IMPROVED STEGANOGRAPHY APPROACH USING COLOR GUIDED CHANNEL IN DIGITAL IMAGE

Mr. Nirav Patel[1], Mrs. Risha Tiwari[2]

[1]*Student, Department of Computer Science and Engineering, Hasmukh Goswami college of Engineering, Ahmedabad, Gujarat, India*
[2]*Assi. Prof., Department of Computer Science and Engineering, Hasmukh Goswami college of Engineering, Ahmedabad, Gujarat, India*

ABSTRACT

In Today's World most of the digital data is transmitted from one place to another through the medium of Internet. So there is a possibility of insecurity of data through the communication medium. Steganography is one of the technique which can be used to hide the data which means it is a method of communicating or connecting in a hidden manner. Data hiding plays a important role to serve this purpose. The proposed data hiding method uses a color image (RGB). In this approach data does not embedded directly. This approach use both direct and indirect form. In this method First of all secret message is converted into binary bit stream i.e. in the form of zeros (0) and ones (1). After that Run Length Encoding is used to compress the bit stream, in which the repeated occurrences are rewritten in the form of digit & its count. Here the data is embedded in direct & indirect form. One channel is used as identifier channel and the other two channels used for carry the data. In direct form bits are directly embedded in the LSBs of channel-1 and channel-2. In indirect form the count is embedded in the 2 LSB of channel-2 and 1 LSB of channel-1.

The results show that the proposed approach gives better performance & proposed method improves visual quality of Stego image in terms of MSE and it has more embedded capacity as compare to existing method.

Keyword: *Steganography, Data Hiding, Data Payload, Direct & Indirect data embedding, Identifier channel, RGB Channels, Run Length Encoding, visual quality.*

1. Introduction

In general term disguise or hiding secret data/message inside a cover object is called "**Data hiding**" [1]. Inside cover objects we can embed secret message. Commonly, to achieve this purpose for the cover object, an image file, a video file or an audio file can be reserved.

However, as one who is Owners of sensitive documents and files must wat to protect themselves from unwanted access, spying, copying, theft and false representation. This problem has been solved by using a technique named with the Greek word "steganography" it is mean hiding information [1]. **"Steganography"** is the one kind of skill or art for hiding data invisibly or secretly in a digital cover media. To hide the element or doubt, that the secret data is present or hidden inside a cover object is the main goal line of Steganography approach. This is achieved by keep the attacker away from communication by avoiding the doubt that the secret message is derived in cover object. The Watermarking is one another form of data hiding [2]. Watermarking can be derived as the injecting information into a digital cover media with the moto of providing authenticating or validating for digital documents. Generally, Watermarking is a protective technique that guard or claims the owner's right on digital stuff [2].

Steganography and Watermarking are can be derived as same thing or same family [1,2]. Cryptography is also connected to them, as using some cryptographic algorithms make the secret data gibberish. On other side, in Steganography the message is cover, so that it cannot be seen. If secret message is derived in the form of cipher text, then it may incite doubt on the recipient side, while a message in invisible formed with any Steganography technique will not. If more security is concern then we can apply any cryptography algorithm on secret message before embedding. The goal of Steganography is crushed, if the existence of hidden secret message is uncovered or even just suspected, even the message content is not retrieved or decoded [2]. If a hidden message is encrypted and then embedded, then in order to access the original message it must also be decrypted, which provides another layer of protection and also higher security [2].

In today's world Information Security is very important. A main tasks for different people of different fields are Verification and Authentication. For ex. military people, research institute and scientist. To protect different digital documents from attacker, Information security using image authentication and verification has become very useful [11]. For ex, Military records are confidential and private records and they must not be exposed to either public or opposing country. So for that reason we must require the best embedding pattern or approach. For military records, hiding the data of a message using encryption is not fully essential, but the obtainability of secret message, hiding its sender and receiver is also important. The affect or chances of Leaking or unauthorized access of military data or documents are very huge as in compare to financial loss [11]. Many organizations and people of group are try to accessing such records, to target the national security as well as personal profits.

2. Literature Survey

Edge adaptive Steganography [2] describe an approach that uses sharp edges of an image to hide secret message bits and later uses smooth regions to conceal remaining secret message bits. Pre-processing and region selection is performed in starting stage. Now data embedding is performed in this selected region, if the selected region is large and it has capacity to cover secret message (M) within in it, then The Algorithm is used to transform plaintext into encrypted text. Now the starting process is derived at first on the sharp edges of the cover image. Here the portion of image where the image changes strictly is known as Sharp image for cover image. In image the first data bits are embed using those Sharp edges and it leave the other smooth regions as they are and also leave remaining data into this region as they are.

Standard Deviation Converges [3] is based on mean & standard deviation of pixel values of cover image. This algorithm works based on variable bit embedding approach. In this paper, different bit hiding method is used in this algorithm. Mean and Standard Deviation of each pixel for the complete image is the main aim or Key element of this algorithm. The algorithm first distinguishes the mean and standard deviation of each pixel for the whole image. Then after this process, the embedding can derived.

In Message Segmentation [4], to improve visual quality of stego images & security, classical Least Significant Bit Replacement algorithm is modified. Which divides a secret message into several segments & independently embeds these segments into cover image. This approach requires a considerable amount of time to perform embedding operation.

A new approach of steganography has been proposed in [5] that uses Huffman coding scheme as its building block. Secret message is converted into Huffman encoded bit stream & then after 3 constituents are embedded inside a cover image. First Huffman encoded bit stream, second the size of Huffman encoded bit stream and last but not least the Huffman table itself for decoding purpose.

Segmenting and Hiding Data Randomly Based on Index Channel [6] method works as fixed bit embedding method & it cleverly uses three channels (R, G and B) of a colour image to hide the secret message. First of all the secret message is segmented into two segments that are EVEN and ODD segment. Figure 1 shows the segmentation operation.

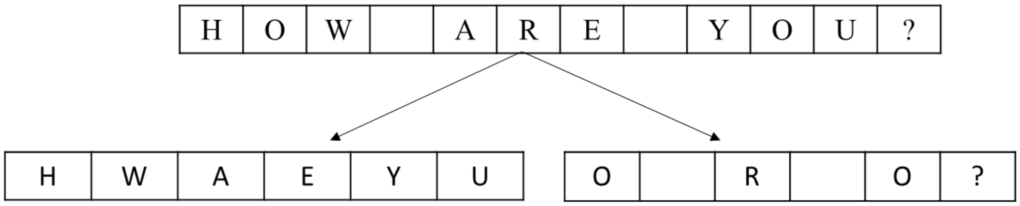

Figure 1. Example of Segmenting Process

After that to hide the data one channel of the three channels is used as index channel and the rest of the channels carry secret data. 4-bits of data can be embedded in single pixel in this type of algorithm. Number of 1s in the index channel decides the embedding process. 4-bits from even segment is embedded into the other two channels 2bits per channel, if the number of 1s in the index channel is even. 4bits from odd segment is embedded into the other two channels 2bits per channel, if the number of 1s in the index channel is odd.

Table shows embedding operation and index channel.

No of 1's in Index channel	First Channel	Second Channel
Is Even	Embed 2 bits from even segment	Embed 2 bits from even segment
Is Odd	Embed 2 bits from odd segment	Embed 2 bits from odd segment

Table 1. Index channel and embedding operation [6]

Now we can derived it with the help of an example in following fig.

	Index Channel	First Channel I	Second Channel II	
Pixel 1:	10101101	000011**00**	101010**01**	Odd
Pixel 2:	10110001	000011**01**	101010**10**	Even
Pixel 3:	10101011	000100**11**	101010**00**	Odd

Figure 2. Example of Index channel and embedding operation

3. Proposed Methodology

The proposed approach uses two images as cover and stego image. The cover image use as colour image RBG. The secret data is inserted into main image using three different channel as R, G, and B of color image. This approach uses two different image as cover image which is the original looking image without changes, known as *Cover image*. And second one is the stego image which contain the secret information that we want to hide in the cover image, known as stego image. There can be different way or form of hiding massage into cover image as plaintext, cipher text, an image file etc.

Now as we know the word Pixels in digital image, as Digital images are stored in the computer as an array of several points, that points are known as Pixels. Different image has the different values for different Pixels. Like, grayscale image can have pixel value in the range of 0 – [2^8-1] i.e. 0 – 255. It is because a grayscale image is of 8-bits and so the range is 0 – 255.

As different the color image has three different monochromes i.e. Red(R), Green (G), Blue (B). Each is of 8-bit, so the whole color image is of 24-bits. From this the pixel value of a color image is in the range of 0 – [2^{24}-1] i.e. 0 – 16777215.

In the proposed approach, the secret message is not directly embedded into the cover image, it started as, all the secret message or information is first converted into binary bit stream (0s & 1s) by replacing each and every character with its equivalent 8 bits binary value. After that compression algorithm apply on the binary bit stream. Here we use the Run Length Encoding algorithm for the same proposed. In this algorithm the data which is repeated is rewritten in form of count and single digit.

Now starting with the Data Embedding phase, in this approach we uses three channels of color images. All three has identify differently as case identifier which is identifier channel which used to derive to indicate **embedding type**. Embedding type can be derived as Direct and **Indirect** case. In case identifier channel **0** in LSB indicates **direct embedding** and **1** indicates **indirect embedding**.

In **Direct embedding**, if 1 LSB of the case identifier channel is set to **0** then it known as direct embedding case. Now other Remaining two channels derived as first Channel-I & second channel-II. The both are used to hide either **1 bit** or **2 bits** from data stream. Now again the first channel–I is used to indicate 1 or 2 bits are embedded, which known as **carrier identifier channel.**

Operation of Direct Embedding is shown in following fig.

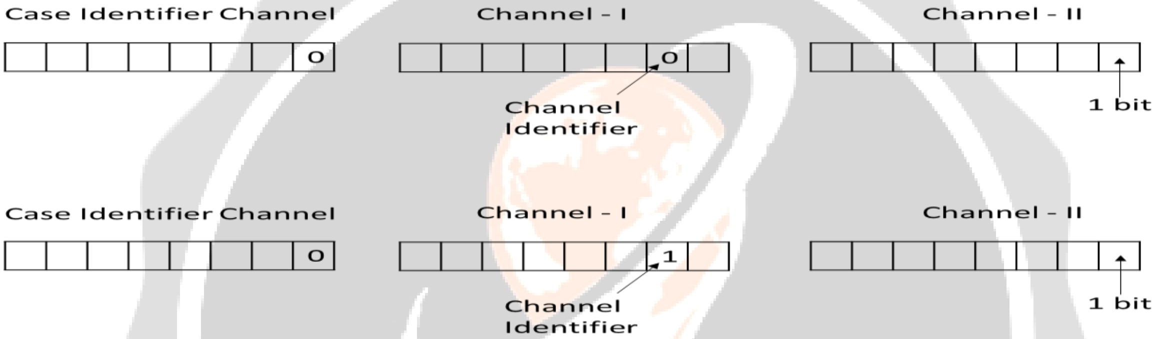

Figure 3. An example of Direct Embedding Operation

In **Indirect embedding case,** if the count is higher than count 3 than digit (0/1) & its count is embedded into the cover image. In indirect embedding case 1 LSB of the identifier channel is set to **1** and remaining Other two channels named same as direct embedding case first Channel-I and second channel-II. Both are used to embed bit identifier digit and its count into the cover image.

Now in first Channel-I, 2nd **LSB** of Channel-I is used for embedding the **Bit identifier digit**. And the **Count** is embedded using **LSB** of Channel-I and **2 LSB** of channel-II.

It means to say that **2 LSB** of channel-II is used to hide **2 LSB** of count and **1 LSB** of Channel-I is used to hide **1 MSB** of count.

Fig 4 shows an example of indirect embedding.

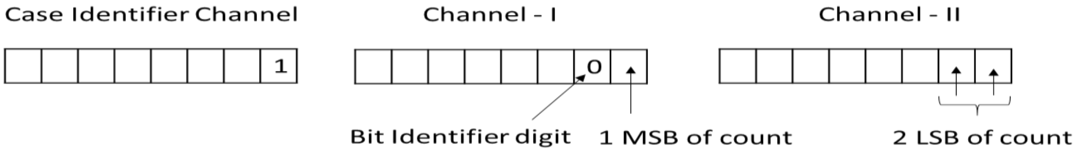

Figure 4. An example of Indirect Embedding Operation

Algorithm 1: Embedding Operation

Inputs: A cover image with size (m × n) and secret message.

Outputs: An image which contained secret message (size m × n)
Step 1: first the secret message is converting into its equivalent binary data stream.
Step 2: Than on binary data stream apply compression algorithm.
Step 3: Now scan next 2 digits as digit & its count from result of compression algorithm encoded bit stream.
Step 4: If count is less than 3(count<3) then set LSB of first channel to 0 else to 1 which is case identifier channel.
Step 5: If the LSB of case identifier channel is set to 0 which indicate Direct embedding case, then check the count.
If count = 2 then set 2^{nd} LSB of channel– I to 0 & changed LSB of channel–II with its digit. If count = 1 then set 2^{nd} LSB of channel–I to 1 & changed LSB of channel-II with its digit.
Else if count is higher than 3 than put 2^{nd} LSB of channel–I to bit identifier digit (0/1) which indicate Indirect embedding case than replace 2 LSB of channel-II with 2 LSB of count and 1 LSB of channel–I with 1 MSB of count.
Step 6: Now Repeat step 3 to step 5 until entire secret message bit stream end.
Step 7: Now Change 2 LSB of second channel – II with 0 and change 1 LSB of first channel – I with 0 in order to put implicit end marker for it.

The extraction operation algorithm is as shown below. Which takes a stego image as an input and produce a secret message file as an output.

Algorithm 2: Extraction Operation

Inputs: An Embedded image which contain the secret message.
Outputs: A hidden secret message.
Step 1: To hold bit steam start with null data stream S.
Step 2: Now from case identifier channel extract 1 LSB.
Step 3: Now check LSB, extract 2^{nd} LSB from channel–I if the LSB of case identifier channel is set to 0 & extract 1 bit from LSB of second channel–II.
And extract 2^{nd} LSB from channel–I for bit identifier digit if the LSB of case identifier channel is set to 1. And then extract 1 LSB from first channel–I and 2 LSB from second channel–II. These all three bits derived the count bit.
Step 4: Attach the extracted digits into null data stream S.
Step 5: Repeat step 2 to step 4 until the count 0.
Step 6: Figure out the original secret hidden message from the final bit stream.

4. Experimental Results

The system is simulated in MATLAB. This chapter provides description about the implementation of Existing method, proposed method & result analysis of them. Here Implementation of both algorithm is carried out using **MATLAB 9.0**. As starting, the cover image is used for inserting secret message is **baboon.png**. Cover image is colour image of size of 512x512 pixels. The message which is to be hide is plaintext file (.txt). The Embedded message is a plaintext of **6400 characters** long. With these all data the visual quality & data embedding capacity is occupied here. The real cover image is compare with the stego image and result is analyzed. In advance for the different color monochrome R, G and B, all these each are taken and analyzed visual quality for each. And it will be done on both cover image and stego image too. Finally histogram for each are provided. The PSNR of cover image and stego image is analyzed.

The following fig. shows the cover image, which is original image which is used to hide secret message into it and the resultant image with 6400 character hidden secret massage will be our stego image.

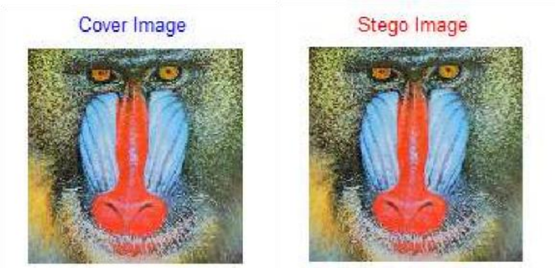

Fig. 5: Cover image and Stego image (colour)

The following Figure shows individual R, G and B monochromes of the cover image and also for the stego image.

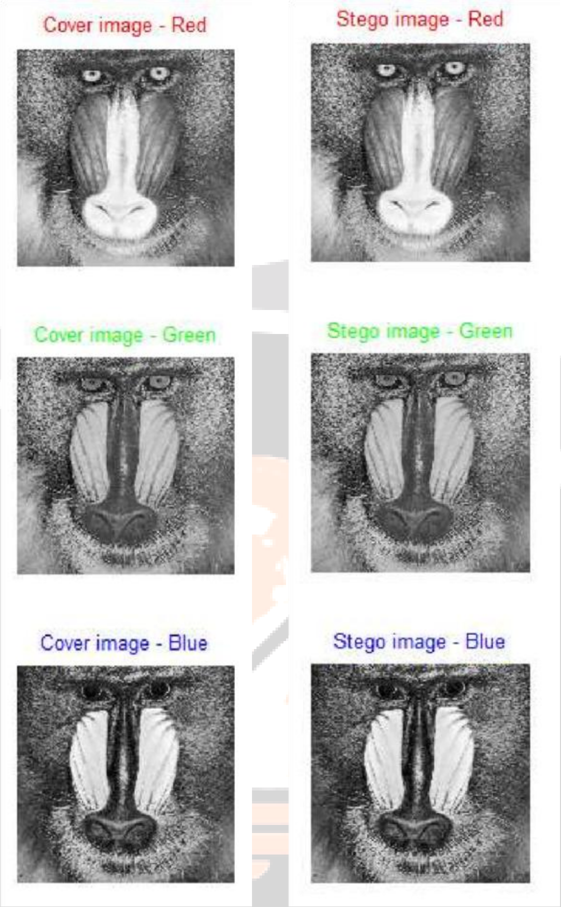

Fig. 6: R, G & B Monochromes of Cover image and Stego image

Next the Histograms of all monochromes R, G and B is derived in Fig 5.7. here the Horizontal axis represents pixel level and vertical axis represents no. of pixels of that particular level. Here we can can figure out that very minor difference between histograms of both cover image and stego image.

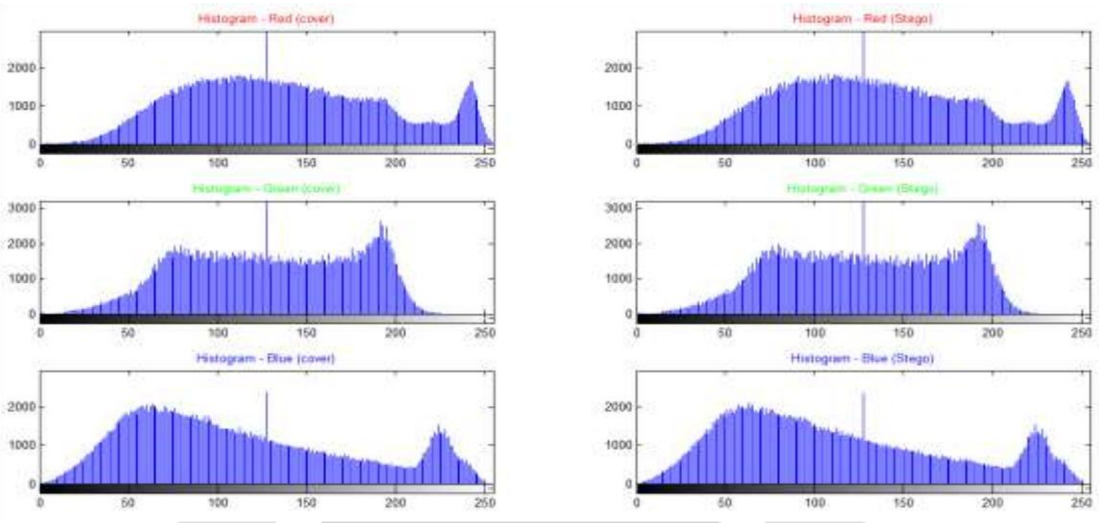

Fig. 7: Histograms for each monochromes of cover image and stego image

Now, the histogram of both cover image and stego image is derived in fig 5.8. here we can easily describe that there is no major visual difference between both histograms of cover image and stego image.

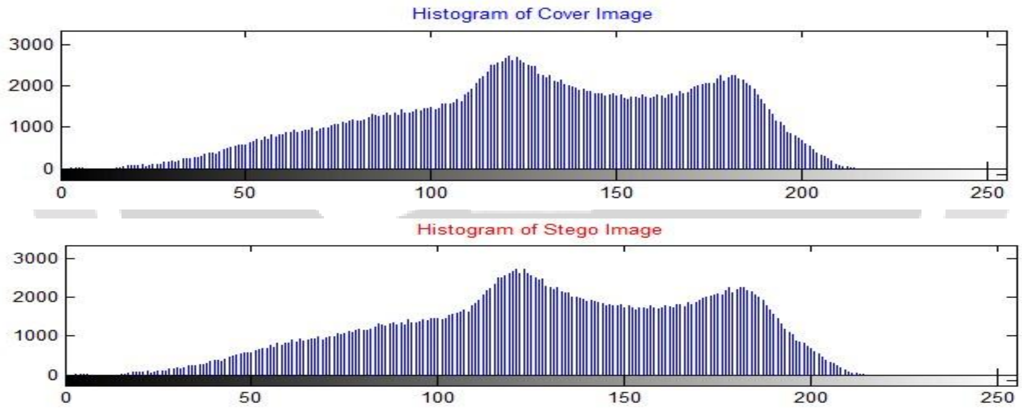

Fig. 8: Histogram comparison of cover image and stego image

After the all operation performed on both Existing and Proposed approach, the graph is generated to compare visual quality of its. In following fig the graph is derived for represent the stego image's three colour components as Red, Blue and Green on block of 50x512 pixels. The following graph represents the no. of pixels is changed with the change in level. The visual quality degrades more, if the number of more pixels are changed.

Following fig. describe the visual quality by comparing with both existing and proposed approach with respective to different color component as Red, Blue and Green as below.

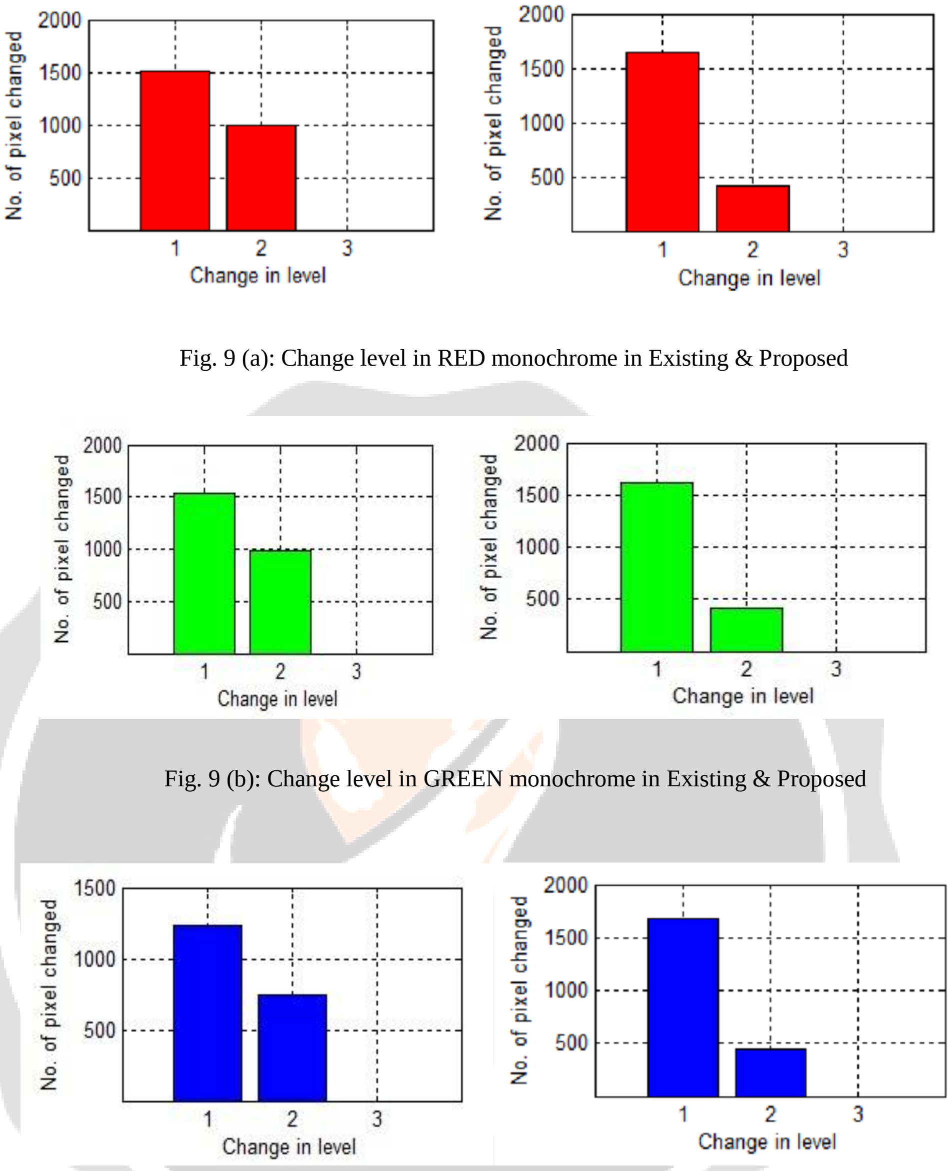

Fig. 9 (a): Change level in RED monochrome in Existing & Proposed

Fig. 9 (b): Change level in GREEN monochrome in Existing & Proposed

Fig. 9 (c): Change level in BLUE monochrome in Existing & Proposed

Fig. 9: Change in level by Existing & Proposed method

Data payload is the capacity can be derived as the capacity of secret data can be embedded in a Stego image. The capacity measure in Bits per pixel.

5. Result Analysis

Now Visual quality also called imperceptibility, which means the Stego image should be as same as the cover image as much possible. If the different between a cover image and a Stego image is now identified than aim will be fulfilled. For the same purpose we have to evaluate PSNR & MSE. The Peak Signal to Noise Ration is the ratio between possible maximum signal value and impact of adding noise by modification of bits. It can be calculated as below.

$$\text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\ [I(i,j) - K(i,j)]^2$$

Where m,n are row and column number of cover image

l(i,j) indicate pixel value for cover image

k(i,j) is the pixel value for stego image

$$\text{PSNR} = 10.\log_{10}\frac{Max_i{}^2}{MSE}$$

Where Max_i^2 is the maximum possible pixel value of the color image.

The PSNR expressed in DB.

Cover Image	**PSNR Value (in db)**	
	Existing	***Proposed***
Baboon	56.47	63.08
Lena	56.44	63.14
Vegetables	49.37	55.91
Airplane	56.49	63.06
Lady	50.59	56.88

Table 2. PSNR values of different cover images

Now the existing approach and the proposed approach will be compared with respect to different parameter as the following table represents,

Here comparisons of both approach is provided. in Table 3 which describe the different parameter wise comparison. From the table 3 we can figure out that the proposed approach provides **6.59 db** high PSNR than the existing approach. Also the existing approach requires **17,434 pixels** less no. of pixels than the existing approach which requires to embed 51,200 bits. Hence the proposed approach improves visual quality by **1.10%** and data embedding capacity by **4.47%.**

It can be describe on the basis of different parameter wise.
Here comparisons of existing and proposed is provided.

Table shows different parameter wise comparison.

	Existing	**Proposed**

No. of bits replaced per pixel	4	3 or 5
Max. no. of bits embedded per pixel	4	9
Security	2 layers	3 layers
Method of Embedding	Direct	Direct & Indirect

Table 3. Comparison of existing & proposed approach

6. Conclusion

In this new era of digital communication, the information security has become a major matter of concern for several fields. So Information security is very important for the same purpose. Implementation results of Existing and Proposed & extensive analysis derived that the proposed method improves visual quality of Stego image in terms of MSE. The visual quality improve **13%** to **30%** in reference to MSE. Also Proposed method derived that more data embedding capacity as compare to Existing method. In more, proposed method provides three layers of security whereas existing method provide two layer of security. So the proposed approach Proposed is a good contender for data hiding as compare to existing method.

References

1. Alaa A. Jabbar Altaay, Shahrin bin Sahib and Mazdak Zamani, “An Introduction to Image Steganography Techniques”, 2012 International Conference on Advanced Computer Science Applications and Technologies IEEE, 2013
2. G. Karthigai Selvi, Leon Mariadhasan, K.L. Sunmugnanthan, ”Steganography Using Edge Adaptive Image”, International Conference on Computing, Electronics and Electrical Technologies [ICCEET] IEEE, 2012
3. Rengarajan Amritharajan, P. Archana, V. Rajesh, G. Devipriya, J.B.B. Rayappan, “Standard Deviation Converges for Random Image Steganography”, IEEE Conference on Information and Communication Technologies (ICT), 2013
4. Dr. Mohammed Abbas Fadhil Al-Husainy, “Message Segmentation to Enhance the Security of LSB Image Steganography”, International Journal of Advanced Computer Science and Applications (IJACSA), 2012
5. Rig Das and Themrichon Tuithung, “A Novel Steganography Method for Image Based on Huffman Encoding”, IEEE, 2012
6. Emad T. Khalaf and Norrozila Sulaiman, “Segmenting and Hiding Data Randomly Based on Index Channel”, IJCSI International Journal of Computer Science Issues, vol. 8, issue 3, No. 1, May 2011
7. G. Sahoo and R.K. Tiwari,” Designing an Embedded Algorithm for Data Hiding using Steganography Technique by File Hybridization”, IJCSNS International Journal of Computer Science and Network Security, VOL. 8 No. 1, January 2008
8. Tamanna Garg, Sonia Vatta,”A Review on Data Compression Using Steganography”, IJCSMC, vol .3, 6 June 2014.

9. R.Amirtharajan, Sandeep Kumar Behera, Motamarri Abhilash Swarup, Mohamed Ashfaaq K and John Bosco Balaguru Rayappan,"Colour Guided Colour Image Steganography" Universal Journal of Computer Science and Engineering Technology 1 (1), 16-23, Oct. 2010. © 2010 UniCSE, ISSN: 2219-2158.
10. Mohammad Tanvir Parvez and Adnan Abdul-Aziz Gutub "RGB Intensity Based Variable-Bits Image Steganography" College of Computer Sciences & Engineering, 2008 IEEE Asia-Pacific Services Computing Conference.

Web site

1. UK Military Exposed to Blackmail Risk Through Lost Data http://countermeasures.trendmicro.eu/uk-military-exposed-to-blackmail-riskthrough-lost-data/
2. Applications of Steganography http://www.datahide.com/BPCSe/applications-e.html