INTERNATIONAL JOURNAL OF ELECTRONICS AND COMMUNICATION ENGINEERING & TECHNOLOGY (IJECET)

ISSN 0976 – 6464(Print)
ISSN 0976 – 6472(Online)
Volume 3, Issue 2, July- September (2012), pp. 484 - 496
© IAEME: www.iaeme.com/ijecet.html
Journal Impact Factor (2012): 3.5930 (Calculated by GISI)
www.jifactor.com

IJECET
© I A E M E

AN IMPROVED ROBUST AND SECURED IMAGE STEGANOGRAPHIC SCHEME

Nagham Hamid[1], Abid Yahya[2], and R. Badlishah Ahmad[3], Osamah M. Al-Qershi[4]

[1, 2, 3](Communication and Computer Engineering School, University Malaysia Perlis, Perlis, Malaysia)
[4](School of Electrical and Electronic Engineering, University of Science Malaysia, Penang, Malaysia)

ABSTRACT

Due to the nature of the current digital world, many techniques have become essential for the protection of secret data. The protection of such secret information has led to the development of different kinds of techniques in different categories. Of all of these, steganography has the advantage of concealing vital information in an imperceptible manner. An improved steganographic system is presented in this paper, which successfully embeds secret data within the frequency domain by modifying the Discrete Cosine Transformation (DCT) coefficients. Based on selection criteria, certain blocks are selected for the concealment of data. To ensure a full recovery for the hidden message, an embedding map is proposed to indicate the selected embedding blocks. To secure the embedding map, Speed-Up Robust Features (SURF) is used to dynamically define the locations in which the embedding map is concealed. In addition, the embedding map is hidden in the frequency domain as well by modifying the Discreet Wavelet Transformation (DWT) coefficients in a content-based manner. The obtained results show the robustness of the proposed system against Additive White Gaussian Noise (AWGN) and JPEG compression attacks. Moreover, the resultant stego-images demonstrate good visual quality in terms of Peak Signal-to-Noise Ratio (PSNR). Nevertheless, the hiding capacity which is achieved is still limited due to the fact that only part of the image serves to hide the embedding map.

Keywords: DCT, DWT, Embedding map, Image files, Steganography, SURF.

1. INTRODUCTION

In the modern era, computers and the internet are major communication media that bring the different parts of the world together as a single global virtual world. As a result, people can easily exchange information, while distance is no longer a barrier to communication; however, the safety and security of long-distance communication remains an

issue. This is particularly important in the case of confidential data. The need to solve this problem has led to the development of steganography techniques. Steganography is a powerful (security) tool that provides a high level of security; particularly when it is combined with cryptography [1]. Unlike cryptography, where the main goal is to secure communications from an eavesdropper, steganographic techniques strive to hide the presence of the message itself from an observer. Steganography does not replace cryptography; it rather enhances security using its obscurity features.

Steganography is the art and science of concealing information in an appropriate multimedia carrier, such as, image, audio and video files. It can be supposed that if a feature is visible, the point of attack is evident. Therefore, the goal is to always hide the very existence of the embedded data [2].

Steganography has many useful applications, e.g. in the copyright control of materials, enhancing the robustness of image search engines and in smart IDs (identity cards) where individuals' details are embedded in their photographs. Steganography can be characterized by three factors: undetectability (imperceptibility), robustness, and hiding capacity [2]. It is not possible to maximize robustness, imperceptibility, and capacity simultaneously; therefore, an acceptable balance of these items must be met by the application. When steganography is used as a method for hiding communication, imperceptibility becomes the most important requirement, while robustness and possibly capacity can be sacrificed [4]. A number of ways exist to hide information in digital images. Some of the common approaches include: Least Significant Bit insertion (LSB), Masking and filtering, and Algorithms and transformations. Each of these techniques can be applied, with varying degrees of success, to different image files [5]. For instance, LSB manipulation is a quick and easy way to hide information; however, it is fragile to small changes resulting from either the process of image processing or from the lossy compression. Masking techniques embed information in significant areas so that the hidden message is more integral to the cover image than being a mere hidden message in the "noise" level. Consequently, masking techniques are more robust than LSB insertion with respect to compression, cropping, and to some image processing. Hence, they are more suitable for use in digital watermarking [6].

Other more robust methods of hiding information in images include applications that involve a manipulation of mathematical functions and image transformations. The widely used transformational functions include DCT, Discreet Fourier Transform (DFT), and DWT [7-12]. The basic approach to hiding information with DCT, DFT or DWT involves transforming the cover image, tweaking the coefficients, and then inverting the transformation. If the choice of coefficients is good and the size of the changes manageable, then the result will be very close to the original [7].

Recently, Mali *et al.* proposed a robust DCT-based steganographic scheme via a powerful coding framework that allows the dynamic choice of hiding locations and the embedding of low and medium DCT coefficients [13]. The robustness of this scheme not only comes from exploiting low and medium DCT coefficients for hiding data, but also mainly from the redundancy, whereby the payload bits are repeated n times in order to add robustness to the system. However, the scheme has a severe drawback that results in a loss of information. In this paper, Mali *et al.*'s scheme and its drawbacks will be presented first. Then, a proper modification is proposed to overcome the adopted scheme drawbacks and make it more applicable using the embedding map.

This paper is organized as follows. In section 2 the related work is presented. In section 3, our modification is proposed. The experimental results are presented in section 4. Finally, the discussion and conclusion are given in sections 5 and 6 respectively.

2. RELATED WORK

DCT has been used widely for steganography and watermarking purposes. The DCT-based methods hide data bits in significant areas of the cover-image in order to make them more robust to attacks. Generally, DCT is applied to image blocks of 8×8 pixels, and selected coefficients, of some selected blocks, are used to hide data bits. The coefficients are modified differently in order to reflect an embedding of "0" or "1".

Recently, Mali *et al.* [13] proposed a robust and secure method for embedding a high volume of text information in digital cover-images without leaving perceptual distortion. It has been found that this method is robust in combating intentional or unintentional attacks such as image compression, tampering, resizing, filtering and AWGN. Fig. 1 shows the steganographic data hiding system proposed by Mali *et al.*

Mali *et al.*'s scheme consists of two main stages: processing the data to be embedded and embedding the data. In the first stage, the pure payload bits undergo three processes:

1- Encryption: to secure the data;
2- Redundancy addition: to reduce the bit error rate (BER); and
3- Interleaving: to ensure that the redundant bits are spread all over the image.

It is unnecessary to go through the details of this stage, and the reader can refer to their algorithms for the details. Both redundancy and interleaving are responsible for the recovery of the robust data at the receiver end. Nevertheless, overall robustness also depends on the embedding procedures. For this reason, the embedding procedures and their drawbacks are discussed in the following section.

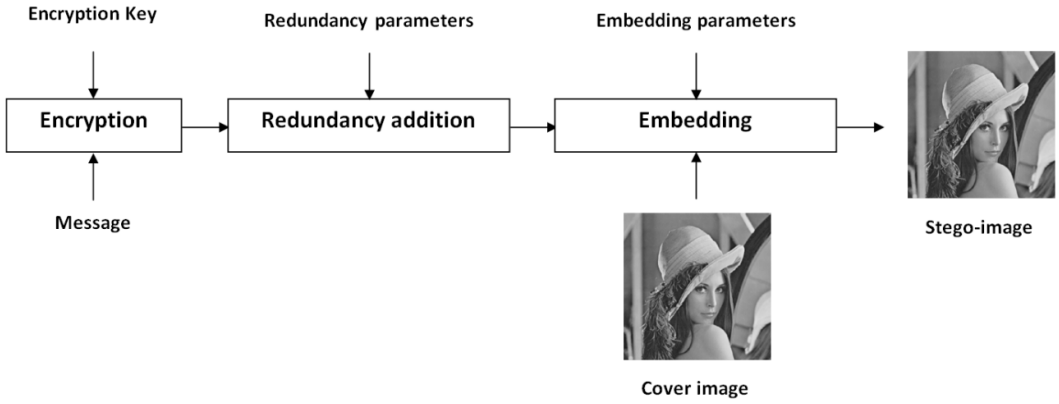

Figure 1: General steganographic system proposed by Mali et al.

2.1 Mali's Embedding Procedures

After processing the data to be embedded, the inputs to the embedding system are a cover-image file (C), the processed text (FBS), Energy Threshold Factor ($\hat{w}$), and JPEG quality factor (QF). The embedding phase can be summarized in the following section. The reader can refer to the original paper for additional details.

Step 1: Divide the image into 8×8 non-overlapping blocks so that DCT is applied to each block a_{ij} to get C_{ij} as:

$$C_{ij} = DCT(a_{ij}) \quad (1)$$

Where $i, j = \{0, 1, 2, \ldots \ldots, 7\}$

Step 2: Calculate the Energy of each block as:

$$E = \sum_{i=1}^{7}\sum_{j=1}^{7}\|C_{ij}\|^2 \quad \forall i, j = \{0, 1, 2, \ldots \ldots \ldots, \quad 7\}, (i, j) \neq 0 \quad (2)$$

Step 3: Calculate the Mean Value of Energy (MVE) of the image using the equation:

$$MVE = \frac{1}{B}\sum_{1}^{B} E_b \quad (3)$$

where B = Total number of blocks and b = block number.

Step 4: Identify the Valid Blocks VBs, which satisfy the Energy Threshold Criteria, $E \geq E_T$, where $E_T = w^{\wedge} \times MVE$.

Step 5: The coefficients of all *VBs* are quantized by dividing them according to their respective elements of the quantization matrix as:

$$C_{ij}^{\wedge} = \frac{C_{ij}}{M_{ij}^{QF}} \quad \forall i, j = \{0, 1, \ldots \ldots, 7\} \quad (4)$$

where, $C_{ij}^{\wedge}$ is the quantized coefficient matrix, M_{ij}^{QF} is the $ij\ th$ element of the quantization matrix for a given value of QF.

Step 6: Identify the Valid DCT Coefficients (*VC*s), which satisfy the non-zero criteria $(C_{ij} \neq 0)$ and which fall into the lower and middle frequency band.

Step 7: The coefficients of all *VC*s are scanned in a zigzag fashion to get the one dimensional vector C_k. The process of embedding data will then be completed by changing the quantized non-zero DCT coefficients, where the odd value for '*bit* = 0' or the even value for '*bit* = 1'. The coefficients with the hidden bits d_k are

given by,

$$d_k^{\wedge} = \begin{cases} Odd\ C_k^{\wedge}, if\ bit = 0 \\ Even\ C_k^{\wedge}, if\ bit = 1 \end{cases} \quad (5)$$

Step 8: The hidden coefficients $d_k^{\wedge}$ are reversely scanned to form an 8 × 8 matrix. It is then multiplied by the JPEG quantization matrix to obtain unquantified coefficients C_{ij}.

Step 9: Apply inverse DCT to each block, and reconstruct the image as stego-image.

From the steps above, it is obvious that the extraction phase depends on identifying the blocks that have been used for the correct embedding. Misidentifying those blocks will cause a loss of portion of the embedded data or extracting unwanted data. During the extracting phase, the blocks that have been used for embedding data should be identified first. Such a process is achieved by the following two steps:

First: The energy of the block (E) should be $\geq E_T$ (step 4).
Second: The lower and middle DCT coefficients of the block should satisfy the non-zero criteria (step 6). If the algorithm fails to identify the blocks in any of the steps, the embedded data is extracted incorrectly.

The algorithm has been implemented and simulated through the use of different images and randomly generated data. The results showed that the algorithm may misidentify the blocks in some cases. For example, in one of the experiments on standard image 'Lena', it was noted that the embedding process changed the MVE and E values. As a result, the embedded block could not be identified during the extraction phase because $E < E_T$. In another situation, reconstructing the stego-image involved a rounding operation to get the integer pixel values. The rounding operation may turn some of non-zero coefficients to zero coefficients or vice versa and also will cause misidentification of the blocks which carry the data; Fig. 2 illustrates the above mentioned case.

71	**68**	**75**	**85**	**75**	**75**	**67**	**81**
71	**67**	**75**	**80**	**73**	**74**	**77**	**88**
77	**78**	**75**	**79**	**81**	**80**	**90**	**100**
78	**74**	**75**	**69**	**82**	**92**	**100**	**112**
80	**75**	**85**	**81**	**88**	**99**	**109**	**121**
80	**81**	**84**	**88**	**100**	**112**	**122**	**133**
77	**81**	**95**	**105**	**113**	**121**	**135**	**141**
81	**91**	**105**	**114**	**122**	**131**	**143**	**144**

MVE = 565,670 and E = 568,185
$w^{\wedge} = 1 \rightarrow E > E_T$
(a)

71	**70**	**83**	**85**	**79**	**80**	**58**	**83**
66	**61**	**75**	**73**	**70**	**77**	**70**	**92**
73	**68**	**72**	**69**	**73**	**81**	**85**	**103**
81	**66**	**76**	**65**	**76**	**94**	**97**	**111**
85	**65**	**89**	**84**	**84**	**104**	**109**	**116**
83	**65**	**86**	**92**	**96**	**119**	**125**	**127**
86	**66**	**97**	**109**	**105**	**126**	**140**	**134**
100	**82**	**111**	**119**	**112**	**135**	**148**	**135**

MVE = 566,780 and E = 560,244
$w^{\wedge} = 1 \rightarrow E < E_T$
(b)

Figure 2: This block is not detected because E <MVE: (a) the original block, (b) after embedding.

Table 1 illustrates the number of misidentified blocks after applying the Mali *et al.* algorithm on image 'Lena' using different quality factor values (QF) and different energy threshold values ($\hat{w}$). Such a problem may affect the integrity of the embedded data; especially, when one cannot identify which blocks have been misidentified. To solve the problem of block misidentification, an embedding map (location map) is proposed in this regard. The concept of the embedding map has been used in many data hiding techniques to correctly identify the location of the blocks or regions where data has been embedded [14-15]. That is to say, not the whole image is used for embedding. In the following section, Mali *et al.*'s algorithm is modified by incorporating an embedding map.

Table 1: Misidentified blocks after applying Mali's algorithm on image 'Lena'

		Number of misidentified blocks			Percentage of misidentified blocks		
		QF = 50%	**QF = 75%**	**QF = 100%**	**QF = 50%**	**QF = 75%**	**QF = 100%**
$\hat{w}$	**0.5**	35	27	51	%1.17	%0.90	%1.71
	0.6	30	28	47	%1.11	%1.03	%1.73
	0.7	35	27	39	%1.39	%1.07	%1.55
	0.8	21	23	44	%0.90	%0.98	%1.88
	0.9	42	14	39	%2.00	%0.67	%1.85
	1	18	21	40	%0.98	%1.15	%2.19

3. THE PROPOSED ALGORITHM

The proposed algorithm is mainly based on that of Mali *et al.* In order to overcome the problem of block misidentification, an embedding map technique is introduced to assure the extraction of the embedded data correctly. Exploiting an embedding map implies generating a binary map of a size equal to the number of blocks in the image. If the image size is $m \times n$, then, the embedding map size is ($\frac{m}{8} \times \frac{n}{8}$), where the block size is 8×8 pixles. Each block in the image is represented by a bit in the embedding map, and if the bit is '1', this means that the corresponding block is used for embedding data and vice versa. The embedding map will be concealed in specific regions of the image, while the data will be concealed in different regions. The regions, *i.e.* the blocks, in which the data is hidden are determined according to Mali *et al.*'s method. Although the embedding map can be embedded in some predefined regions selected by the user, this option may affect the data

security because the same regions are used every time. Therefore, it is preferable to adopt a more secure method to hide the embedding map, as described in the next section.

3.1 Hiding the Embedding Map

The embedding map is necessary to initiate the extracting phase, which means that it must be concealed in the image in an extremely secure and robust way. To achieve these objectives, two powerful techniques are used. The first technique is used to guarantee the security of the embedding map and is performed by selecting the regions to be embedded in a dynamic way, depending on the key-points of the image. For this purpose, Speed-Up Robust Features (SURF) is used to extract the distinctive local features in the image and to produce the key-point descriptors that demonstrate those features. Feature vectors/descriptors are invariant to rotation, translation, and scaling; they are further partially invariant to illumination changes and are robust to local geometric distortion [16]. Each feature vector has some information that describes its corresponding key-point. The important part of the information for the algorithm is the coordinates of the center of the key-point and its scale. Twelve non-overlapping key-points with the highest scales are selected and used to hide the embedding map. These key-points are the most robust features in the image and can be detected even when the stego-image undergoes different types of operations, such as JPEG compression and Gaussian noise. As to the second requirement, robustness, a DWT-based embedding technique is adopted in which the data is embedded in a content-based manner. For more details on using SURF and content-based embedding, the reader can refer to [17, 18].

3.2 Embedding Phase

Having introduced the embedding map, the proposed algorithm can be described through the following steps:

Step 1: The SUR is applied to the image, and 12 key-points with the highest scales are identified. The center coordinates of the key-point defines 12 square regions of 32×32 sized pixels with the same center coordinates. These regions are used to hide the embedding map.

Step 2: The image is divided into 8×8 non-overlapping blocks so that DCT is applied to each block a_{ij} to get C_{ij}, as in Equation (1). Notice that the blocks that intersect with the 12 square regions that have been obtained in Step 1 are discarded as those regions are used to hide the map not the data.

Step 3: The blocks are scanned, and the blocks which can be used for embedding are found (according to Mali *et al.*'s algorithm), as described in section 2.1.

Step 4: An embedding map is built to indicate the blocks in which the data bits will be embedded.

Step 5: The secret data is embedded in the blocks defined in Step 3 by modifying the DCT coefficients according to Mali *et al.*'s algorithm.

Step 6: The embedding map is embedded in the 12 square regions defined in Step 1 by modifying the DWT coefficients (in a content-based manner), as described in section 3.1. Notice that addition of the redundant bits and interleaving techniques have been used to prepare both data and the embedding map to be hidden. Fig. 3 illustrates the process of embedding data.

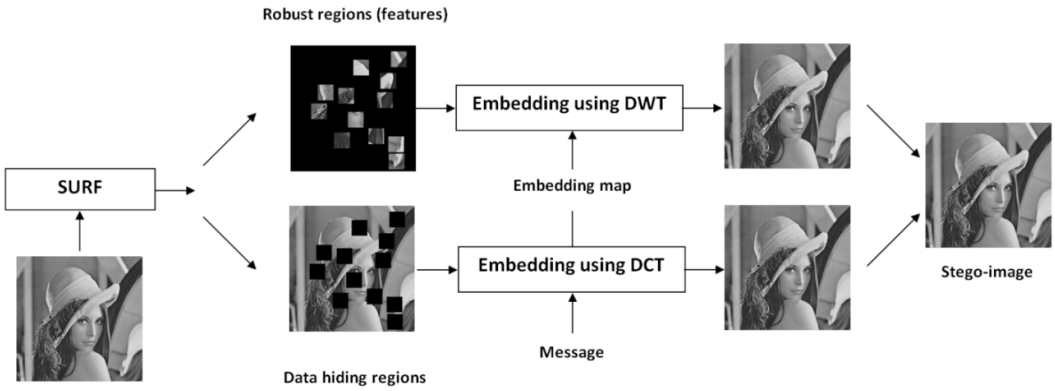

Figure 3: The process of embedding data using the proposed algorithm

4. EXPERIMENTAL RESULTS

To assess the performance of the proposed algorithm, experiments were performed on three standard grayscale images; 'Lena', 'Boat', 'and 'Gold hill', with the size 512×512, as shown in Fig. 4. In order to evaluate the reliability of the proposed algorithm compared to Mali *et al.*'s algorithm, different levels of attack were applied on the stego-image. The attacks involved were JPEG compression and AWGN. A comparison between the proposed algorithm and Mali *et al.*'s algorithm in terms of robustness is presented in Tables 2 and 3.

It is worth mentioning that the number of missed blocks can be calculated by subtracting the number of the detected blocks in the receiving end from the blocks that have

been used for secret data embedding (the actual embedding blocks). Accordingly, when the number of the detected blocks is less than the actual number of the embedding blocks, the result will be a positive number. On the other hand, due to the applied attacks to the stego-image, erroneous blocks may be detected. As a result, the number of the detected blocks may be more than the actual number of embedding blocks. This is the reason behind having numbers with negative values representing the misidentified blocks, as shown in Tables 2 and 3.

Table 2: A comparison between the proposed algorithm and Mali's algorithm in terms of reliability (Number of misidentified blocks)**.** (QF = 75% & $\hat{w}$ = 0.5)

	Number of misidentified blocks					
	The proposed algorithm			Mali's Algorithm		
	Lena	Boat	Gold hill	Lena	Boat	Gold hill
No attack	0	0	0	27	15	17
JPEG 100%	0	0	0	-14	-9	-5
JPEG 80%	0	0	0	316	249	228
Gaussian Noise 45dB	0	0	0	-26	-12	-5
Gaussian Noise 35dB	0	0	0	-32	-12	2

Table 3: A comparison between the proposed algorithm and Mali's algorithm in terms of reliability (Number of misidentified blocks). (QF = 75% & $\hat{w}$ = 0.8)

	Number of misidentified blocks					
	The proposed algorithm			Mali's Algorithm		
	Lena	Boat	Gold hill	Lena	Boat	Gold hill
No attack	0	0	0	23	14	32
JPEG 100%	0	0	0	-12	-3	6
JPEG 80%	0	0	0	250	247	175
Gaussian Noise 45dB	0	0	0	-31	-12	10
Gaussian Noise 35dB	0	16	0	-28	-20	2

In terms of imperceptibility, the visual quality of the obtained stego- image with the proposed algorithm is better compared to those obtained by Mali *et al.*'s algorithm as illustrated in Table 4. The visual quality is measured by the PSNR, as given in (6)

$$PSNR(I, I_S) = 10\, log_{10} \frac{MAX_I{}^2}{\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\|I(i,j) - I_{S(i,j)}\right\|^2} \quad (6)$$

Where I is the original image; I_S is the stego-image; MAX_I is the maximum possible pixel value of the image I. The results obtained demonstrated that the proposed algorithm has a better visual quality. For the purpose of evaluation, another comparison was made in terms of the hiding capacity. The capacity is measured as the number of payload bits that can be embedded in the image and retrieved successfully. The obtained results are shown in Table 5.

'Lena'

'Boat'

'Gold hill'

Figure 4: The images used for evaluation

Table 4: A comparison between the proposed algorithm and Mali's algorithm in terms of PSNR

		PSNR values (dB) @ QF = 75%					
		The proposed algorithm			Mali's Algorithm		
		Lena	Boat	Gold hill	Lena	Boat	Gold hill
$\hat{w}$	**0.5**	37.28	34.10	36.42	33.19	32.66	33.36
	0.6	37.62	34.24	36.98	33.56	32.78	33.82
	0.7	38.33	34.67	37.61	33.91	32.94	34.28
	0.8	38.37	35.10	38.51	34.22	33.18	34.89
	0.9	38.79	35.74	39.18	34.69	33.53	35.58
	1	39.74	36.76	39.53	35.31	34.16	36.57

Table 5: A comparison between the proposed algorithm and Mali's algorithm in terms of hiding capacity

		Hiding Capacity (bits)					
		The proposed algorithm			Mali's Algorithm		
		Lena	Boat	Gold hill	Lena	Boat	Gold hill
$\hat{w}$	**0.5**	25,536	46,144	30,016	83,664	93,996	80,304
	0.6	22,400	44,352	25,536	75,936	91,196	72,072
	0.7	19,264	38,976	21,056	70,476	87,892	64,344
	0.8	18,816	33,152	16,128	65,688	83,272	56,196
	0.9	15,680	25,088	12,096	58,884	76,580	47,796
	1	11,648	16,128	10,752	51,212	66,724	38,444

5. DISCUSSION

In this paper, an interesting algorithm proposed by Mali *et al.* was reviewed. Their algorithm shows adequate levels of robustness due to combining DCT and adding redundancy bits. However, some of the blocks that carry data might be misidentified during the extraction process. The present paper aims at enhancing the reliability of the original algorithm by overcoming the problem of the misidentified blocks. To do so, an embedding map has been adopted to indicate the location of the blocks, which have been used for embedding. This means that some regions of the image will be exploited to hide data, while others will be used to hide the embedding map. The blocks in which the data is concealed are determined according to Mali *et al.*'s algorithm. The regions in which the embedding map is concealed are determined in an extremely dynamic way to increase the security of the algorithm. This goal has been achieved by the use of the SURF technique, which is used to find the robust key-points of the image, due to the fact that each image has different key-points. The embedding map is incorporated in those regions using a DWT-based method.

The experimental results in Tables 2 and 3 show that the proposed algorithm can overcome the problem of block misidentification, even when the stego-image undergoes JPEG compression or Gaussian noise. The high reliability of the algorithm in identifying the blocks comes from the ability of SURF to detect the key-points, even after applying the attacks. The DWT-based embedding technique also plays an important role in keeping the embedding map intact. Nevertheless, strong attacks, such Gaussian noise may cause the blocks to be slightly misidentified, as is the case in the image of 'Boat' in Table 4.

In comparison to Mali *et al.*'s algorithm, the proposed algorithm shows a better visual quality in terms of PSNR, as shown in Table 4. This can be said for the three images used for testing and with all ($w^{\wedge}$) values used. However, Table 5 indicates that the hiding capacity of the proposed scheme is somewhat lower than that achieved by Mali *et al.*'s algorithm. The reason behind the capacity reduction is the exploitation of some regions of the image to hide the embedding map.

The reason for the robustness of the algorithm in terms of Bit Error Rate (BER is not tackled in this paper) is that the same embedding technique used in Mali *et al.*'s algorithm for hiding the data has been used in the present study, which means that both algorithms have the same level of robustness. The aim was also to enhance the reliability of Mali's algorithm. (Increasing the robustness is out of the scope of this paper).

6. CONCLUSIONS

The original algorithm presented by Mali *et al*. gives very good results in terms of robustness; however, the algorithm cannot be considered reliable as some data may be lost. In order to overcome the problem of lost data due to misidentified blocks, an embedding map was used to specify the location of the blocks which were used for embedding secret message, as the embedding map is very important to start the extracting phase correctly and accurately. Any loss in the embedding map will in turn lead to data loss. Consequently, the embedding map is hidden using SURF and DWT to assure robustness and to increase the security level of the proposed system. The secret data can be embedded using the original algorithm (DCT-based) proposed by Mali *et al*. While the experimental results show the ability of the new algorithm to overcome the problem of lost data even with JPEG compression or Gaussian noise, exploiting the embedding map reduces the available hiding capacity. To increase this capacity, more embedding techniques with high hiding capacity should be considered to hide the embedding map as this will permit more data (messages) to be embedded in the image and retrieved effectively. Moreover, more possible attacks should be investigated.

REFERENCES

[1] S.A.Halim and M.F.A.Sani, Embedding using spread spectrum image steganography with GF (2^m), Proc. IMT-GT-ICMSA, Kuala Lumpur, Malaysia, 2010, 659-666.
[2] A. Cheddad, Steganoflage: A new image steganography algorithm, doctoral diss., University of Ulster, Northern Ireland, UK, 2009.
[3] A. Cheddad, J. Condell, P.M. Kevitt, Digital image steganography: Survey and analysis of current methods, Signal Processing Journal, 90(3), 2010, 727-752.
[4] B, Li, J. Huang, and Q.Y. Shi, A Survey on image steganography and steganalysis, Journal of Information Hiding and Multimedia Signal Processing, 2(2), 2011, 123-138.
[5] N.F. Johnson and S. Jajodia, Exploring steganography: seeing the unseen, IEEE Computer Journal, 31(2), 1998, 26-34.
[6] K. Curran and K. Baily, An evaluation of image based steganography methods, International Journal of Digital Evidence, 2(2), 2003, 1-40.
[7] A. Nag, S. Biswas, D. Sarkar, and P.P. Sarkar, A novel technique for image steganography based on Block-DCT and Huffman Encoding, International Journal of Computer Science and Information Technology, 4(6), 2010, 103-112.
[8] C.C. Chang, T.S. Chen, and L.Z. Chung, A steganographic method based upon JPEG and quantization table modification, Information Sciences Journal, 2002, 141(1,2), 123-138.
[9] M. Ashourian, R.C. Jain, and Y.H. Ho, Dithered quantization for image data hiding in the DCT domain, Proc. of IST2003, 2003, 171-175.
[10] M. Iwata, K. Miyake, A. Shiozaki, Digital steganography utilizing features of JPEG images, IEICE Trans. Fundamentals, E87-A, 2004, 929–936.
[11] C.C. Chang, C.C. Lin, C.S. Tseng, and W.L. Tai, Reversible hiding in DCT-based compressed images, Information Sciences Journal, 177(13), 2007, 2768-2786.
[12] C.C. Lin, P.F. Shiu, High capacity data hiding scheme for DCT-based images. Journal of Information Hiding and Multimedia Signal Processing, 1(3), 2010, 123-138.
[13] S.N. Mali, P.M. Patil, and R.M. Jalnekar, Robust and secured image-adaptive data hiding, Digital Signal Processing, 22(2), 2010, 314–323.
[14] J. Tian, Reversible data embedding using a difference expansion, IEEE Transactions on Circuits and Systems for Video Technology, 13(8), 2003, 890-896.
[15] A.M. Alattar, Reversible watermark using the difference expansion of a generalized integer transforms, IEEE Transactions on Image Processing, 13(8), 2004, 1147-1156.
[16] H. Bay, From wide-baseline point and line correspondences to 3D, doctoral diss., Swiss Federal Institute of Technology, Switzerland, 2006.
[17] N. Hamid, A. Yahya, R.B. Ahmad, and O.M. Al-Qershi, Characteristic region based image steganography using Speeded-Up Robust Features technique. Proc. of IEEE International Conference on Future Communication Networks (ICFCN 2012), Iraq, 2012, 141-146.
[18] L. Li, J. Qian, and J.S. Pan, Characteristic region based watermark embedding with RST invariance and high capacity. AEU-International Journal of Electronics and Communications, 65(1), 2011, 435-442.