A NEW COLOUR IMAGE STEGANOGRAPHY USING LSB APPROACH WITH HALFTONING DETERMINATION EMBEDDING POSITION

Reband Jamil Hassan[1], Ghazali Sulong[2]

[1], [2] Faculty of Computing, Universiti Teknologi Malaysia (UTM) 81310, Johor Bahru, Malaysia

E-mail: [1] reband.jamil@gmail.com, [2] ghazali@utmspace.edu.my

Abstract— Steganography is the art and science of encoding secret messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the hidden message. The Least Significant Bit (LSB) insertion is a well-established method for embedding the secret message that is known for its superiority in terms of imperceptibility; however, it suffers from robustness against Chi-square attack. In this study, a new colour image steganography technique is proposed using the LSB insertion coupled with a halftone image, which is used to determine embedding pixels. The technique involves four stages: secret message preparation, image halftoning, embedding and extraction. Firstly, the secret message is converted from character to a bit-stream. Secondly, a halftone image is created from the cover image in order to determine the embedding pixels. Thirdly, the secret message's bit-steam is sequentially embedded onto the embedding pixels until all the bits are exhausted. Upon the embedding, final stage is to extract the embedded secret message; this is done in the same way as embedding process except it is performed in a reverse order. A series of experiments is conducted using a standard dataset to evaluate the performance of the proposed method in terms of both imperceptibility and robustness. The experimental results have revealed that the method yielded very high imperceptibilty with Peak Signal-to-Noise Ratio averaged at 67dB, and was able to withstand against the Chi-square attack.

Index Terms— Steganography, Cover image, Stego image , Halftoning, LSB, Secret message.

1 INTRODUCTION

Steganography is a technique used to hide a message or disguise information from the view of all but other the authentic receiver [7]. The term steganography literally means "covered writing" [3,5]. Before the age of computer, steganography is used by people to secretly hide messages in various objects in order to disguise these messages from third party viewing. In today's world the technique is used ensure privacy, anonymity and secure communication of sensitive data on internet. The main idea of this technology is to hide a message inside other content which is less important, thereby drawing attention away from the hidden message.

The technique is a branch of cryptography but unlike the original technique which encrypts messages and makes them unreadable, steganography conceals the presence of the message entirely. The technique can be applied to text, audio, video or image media.

In steganography the carrier media include: audio, text, video, and any other type of digital medium [6]. Audio steganography hides messages inside an audio medium. In this type the hidden audio must remain undetectable; the techniques include Amplitude Modification, Spread Spectrum Coding, Echo Hiding, Phase Coding, and Least Significant Bit (LSB).

For text Steganography the carrier medium is text file. Many techniques can be used for text steganography including Feature coding, Word-Shift Code, Line-Shift Code, syntactic, cover generation, and semantic techniques [3,11,13]. Image Steganography used image files as carrier mediums to hide the secret message. The Least Significant Bit (LSB) technique is one of the image steganography techniques used for digital files [3,8]. The method is good for its simple embedding and de-embedding processes.

2 PROBLEM BACKGROUND

The Least Significant Bit (LSB) approach is the most commonly used technique of image steganography. It is simple to use and the resulted file does not arouse third party suspicion since the method is based on dismissing least significant bits of all bytes and replacing these bits of embedded information [4]. The technique provides better imperceptibility of image without distortion and increase capacity without affecting the image quality.

The main drawbacks of Least Significant Bit (LSB) technique of image steganography it is vulnerable to statistical [1] and visual attack [14] and still suffers the problem of security [4]. The approach cannot be used with GIG or JPEG files formats. The approach inserts a bit stream message in the continuous pixels of the host image whenever examine transferring image for suspicious data which consequently reveals the secret data to third parties. The original message is required to extract data from cover message [4].

ISSN 2229-5518

This research is an attempt to improve the security level of LSB technique of image steganography by using halftoning technique (2x2 dithering) that embeds a secret message into RGB colour host image.

3 RELATED WORKS

There are several similar researches by different authors on the subject of image steganography and Least Significant Bit (LSB) approach in particular.

Reddy et al. (2011) proposed a method for image steganography called the Selected Least Significant Bits method (SLSB). SLSB utilizes the color component of insignificant pixels so that the changes caused by the embedded data will not be noticed and then altering the remaining components of the pixel color so that they take on the color nearest the original color. The method decreases the chances of secret data exposure to third parties but the result of the evaluation shows a very low PSNR value.

Maryam Habib (2013) proposed a method to increase the robustness of text message steganography by hiding the message in appropriate places within the image using shuffled leaping frog algorithm (SLFA). The result obtained by applying the proposed method on 20 sample images shows the appropriateness of method but the proposed technique still suffers the problem of security.

The work of Karim et al., (2011) proposed a new approach for LSB based image steganography using secret key. The technique is to store hidden messages into different.

LSB image positions using secret key. However PSNR value found in the evaluation was relatively low 53.76-53.79 and the capacity is only 24bits/pixel.

In general, the security level of LSB method of image steganography is still relatively low despite all the attempts to improve it in the past. The method is good for its simplicity of usage but the aspect of security still needs to be improved.

4 THE PROPOSED APPROACH

The procedure begins by defining halftone dither (2x2) matrix then dividing the host image into R, G and B channels. The B channel is selected and divided into Q blocks of size (2x2) pixels. A block of Q blocks is selected and compared with the defined halftone dither matrix in order to create the halftone image, as shown Figure 1.

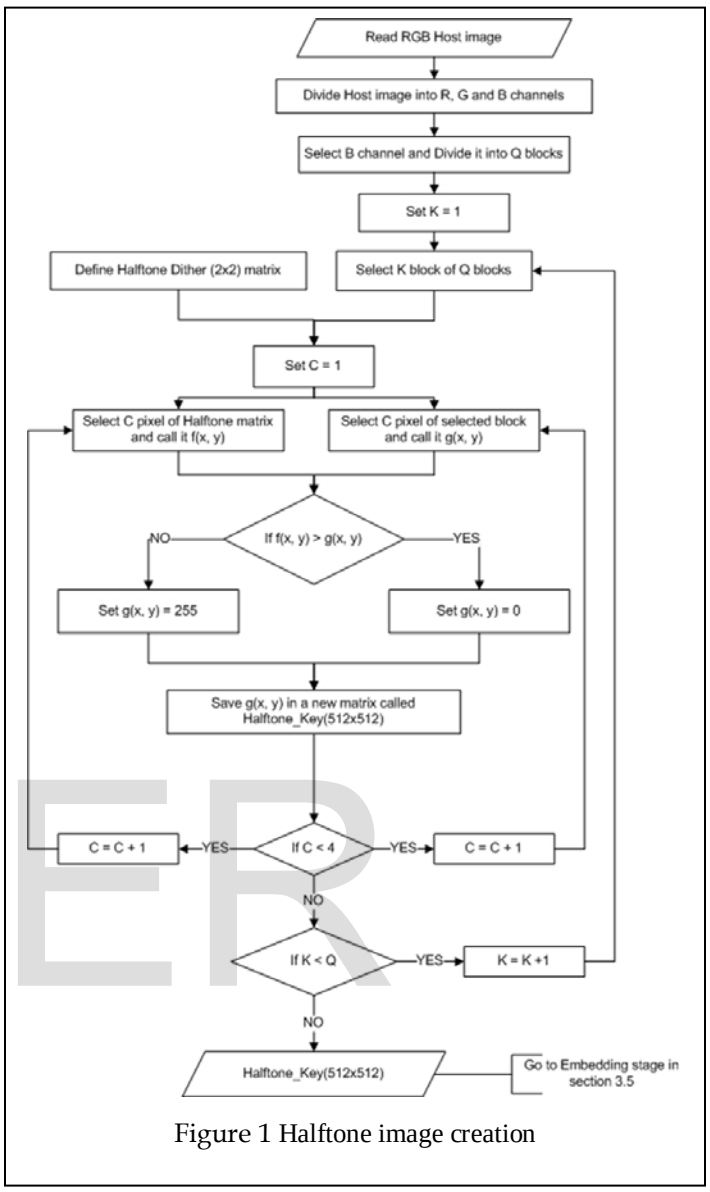

Figure 1 Halftone image creation

The following steps explain how the secret message embedding algorithm works:

Input: Host image, Halftone image.
Output: Stego image.
.

Step 1: Read Host image.

Step 2: Divide host image into R, G and B channels.

Step 3: Select B channel and divide it into Q blocks of (2x2) pixels.

Step 4: Read Halftone image.

Step 5: Find the summation for Halftone_Key black and white pixels as BP and WP respectively.

Step 6: if BP is greater than or equal to WP then, set SP to zero; otherwise, set SP to one and go to step 11.

Step 7: Divide Halftone image into Q blocks of (2x2) pixels.

Step 8: Set $K = 1$, $C = 1$ and ML = length of the message vectors.
Step 9: Select K block of Q blocks of each of B channel and Halftone image.

Step 10: Select a pixel of K blocks of each of B channel and Halftone image and call them $f(x, y)$ and $g(x, y)$, respectively.
Step 11: If $f(x, y) = SP$, proceed to the next step; otherwise, go to step 15.
Step 12: If $C > ML$ then, go to step 15; otherwise, proceed to next step.

Step 13: Select C bit of secret message vectors.

Step 14: If the selected bit is equal to 0 , change the $g(x, y)$ to bit stream and change its LSB to 0 and proceed to next step; otherwise, change its LSB to 1 and proceed to next step. Figure 1.2 shows embedding four bits of the secret message into four pixels of B channel. (a) Four bits of secret message. (b) Four pixels bit stream of B channel before embedding. (c) Four pixels bit stream of B channel after embedding.

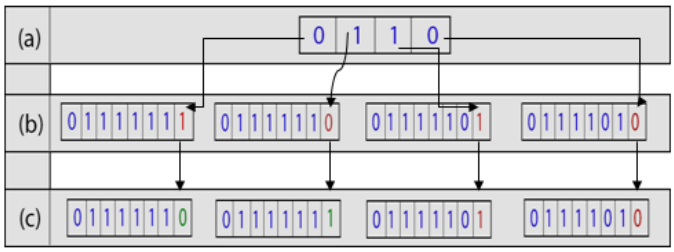

Figure 1.2 four bits of secret message and four pixels bit stream of B channel before and after embedding.

Step 15: Convert back $g(x, y)$ value to decimal and save it in a new matrix called stego_B(512x512).

Step 16: Increment C by one.

Step 17: If the selected block pixels are all selected , proceed to the next step; otherwise, go back to step 10.

Step 18: If K < Q , increment K by one and go back to step 9; otherwise, proceed to the next step.

Step 19: Stego_B (512x512) is obtained.

Step 20: Merge R, G channel with Stego_B (512x512) matrix to obtain RGB Stego image.
Step 21: RGB Stego image.

On receiving the message the secret message will be extracted in the following steps:

Input: Stego image, Halftone_Key, ML (Message Length).

Output**:** Extracted Secret Message.

Step 1: Read Stego image.

Step 2: Divide Stego image into R, G and B channels.

Step 3: Select B channel and divide it into Q blocks of (2x2) pixels.

Step 4: Read Halftone_Key.

Step 5: Find summation for Halftone_Key for each of black and white pixels as BP and WP respectively.

Step 6: if BP is greater than or equal to WP then, set SP to zero; otherwise, set SP to one and go to step 11.

Step 7: Divide Halftone_Key into Q square blocks of (2x2) pixels.
Step 8: Set $K = 1$ & $C = 1$, where K represents the selected block and C represents the extracted message bit position.

Step 9: Select K block of Q blocks of each of B channel and Halftone_Key.

Step 10: Select a pixel of K block of each of B channel and Halftone_Key at a time and call them $f(x, y)$ and $g(x, y)$, respectively.
Step 11: If $f(x, y) = SP$, proceed to next step; otherwise, go to step 15.
Step 12: If $C > ML$, go to step 17; otherwise, proceed to next step.
Step 13: Convert $g(x, y)$ to bit stream and extract its LSB bit.

Step 14: Save the extracted LSB bit into a new matrix called A (1, ML).
Step 15: increment C by one.

Step 16: If the selected block pixels are all selected , proceed to next step; otherwise, go back to step 10.
Step 17: If $K \geq Q$ or $C > ML$, proceed to next step; otherwise, increment K by one and go back to step 9.

Step 18: Matrix A (1, ML) is obtained.

Step 19: Convert matrix A to ASCII code.

Step 20: Convert ASCII code to character.

Step 21: Extracted Secret message.

5 DATASET

The data used in this study will be explained briefly. These data consist of six standard color images of (512 × 512) pixels including: Lena, Baboon, Pepper, Airplane, Car and Sailboat. These images are used as cover images. In addition, these images contain different color images and help to obtain precise results for evaluating the imperceptibility. Then, an attack is applied on these images after the embedding process. Usually, these images are used in all studies that deal with data hiding.

6 RESULT AND DISSCUSION

The proposed method obtained sufficient result, and can be evaluated for imperceptibility, capacity, robustness and security. The analysis results are discussed in the following subsections.

6.1 Imperceptibility

In this study, the quality of the image is estimated by PSNR metric. A greater PSNR value indicates a lower degree of generated image distortion by embedding algorithm.

The results of the proposed LSB technique used on different images of Lena, Baboon, Pepper, Airplane, Car and Sailboat, are shown inTable (1, 2, 3, 4, 5 and 6 respectively) as cover images. The results show that the proposed method produces high PSNR values in all the various images used in the experiment.

Proposed LSB	Lena (512 x 512)				
Max capacity (KB)	17				
Message size (KB)	1	5	10	15	17
PSNR (dB)	74.1915	67.0545	63.8073	62.042	61.3765

Table 1 PSNR for stego Lena image

Proposed LSB	Baboon (512 x 512)				
Max capacity (KB)	18				
Message size (KB)	1	5	10	15	18
PSNR (dB)	74.3551	67.0705	63.8018	62.0224	61.1143

Table 2 PSNR for stego Baboon image

Proposed LSB	Pepper (512 x 512)				
Max capacity (KB)	19				
Message size (KB)	1	5	10	15	19
PSNR (dB)	74.1804	67.0766	63.8487	62.1029	60.9944

Table 3 PSNR for stego Pepper image

Proposed LSB	Airplane (512 x 512)						
Max capacity (KB)	29						
Message size (KB)	1	5	10	15	20	25	29
PSNR (dB)	74.2161	67.0619	63.8118	62.0032	60.5969	59.6192	58.9585

Table 4 PSNR for stego Airplan image

Proposed LSB	Car (512 x 512)					
Max capacity (KB)	22					
Message size (KB)	1	5	10	15	20	22
PSNR (dB)	74.2771	66.7673	63.6208	61.8883	60.5658	60.1824

Table 5 PSNR for stego Car image

Proposed LSB	Sailboat (512 x 512)				
Max capacity (KB)	19				
Message size (KB)	1	5	10	15	19
PSNR (dB)	74.3067	66.9925	63.7924	62.0348	60.9337

Table 6 PSNR for stego Sailboat image

In Lena image the number of black pixels is (121534) and white pixels is (140610) therefore white pixel is selected for embedding purpose based on its larger number. In Baboon image the number of black pixels is (113263) and that of white pixel is (148881) so white pixel is selected for embedding. In Pepper image the number of black pixels is (157529) and number of white pixel is (104615) so black pixel is selected. And for the Airplane image number of black pixels is (23683) and white pixels is (238461) so white pixel is selected. The same applies to Sailboat, in which the number of black pixels is (109121) and that of white pixel are (153032) hence the white pixel is selected.

Since the PSNR value in all of the experiments is above 36 dB, the changes in the images are not detectable by the human eye as shown in Figure (2,3,4,5,6 and 7).

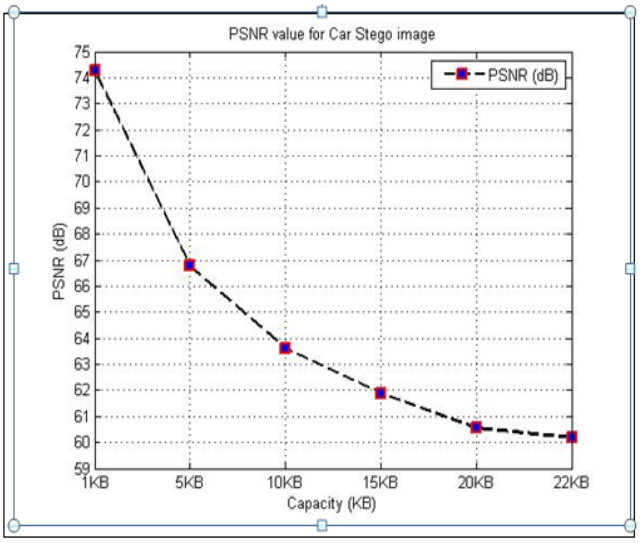

Figure 2 Performance of PSNR for Car Stego image

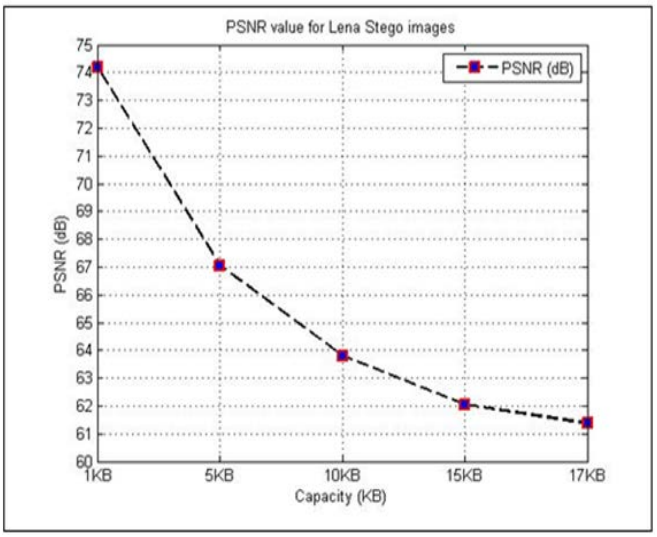

Figure 5 Performance of PSNR for Lena Stego image

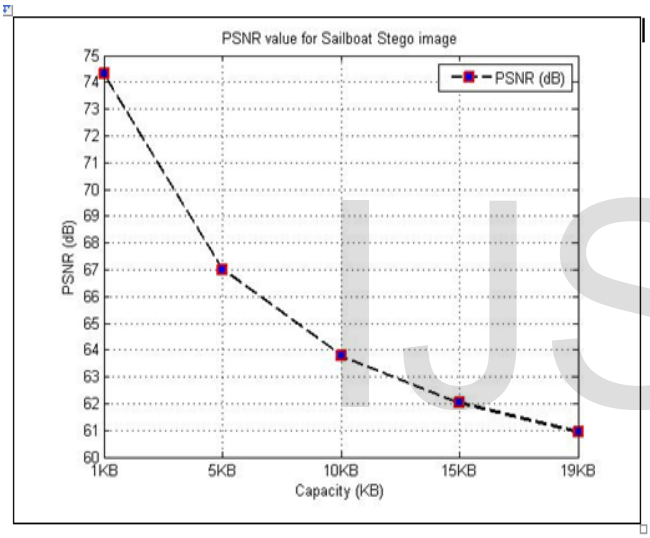

Figure 3 Performance of PSNR for Sailboat Stego image

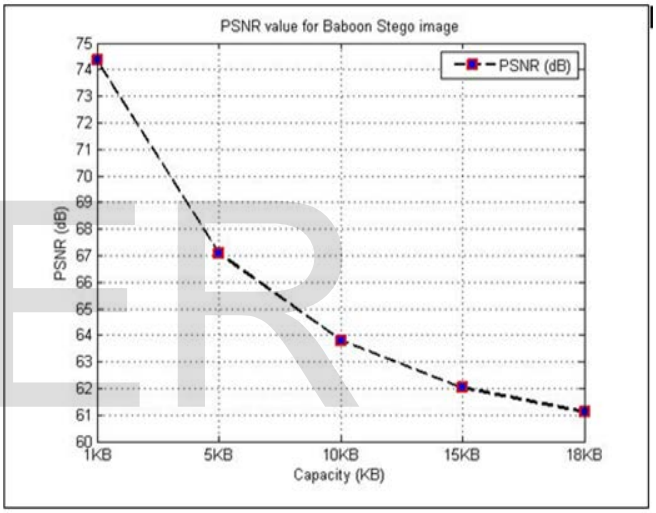

Figure 6 Performance of PSNR for Baboon Stego image

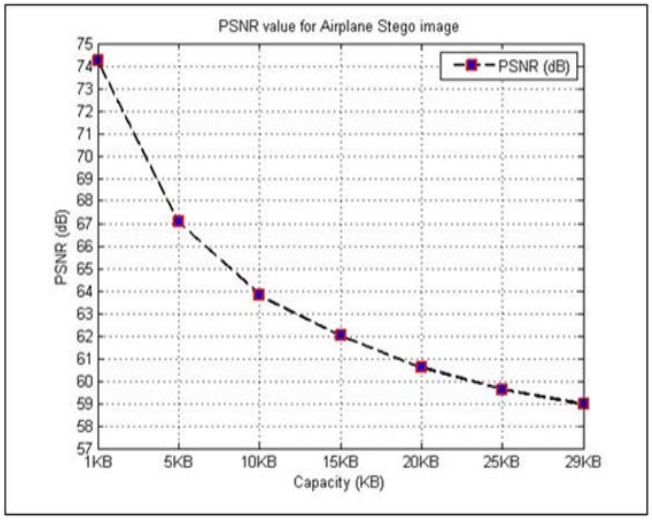

Figure 4 Performance of PSNR for Airplane Stego image

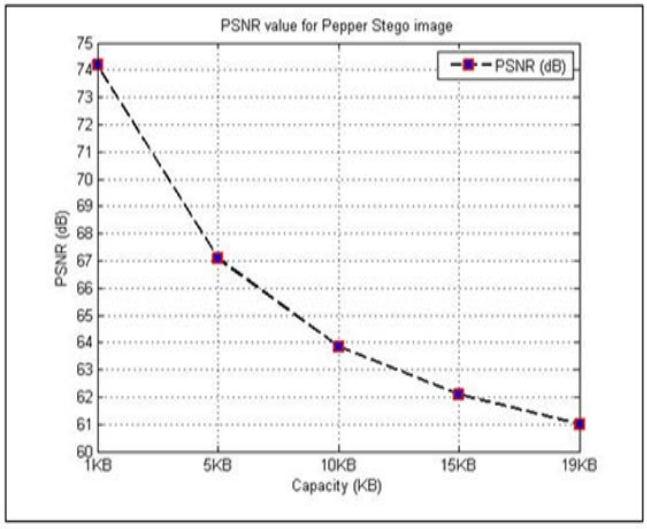

Figure 7 Performance of PSNR for Pepper Stego image

6.2 Chi-Square Test

The technique of Chi-square is one of the most reliable attack method measuring the robustness of secret message in Stego-image due to its ability to determine the probability of hidden messages without physically tempering the image [10].

The Lena image was used in this research for chi-square test. The percentage of the image was varied from 0 to 100, as shown in Figure 8. The probability of secret message in the image range from 0 to the value obtained from Chi-square attack. A value equal to zero indicates no secret message and higher certainty is achieved as the value approaches one. For all the percentages of the image the probability is about zero which indicates the cover image contains no hidden message.

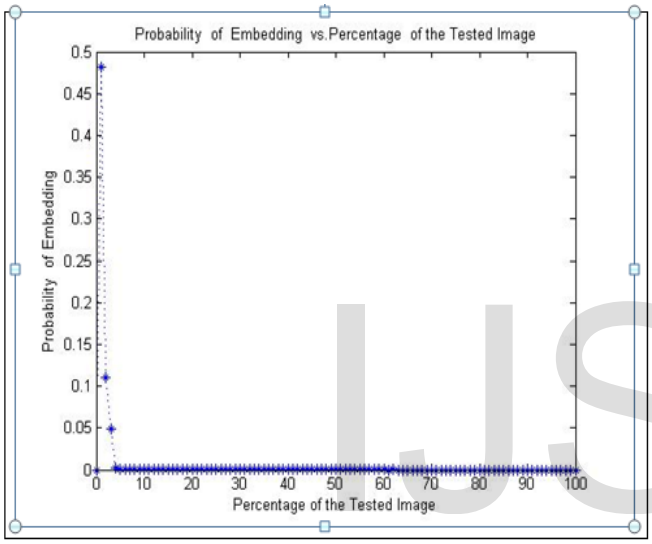

Figure 8 result of Chi-square attack on original Lena image

6.3 Benchmarking

The Airplane image 512x 512 in size with a maximum size of 29KB was used for benchmarking against a previous work Ahmed.A.M. (2012). A positive result was obtained as shown in Table 7 with a clear improvement in both the PSNR and capacity using the Airplane image of same size of message .

Proposed method		Ahmed.A.M.(2012) method	
Message size (Byte)	**PSNR**	**Message size (Byte)**	**PSNR**
1024	**74.2161**	**n/a**	**n/a**
5120	**67.0619**	**5000**	**66.10**
10240	**63.8118**	**10000**	**63.11**
15360	**62.0032**	**15000**	**61.35**
20480	**60.5969**	**20000**	**60.11**
25600	**59.6192**	**n/a**	**n/a**
29696	**58.9585**	**30000**	**58.35**

Table 7 shows the benchmark proposed method with Ahmed.A.M. (2012)

7. CONLUSION

The research proposed a new colour image steganography method using LSB approach with Halftoning technique (2x2 dithering) that embeds a secret message into RGB colour host image. The proposed algorithm was evaluated using PSNR and chi-square testing. Six stages including preparation, image creation, embedding, applying attack, message extraction and performance evaluation are applied on of six images of Lena, Baboon, Pepper, Airplane, Car and Sailboat of size 512 × 512. The result of Chi-square
testing shows the stablity of the method against various forms of attacks which make extracting the hidden message by third person more difficult.

8. REFERENCES

[1] Avcibas, I., Memon, N. and Sankur, B. (2003). Steganalysis using image quality metrics. Image Processing, IEEE Transactions on, 12, 221-229.
[2] Ahmed.A.M.(2012). A 2-Tire Datahiding technique using an improved exploiting Modification Direction Method and Huffman coding. Msc, Universiti Technologi Malaysia,Johor.
[3] Bennett, K. (2004). Linguistic steganography: Survey, analysis, and robustness concerns for hiding information in text.
[4] Chan, C. K. and Cheng, L. M. (2004). Hiding data in images by simple LSB substitution. Pattern recognition, 37(3), 469-474.
[5] Cheddad, A., Condell, J. ,Curran, K. and McKevitt, P. (2010). Digital image steganography: Survey and analysis of current methods. Signal Processing, 90, 727-752.
[6] Durić, Z. and Jajodia, S.(2001). Information hiding: steganography and watermarking: attacks and countermeasures, Springer.

[7] Habibi, M., Karimi, R., & Nosrati, M. (2013). Using SFLA andLSB for Text Message Steganography in 24-Bit RGBColor Images. International Journal of Engineering, 2(3), 68-75.
[8] Kessler, G. C. (2004). An overview of steganography for the computer forensics examiner. Forensic Science Communications, 6, 1-27.
[9] Karim, M. (2011). A new approach for LSB based image steganography using secret key. International Conference on Computer and Information Technology on, 2011. 286-291.
[10] Lee, Y.K., Bell, G., Huang, S.Y., Wang, R.Z. and Shyu, S.J. (2009).An advance Least-Significant-Bit Embedding Scheme for Steganographic Encoding .Advance in Image and Video Technology. Berlin/Heidelberg:Springer,349-360.
[11] Rabah, K. (2004). Steganography-the art of hiding data. Information Technology Journal, 3, 245-269.
[12] Reddy, V. L., Subramanyam, A. and Reddy, P. C. (2011). Implementation of LSB Steganography and its Evaluation for Various File Formats. Int. J. Advanced Networking and Applications, 2, 868-872.
[13] Westfeld, A. and Wolf, G. (1998). Steganography in a video conferencing system. Information Hiding, Springer, 32-47.
[14] Westfeld, A. and Pfitzmann, A. (2000). Attacks on steganographic systems. Information Hiding, Springer, 61-76.