform and Histogram Shifting," *IJCSIS*, vol. 7, pp. 283-289, 2010.
- [9] M. F. Tolba, M. A. S. Ghonemy, I. A. H. Taha, and A. S. Khalifa, "High capacity image steganography using wavelet-based fusion," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, 2004, pp. 430-435 Vol.1.
- [10] L. Fan, T. Gao, and Q. Yang, "A novel zero-watermark copyright authentication scheme based on lifting wavelet and Harris corner detection," *Wuhan University Journal of Natural Sciences*, vol. 15, pp. 408-414, 2010/10/01 2010.
- [11] C.-C. Wu, Y. Su, T.-M. Tu, C.-P. Chang, and S.-Y. Li, "Saturation Adjustment Scheme of Blind Color Watermarking for Secret Text Hiding," *Journal of Multimedia*, vol. 5, pp. 248-258, 2010.
- [12] S. Tripathi, N. Ramesh, B. A., and N. K J, "A DWT based Dual Image Watermarking Technique for Authenticity and Watermark Protection," *Signal & Image Processing*, vol. 1, pp. 33 – 45, 2010.
- [13] D. Neeta, K. Snehal, and D. Jacobs, "Implementation of LSB Steganography and Its Evaluation for Various Bits," in *Digital Information Management, 2006 1st International Conference on*, 2007, pp. 173-178.

- [14] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, pp. 1613-1626, 2003.
- [15] W.-Y. Chen, "Color image steganography scheme using set partitioning in hierarchical trees coding, digital Fourier transform and adaptive phase modulation," *Applied Mathematics and Computation*, vol. 185, pp. 432-448, 2007.
- [16] C.-C. Lin and P.-F. Shiu, "High capacity data hiding scheme for dct-based images," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, pp. 220 – 240, 2010.
- [17] H. A. Abdallah, M. M. Hadhoud, A. A. Shaalan, and F. E. Abdelsamie, "Blind wavelet-based image watermarking," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 4, pp. 15 – 28, 2011.
- [18] N. Kashyap and G. R. SINHA, "Image Watermarking Using 3-Level Discrete Wavelet Transform (DWT)," *International Journal of Modern Education & Computer Science*, vol. 4, pp. 50 – 56, 2012.
- [19] A. Khan, S. A. Malik, A. Ali, R. Chamlawi, M. Hussain, M. T. Mahmood, *et al.*, "Intelligent reversible watermarking and authentication: Hiding depth map information for 3D cameras," *Information Sciences*, vol. 216, pp. 155-175, 2012.
- [20] S. Lagzian, M. Soryani, and M. Fathy, "Robust watermarking scheme based on RDWT-SVD: Embedding data in all subbands," in *Artificial Intelligence and Signal Processing (AISP), 2011 International Symposium on*, 2011, pp. 48-52.
- [21] V. Santhi and A. Thangavelu, "DC coefficients based watermarking technique for color images using singular valuedecomposition," *International Journal of Computer and Electrical Engineering*, vol. 3, pp. 8 – 16, 2011.
- [22] M. Teruya and A. Kentaro, "A blind digital image watermarking method using interval wavelet decomposition," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 3, pp. 59 – 72, 2010.
- [23] A. Khalifa and S. Hamad, "A Robust Non-blind Algorithm for Watermarking Color Images using Multi-resolution Wavelet Decomposition," *International Journal of Computer Applications*, vol. 37, 2012.
- [24] L. Sunil, C. D. Yoo, and T. Kalker, "Reversible Image Watermarking Based on Integer-to-Integer Wavelet Transform," *Information Forensics and Security, IEEE Transactions on*, vol. 2, pp. 321-330, 2007.
- [25] M. Khatirinejad and P. Lisoněk, "Linear codes for high payload steganography," *Discrete Applied Mathematics*, vol. 157, pp. 971-981, 2009.
- [26] C. K. Koc, C. Guanrong, and C. K. Chui, "Complexity analysis of wavelet signal decomposition and reconstruction," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 30, pp. 910-918, 1994.
- [27] N. F. Law and W. C. Siu, "A fast and efficient computational structure for the 2D over-complete wavelet transform," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, 2003, pp. III-309-12 vol.3.
- [28] V. Ashok, T. Balakumaran, C. Gowrishankar, and I. Vennila, "The fast Haar wavelet transform for signal & image processing," *arXiv preprint arXiv:1002.2184*, 2010.
- [29] B. Toufik and N. Mokhtar, "The Wavelet Transform for Image Processing Applications," *Advances in Wavelet Theory and Their Applications In Engineering, Physics And Technology*, pp. 395-422, 2012.
- [30] E. Kawaguchi and R. O. Eason, "Principles and applications of BPCS steganography," 1999, pp. 464-473.
- [31] C.-C. Chang, C.-C. Lin, C.-S. Tseng, and W.-L. Tai, "Reversible hiding in DCT-based compressed images," *Information Sciences*, vol. 177, pp. 2768-2786, 2007.
- [32] C.-C. Lin and P.-F. Shiu, "DCT-based reversible data hiding scheme," presented at the Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication, Suwon, Korea, 2009.
- [33] M. F. Tolba, M. A.-S. Ghoniemy, I. A.-H. Taha, and A. S. Khalifa, "Reliable blind informamtion hiding into colored images using reversible wavelet transforms," *I. J. Comput. Appl.*, vol. 12, pp. 133–140, 2005.
- [34] J. Spaulding, H. Noda, M. N. Shirazi, and E. Kawaguchi, "BPCS steganography using EZW lossy compressed images," *Pattern Recognition Letters*, vol. 23, pp. 1579-1587, 2002.