International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:12, No:2, 2018

Improved Processing Speed for Text Watermarking Algorithm in Color Images

Hamza A. Al-Sewadi, Akram N. A. Aldakari

***Abstract*—**Copyright protection and ownership proof of digital multimedia are achieved nowadays by digital watermarking techniques. A text watermarking algorithm for protecting the property rights and ownership judgment of color images is proposed in this paper. Embedding is achieved by inserting texts elements randomly into the color image as noise. The YIQ image processing model is found to be faster than other image processing methods, and hence, it is adopted for the embedding process. An optional choice of encrypting the text watermark before embedding is also suggested (in case required by some applications), where, the text can is encrypted using any enciphering technique adding more difficulty to hackers. Experiments resulted in embedding speed improvement of more than double the speed of other considered systems (such as least significant bit method, and separate color code methods), and a fairly acceptable level of peak signal to noise ratio (PSNR) with low mean square error values for watermarking purposes.

***Keywords*—**Steganography, watermarking, private keys, time complexity measurements.

I. Introduction

IN the digital world, protecting digital media files during storage or transit from being leaked, modified, misused, or stolen and claimed by others is a crucial matter nowadays. Hence, information security has become more of an issue of preserving data and protecting its copyrights, ownership, and validity, as well as keeping the secrets not being unveiled to unauthorized persons. Information security can be classified into two types; Cryptography and Data Hiding [1]. Cryptography involves converting intelligible media into an unintelligible one using certain procedures and secret keys, whereas data hiding means embedding data into certain multimedia without causing perceptual degradation or any noticeable effect [2], [3]. Therefore, cryptography serves the purpose of protecting secret data or information from being leaked, tampered with or modified during storage or transit over the communication channel, while, data hiding comes in two types; steganography by hiding secret messages into multimedia, and watermarking that serves copyright protection and ownership judgment. Watermarks can be either visible or invisible [4].

In general, data hiding techniques are classified into either spatial (time) domain or transformational (frequency) domain. In spatial domain, embedding is achieved by modifying the multimedia bits directly, while in frequency domain, embedding is done into a transformed form of the multimedia. An example of spatial domain is the least significant bit (LSB). Examples of the transformational domain are the discrete wavelet transform (DWT), discrete Fourier transform (DFT), and discrete cosine transform [5]. LSB is a simple and fast watermarking technique; hence, it is found to be suitable for applications that require fast embedding and extraction of text or signature watermarks into images.

An improved speed is achieved by the proposed scheme in this paper for embedding and extraction of text watermarks into still images. This modified schema relies on a random number generator for the embedding and extracting processes of text watermarks into images that works faster than the LSB technique. It relies on pixel replacement presented as noise rather than bits replacement. Moreover, an encryption stage of the text watermark is suggested before embedding with the intention of thwarting the original owner of the watermarked image.

After this brief introduction in Section I, Section II will list the most related work. Relevant image processing methods will be outlined in Section III, and then Section IV outlines the proposed watermarking scheme. Section V includes implementation, results and discussion. Finally, the conclusion is included in Section VI.

II. Related Works

A number of related works will be examined in this section. Since this research is concerned with text watermarking, the following literature survey describes only the previous work done on digital watermarking in a spatial domain on text watermarking in particular.

Digital data hiding depending on the psycho visual repetition in digital images of grey scale were examined by Hossain et al. [6], using neighborhood information. Their importance arises from the ability to know exactly the data included within the image input pixel, without making differences. The neighborhood connection sheds light on the smooth parts of an image that reflects the limited amount of unrevealed data, in addition to the complex parts of an image represented in the big amounts of hidden data. Smooth areas are less resistant to modification than edge parts. The greater number of image pixels were not contingent to the hidden information, as only three bits were approved to be kept secret in smooth areas, and other changing bit numbers were maintained concealed in the edge areas.

Al-Dwairi et al. [7] developed a method based on the direct and inverse color image alteration. They achieved an advanced real color image alternation, through the reduction of

Hamza A. Al-Sewadi and Akram N. A. Aldakari, MSc student, are with the Computer Department, Faculty of Information Technology, Middle East University, Jordan-Amman (e-mail: hsewadi@meu.edu.jo, akramaldgaree77@gmail.com).

necessary time to manage inverse alteration using R'G'I design instead of HSI design. Also in 2010, Bamatraf et al. [8] presented a simple and robust watermarking algorithm using grayscale image, and concerning the concealment of data. They implement the third LSB and the fourth LSB for the digital watermark pattern, and claim that it is more robust than the traditional LSB technique.

A hash based LSB technique in spatial domain is reported by Dasgupta et al. [9]. It is a utilization of an algorithm portrayed with audio video interleave (AVI) file as a cover medium. A video stream composed of collection of frames and the secret data is concealed in these frames as payload. This technique conceals 8 bits of embedded data at a time in LSB of RGB pixel value of the carrier frames in 3, 3, 2 order, respectively.

In 2014, Sruthi et al. [10] proposed a hybrid watermarking method that involves LSB and DCT. It showed some robustness against Gaussian and speckle noise.

A novel dual purpose spatial domain algorithm for digital image watermarking and cryptography using Extended Hamming Code is reported Ghosh et al. [11]. It achieved more advanced and valid level of imperceptibility through the exploitation of the couple purpose robust algorithm, concerning image cryptography and digital watermarking.

In 2016, Kumar and Dutta [12] presented a new technique for image watermarking in the spatial domain utilizing the concept of information theory with LSB algorithm. The host image is segregated into blocks and the watermark is embedded into the block(s) with the maximum entropy value, claiming a verified perceptibility and robustness over a variety of host and watermark images. At the same time, Mathur et al. presented a spatial domain based image watermarking using shell based pixel selection. The importance of this algorithm arises from its ability to present an advanced degree of security, as well as, draw out the watermark. What distinguishes the algorithm is also its ability to make watermark locations unpredictable using pixel selection, which form the basis of the shell.

The research in this paper concentrates on gaining good performance for text watermarking into still images. This will be achieved by suggesting an algorithm that depends on the random determination of host image locations and performing the embedding of the text watermark in these locations as noise rather than bits in the pixels, as in the case of LSB. The proposed algorithm will be tested and compared with the traditional LSB technique for embedding and extraction speed of text messages.

III. Image Processing Methods

Three types of color processing methods are considered and studied in order to select the fastest one to be adopted in the proposed embedding algorithm; Separate color channels, Direct conversion to gray color, and YIQ Model.

A. Separate Color Channels (SCC)

The RGB color image components are separated into three channels (R, G, and B). They are treated separately, then recombined together in order to produce the resulting processed color image, as illustrated in Fig. 1 [12], [13].

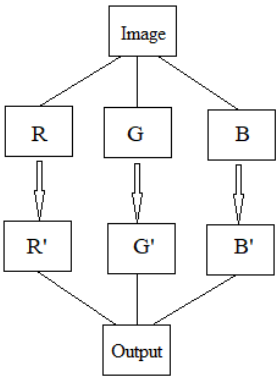


Fig. 1 SCC [13]

B. Direct Conversion to Gray Color (DC)

The RGB color image is converted directly to gray image first, then the gray image is handled for any purpose such as embedding the secret information, logos or signatures, and then converted back to a color image. The conversion to grey color in this method involves three components; two of them for chrominance (Rd and Gd) and one for intensity I, [14], as given by (1).

$$\begin{aligned} Rd &= (R * 256)/(R + G + B) \\ Gd &= (G * 256)/(R + G + B) \\ I &= (R + G + B)/3 \end{aligned} \quad (1)$$

After the embedding process, they are converted back to RGB, as given by (2).

$$\begin{aligned} RI &= (3 * Rd * I)/256 \\ GI &= (3 * Gd * I)/256 \\ BI &= (3 * (256 - Rd - Gd) * I)/256 \end{aligned} \quad (2)$$

C. YIQ Model

Deferent methods are now available to convert color image to gray and vice versa such as YUV and YIQ [7], where color information is separated from brightness information. For YIQ model, Y is the luminance component, while I and Q hold the color information (chrominance). Many advantages of the YIQ model over RGB can be noticed, such as the brightness information Y is separated from the color information, the correlations between the color components are reduced, and most of the information is collected to the Y component, while the information content in I and Q is much less. This work considers YIQ; hence, the conversion to grey color follows (3).

$$\begin{aligned} Y &= 0.299 * R + 0.587 * G + 0.114 * B \\ I &= 0.596 * R - 0.275 * G - 0.321 * B \\ Q &= 0.212 * R - 0.523 * G + 0.311 * B \end{aligned} \quad (3)$$

The process of converting an RGB color image to the YIQ model and back to RGB is shown diagrammatically in Fig. 2. Therefore, embedding can be done after the conversion to gray, then converted back to a color image.

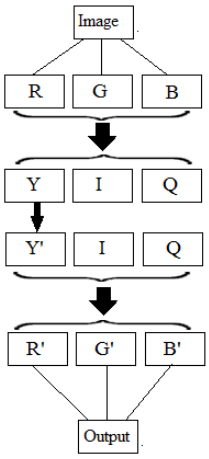


Fig. 2 Conversion between RGB and YIQ [15]

Experimentation with these three methods has shown that the YIQ model requires a minimum time of conversion as compared with others, and hence, it is used in the proposed methodology of data hiding.

IV. The Text Watermarking Scheme

In this section, the embedding and extraction algorithms for the proposed text watermarking scheme into color images are outlined. Initially, some important terminologies are defined as follows:

A. Definitions

- *Secrete message*: the messages text content that is to be embedded as a watermark into a color image, and then retrieved when needed. This message is the proof of copyright or ownership of the original color image.
- *Private Key*: A randomly generated vector matrix, having the size of the secret message, is used as the key to locate the embedding positions.
- *Embedding time*: The time taken to embed the message into the color image.
- *Extracting time*: The required time to extract the watermark from color image.
- *Speed gain*: An efficiency measure for the embedding or extraction speed.

The watermarking process is performed into the host images as noise insertion after they have been converted to grey scale in the YIQ format; then, the watermarked images are converted back to RGB. The YIQ format is chosen for the embedding process as it is found to give the shorter processing time in comparison with the SCC and the direct conversion method, DC, as will be seen in the next section. The embedding of the water mark as noise into the images is adopted due to the fact that text watermarks are usually short messages, and hence, their existence in color images, which are normally much greater in size, is tolerated and hardly noticed. To increase the owner authenticity, an extra stage may be added to the watermarking by encrypting the text watermark using any encryption technique.

B. Embedding Algorithm

The embedding process of the text watermark into color images is illustrated in the flow chart shown in Fig. 3 and achieved by the following steps.

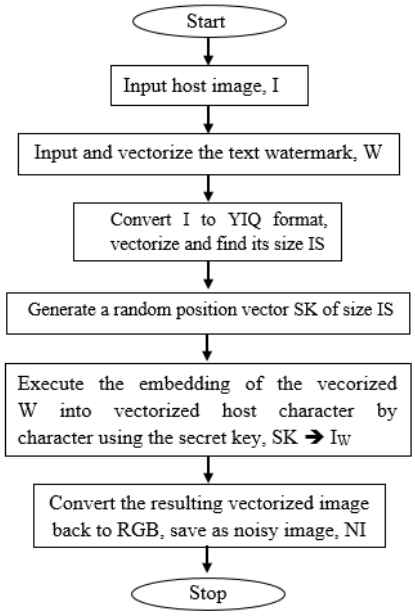


Fig. 3 Flow chart of the embedding process

Step 1. *Load host image*: Input the original host image I, which is to be watermarked. This image can be gray or color. In the experiment of this paper, large sizes and high resolution RGB color images are used. Consider an RGB image of ***r*c*** size, where ***r*** is the number of rows and ***c*** is the number of columns, hence its pixels are represented by three ***r*c*** arrays, each of 8-bits length, i.e. one array for each color component, Red, Green, and Blue. Therefore, the total size (in bits) of the image array is given by (4).

$$IS = 3 * r * c \quad (4)$$

Step 2. Load the watermark: Input the watermark data W, which is a text data of any number of characters, put them in ASCII code form, and then arrange them as a vector, whose elements are the binary representations of the characters.

Step 3. *Process the host image*: The RGB image array values obtained in step 1, which consists of three matrices, each having a size of ***r * c*** pixels, are converted into three matrices in the YIQ format using (3). These three matrices are vectorized, i.e. rearranged into one dimensional array having a size of 3* ***r * c*** elements, ready to be used for watermark embedding.

Step 4. *Key generation*: Randomly generate a vector of size equal to the number of characters in the text watermark (KS) and values in the range 1 to IS, to be used as the embedding key vector (secret key, SK), and must be privately saved to be used for the extraction process.
Step 5. *Embedding*: Use the generated key vector to successively locate and replace elements of the host image vector by the elements of the watermark vector.
Step 6. Rearrange the resulting watermarked image vector in order to get the in YIQ format, and then convert it back as the watermarked RGB image I_W, which is referred to here as the noisy image, NI. The proposed embedding process of text or signature watermark hiding is highly secure due to the difficulty of guessing the key by unauthorized user, as will be shown later.
Step 7. *Optional*: To increase the difficulty for hackers or thieves, and thwarting them from knowing the original owner of the protected watermarked image, an encryption stage can be added before the insertion step. The text watermark must be encrypted using any encryption technique before it is inserted into the image. In this work, a matrix manipulation encryption technique is adopted, which is analogous to the Hill cipher technique, to be used optionally if watermark encryption is required.

C. Extraction

To detect the hidden watermark in the watermarked image, the watermarking extraction process is performed. It is exactly the inverse of the watermark embedding process. The extraction algorithm starts by accepting the watermarked image NI, converts it to YIQ format, vectorizes it, and then uses the same key vector (SK) that is randomly generated and used for the embedding process to retrieve the watermark text from the vectorized image watermark. The extraction process is briefly illustrated in Fig. 4.

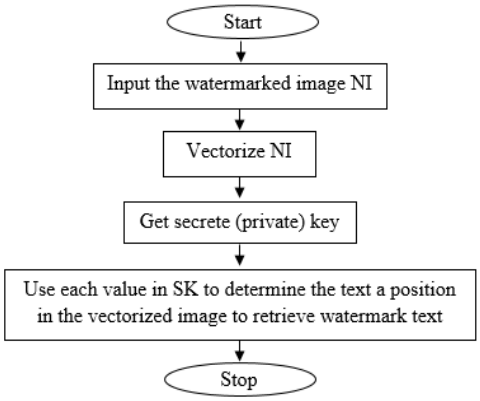


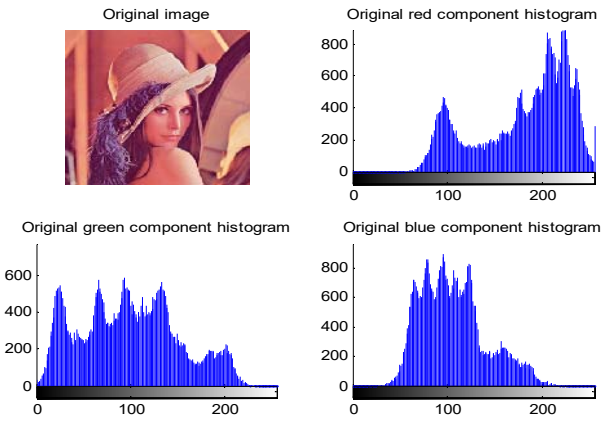
Fig. 4 Flow chart of the Extraction process

V. Implementation and Discussion

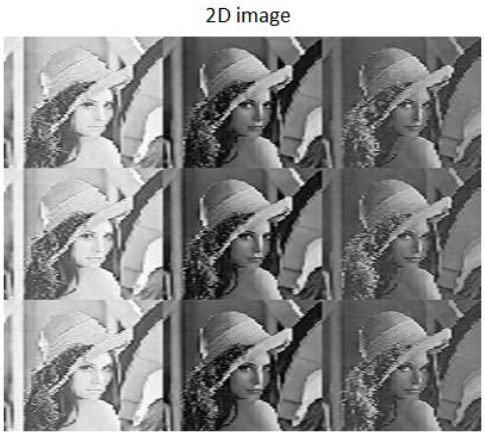
Implementation and testing of the proposed algorithm for text watermarking into color images is done using MATLAB codes, which is run on an i7 PC with 4G bytes RAM. It starts with the testing of three image processing types, namely SCC, DC, and YIQ methods. Then, the testing of embedding and extraction processes for processing speed (or processing time complexity) and the effects of various kinds of noise and other external disturbances such as compression, resizing, rotation, skewing, cropping, etc.

A. The Image Processing Methods

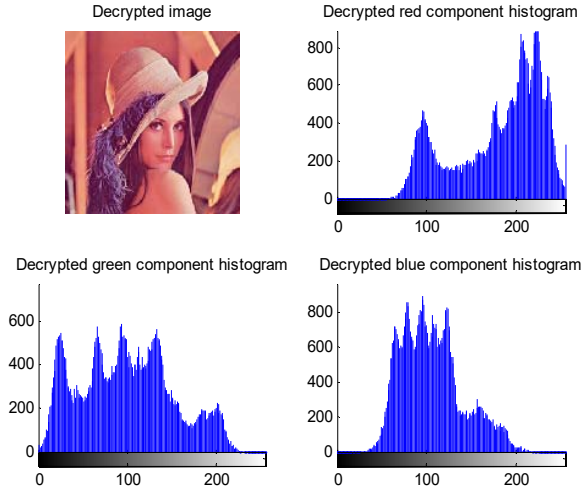
Three image processing methods described in section 2.6.2, namely; the SCC, the direct conversion to gray, and the YIQ model, are tested for processing speed. This is done in order to find out and choose the most suitable or the fastest method to be adopted in the proposed algorithm. For each of these methods, a color image is taken in, separate its RGB components, determine and plot the histogram for each color component, and encrypt the image using the matrix manipulation technique. Then it is decrypted and plotted together with the histograms for its three R, G, and B color components. Hundreds of images are processed in this experiment; however, as a representative example, Lena color image is listed in Fig. 5. It shows the image and its components histograms before and after the encryption/ decryption processes.



(a)



(b)



(c)

Fig. 5 YIQ model (a) original RBG (b) YIQ-2D Image (c) RGB after processes

The consumed times of encryption and decryption for the three processing methods are calculated and compared with each other in order to select the fastest image processing method for the proposed scheme. A collection of 900 color images of various objects, sizes, color combinations and definitions were selected and processed with the three image processing methods described in Section III. The encryption and decryption processing times are measured (in seconds) for all the images, their averages are calculated and listed in Table I together with total average processing time.

TABLE I
AVERAGE CALCULATED TIME FOR THE THREE PROCESSING METHODS

Processing method	Average processing time (seconds) Encryption	Decryption	Total
SCC	0.3841	0.3180	0.7021
DC	0.6495	0.2439	0.8934
YIQ	0.1080	0.1980	0.3060

Table I shows that the image encryption/decryption processing time for the image processing using the YIQ model is much shorter as compared with the SCC model and the direct conversion to grey image processing (DC) methods. Therefore, the YIQ model for color image processing is chosen to be used in the suggested text watermark embedding into color images as it is generally more than twice faster than any of the two other methods (i.e. total time for YIQ model ≈ 2.3 and ≈ 2.9 times faster than SCC and DC, respectively). However, the encryption speed is found to much faster for YIQ method compared with the other method, namely it is ≈ 3.6 and ≈ 6 times faster than SCC and DC, respectively.

B. Embedding Tests

The hiding technique is tested for embedding different watermark text size together with variation in image size. Watermarks text of lengths equal 100, 250, 750, 1250, and 2000 characters, were used for testing the algorithm into hundreds of color images with different sizes. However, only three images were listed here as representative examples, namely; Jarash, Einstein, and Lena profiles with sizes 800*469, 337*268, and 225*225 pixels, respectively. The presented results include embedding imperceptibility, embedding and extraction processing time, and the peak signal to noise ratio (PSNR) comparison.

The images before and after embedding of the text watermark using text length of 2000 bytes into Jarash, Einstein, and Lena profiles when the proposed watermarking algorithm is implemented is shown in Fig. 6, from which it can be seen clearly that these image are visually imperceptible.

Fig. 6 Embedding of text length of 2000 into images

The embedding and extraction processing time and the PSNR for executing the proposed algorithm are computed for watermarking texts of different character combinations and different lengths embedded into various color images of various sceneries and sizes, too. Table II lists the processing time measurement for 2000 characters text watermark hiding into the same images used previously, i.e. Jarash, Einstein, and Lena images using the proposed algorithm as compared with the use of the DC to grey method (i.e. LSB) and the SCC method.

Table III and Fig. 7 show that the PSNR for the proposed algorithm is in the same range of that for the SCC algorithm, but it is less than that of the LSB algorithm, besides the values lower as the embedded text size increases.

This can be explained as being attributed to the fact that the proposed method depends upon the replacement of the whole pixel in the selected place, while it is the change of only the LSB of the selected pixels in the case of the LSB method.

However, PSNR values are still over 70 dB for text lengths of 2000 characters which is normally acceptable. In other words, the proposed method introduces more noise into the host images, but as watermarks are usually chosen small in size, therefore the proposed algorithm is more suitable for text watermarks, rather than for steganography long messages.

TABLE II
EMBEDDING AND EXTRACTION EXECUTION TIME COMPARISON (SECONDS)

Name and size of the image	SCC method		DC method (LSB)		YIQ method (Proposed)	
	Emb.	Ext.	Emb.	Exe.	Emb.	Exe.
Jarash 800*469	0.0466	0.9557	1.8103	0.6730	0.0157	0.0042
Einstein 337*268	0.0720	0.0140	0.6838	0.5903	0.0215	0.0032
Lena 225*225	0.0469	0.0631	0.4556	0.4697	0.0028	0.0028

Emb. - Embedding and Exe. - Extraction

TABLE III
PSNR VALUES COMPARISON

Image name and size	Text size byte	SCC method	LSB algorithm	Proposed algorithm
Jarash 800*469	100	128.1517	156.2086	122.4811
	250	114.4798	145.6573	113.0975
	750	101.5126	133.8458	101.3833
	1250	96.0315	128.5472	95.7979
	2000	91.0393	123.7436	91.0756
Einstein 337*268	100	106.0534	139.9674	105.6661
	250	95.3762	130.7687	95.5660
	750	86.7080	120.0071	85.0605
	1250	79.8441	115.4919	79.5655
	2000	76.7915	114.3283	74.8961
Lena 225*225	100	106.0534	139.9674	105.6661
	250	95.3762	130.7687	95.5660
	750	86.7080	120.0071	85.0605
	1250	79.8441	115.4919	79.5655
	2000	76.7915	114.3283	74.8961

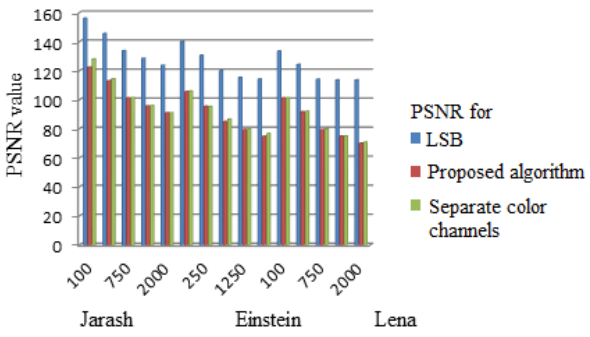


Fig. 7 PSNR values histogram comparison

VI. CONCLUSIONS

The suggested algorithm in this paper produces watermarks that are imperceptible by visual inspection, and was implemented and tested for embedding/extraction processing speed improvement. In this algorithm, the text characters embedding is based on spatial domain embedding technique as noise insertion in the host image to be protected, relying on a randomly generated secret key of variable length that is related to the text length or the message size.

The processing speed of the YIQ model is found to be faster than both direct conversion to grey and separated color code methods, which makes it lend itself for efficient watermarking implementation.

The obtained value of PSNR were over 70 dB for a text watermark of 2000 characters which can be considered good enough as compared with other techniques.

The proposed algorithm showed an encouraging improvement in having comparatively fast embedding and extraction processes speed as compared with other spatial watermarking techniques, as the embedding and extraction time was much shorter compared with the LSB algorithm and the separated color channels method. This is due to the fact that the algorithm relies on the random insertion of the text as noise into the images rather than bits into pixels.

The algorithm suggested an option of encrypting the watermark, which although will increase the execution time somehow, it would be useful for applications that require watermark authentication. Therefore, it can be concluded that the proposed text watermarking algorithm is suitable for short text messages embedding into color image for the purpose of copyright protection and ownership judgment applications, rather than for steganography application, i.e. not for encrypting and decrypting long messages for secure data storage or messages transfer over the internet.

REFERENCES

[1] A. Raphael, and V. Sundaram, "Cryptography and Steganography – A Survey," International Journal of Computer Technology Applications, vol. 2, 2011, PP 626-630.

[2] R. M. Patel, and D. J. Shah, "Concealogram: Digital image in image using LSB insertion method", International journal of electronics and communication engineering & technology (IJECET), 2013.

[3] N. Akhtar, J. Pragati, and S. Khan, "Enhancing the security and quality of LSB based image steganography", *in the 5th International Conference on Computational Intelligence and Communication Networks*, 2013.

[4] M. Juneja, and P. S. Sandhu, "An improved LSB based Steganography with enhanced Security and Embedding/Extraction", *3rd International Conference on Intelligent Computational Systems (ICICS'2013)* Jan 26-27, Hong Kong (China), 2013.

[5] S. M. Karim, M. S. Rahman, and M. I. Hossain, "A New Approach for LSB Based Image Steganography using Secret Key", *Proceedings of 14th International Conference on Computer and Information Technology (ICCIT 2011), 22-24 Dec 2011*, Dhaka, Bangladesh.

[6] M. Hossain, S. Al Haque, and F. Sharmin, "Variable Rate Steganography in Gray Scale Digital Images Using Neighborhood Pixel Information. The International Arab Journal of Information Technology (IAJIT), Vol. 7, No. 1, 2010, PP34-38.

[7] M. O. Al-Dwairi, Z. A. Alqadi, A. A. Abujazar, and R. A. Zneit, "Optimized True-Color Image Processing," World Applied Sciences Journal, vol. 8, 2011, PP 1175-1182.

[8] A. Bamatraf, R. Ibrahim, and, M. N. B. M. Salleh, "Digital watermarking algorithm using LSB", IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE), 2010, PP 155-159.

[9] K. Dasgupta, J. K. Mandal, and P.Dutta, "Hash Based Least Significant Bit Technique for Video Steganography (HLSB)", International Journal of Security. Privacy and Trust Management (IJSPTM), Vol. 1, No. 2, 2012.

[10] N. Sruthi, A. V. Sheetal, and V. Elamaran, "Spatial and spectral digital watermarking with robustness evaluation", 2014 IEEE International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC), 2014, PP 500-505.

[11] S. Ghosh, S. De, S. P. Maity, and H. Rahaman, "A novel dual purpose spatial domain algorithm for digital image watermarking and

cryptography using Extended Hamming Code", *IEEE 2nd International Conference on, Electrical Information and Communication Technology (EICT)*, 2015, PP167-172.
[12] R. C. Gonzalez, and R. E. Woods, "Digital Image Processing", 3rd Ed., Pearson Prentice Hall Pearson Education, 2008.
[13] O. E. Ramos, and B. Rezaei, "Scene Segmentation and Interpretation Image Segmentation using Region Growing", 2010.
[14] N. Chowdhury, B. Banerjee, and T. Bhattacharjee, "color image segmentation technique using natural grouping of pixels", International Journal of Image Processing, 2010, PP 1985-2304.
[15] S. K. Naik, and C.A. Murthy, "Hue-preserving color image enhancement without gamut problem," IEEE Trans. on Image processing, vol.12, no.12, December 2003, PP1591-1598.

Hamza A. A. Al_Sewadi is currently a professor at the Faculty of Information Technology, Middle East University (Jordan). He got his B.Sc. degree in 1968 from Basrah University, Iraq, then M.Sc. and Ph.D. degrees in 1973 and 1977 respectively, from University of London (UK). He worked as professor at various universities such as Basrah University (Iraq), Zarqa University, Isra University, and Princess Sumaya for Technology (Jordan), visiting professor at University of Aizu (Japan). His research interests include Cryptography, Steganography, Information and Computer Network Security, Artificial Intelligence and Neural Networks.

Akram N. A. Aldakari, is currently working as lecturer at the High Institute for Comprehensive professions, Baidha (Libya). He is recently got his MSc degree in computer Science from the Faculty of Information Technology, Middle East University (Jordan). His main interest is in Computer security, Data hiding, and Image processing.