ISSN No. 0976-5697

Volume 2, No. 1, Jan-Feb 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A High Security Approach For Image Steganography By Rapid Information Retrieval

Khosravi, Sara*
Department of Computer Engineering, Science and
Research Branch, Islamic Azad University
Khouzestan-Iran
sara_khosravi_1362@yahoo.com

Abbasi Dezfouli, Mashallah
Department of Computer Engineering, Science and
Research Branch, Islamic Azad University
Khouzestan-Iran
Abbasi_masha@yahoo.com

Yektaie, Mohammad Hossein
Faculty Member Of Islamic Azad University Abadan Branch
Abadan, Iran
Mh.yektaie@gmail.com

Abstract: There are many ways to hide information or transmission of information secretly. In this sense steganography is the best part of sending information secretly. The technology has certainly been the topic of widespread discussion among the IT community. Steganography, like watermarking and fingerprinting is a branch of science to hide information but, unlike watermarking and fingerprinting, steganogaraphy is the art of writing message or information in such a way that no one apart suitable recipient knows the meaning of the message or information. There are many techniques to perform steganography on electronic media, most notably audio and image files.

The out come of this paper is to generate a cross platform that can effectively hide a message inside a digital image file. We are presenting a technique which works by changing a few pixel color value at selected pixel in the image. We divide the image into N blocks and in each block, To increase the security level, we create distribution in selected pixels whit using strassen multiplication in each block.

Also, when retrieving a message for avoid repeating process which hidden text storage and speed up information retrieval, we save address of selected pixel. It is also, try not to degrade image quality and as far as possible does not change the image size.

Key words: Steganography, watermarking, fingerprinting, strassen multiplication

I. INTRODAUCTION

Since the rise of the Internet one of the most important factors of information technology and communication has been the security of information. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique used to implement this, is called steganography.

The word steganography is derived from the Greek words “stegos” meaning “cover” and “grafia” meaning “writing” [1] defining it as “covered writing“.

Steganography is the art of hiding the fact that communication is taking place, by hiding the information in or under information. There are different kinds of steganography used in communication channel but in digital file format the format that are more suitable are those with a high degree of redundancy. This can be defined as the bits of an object that provide accuracy far greater than necessary for the object’s use and display. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Hiding information in text is historically the most important method of steganography. An obvious method was to hide a secret message in every nth letter of every word of a text message text steganography using digital files is not used very often since text files have a very small amount of redundant data. Given the proliferation of digital images, especially on the network, and given the large amount of redundant bits present in the digital representation of an image, images are the most popular cover objects for steganography [2].

This is a project where work mainly done on BMP image based steganography. BMP image is chosen because every pixel is independent.

Steganography differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret [3]. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated [3]. The strength of steganography can thus be amplified by combining it with cryptography. Two other technologies that are closely related to steganography are watermarking and fingerprinting [4]. These technologies are mainly concerned with the protection of intellectual property, thus the algorithms have different requirements than steganography. These requirements of a good steganographic algorithm will be discussed below. In watermarking all of the instances of an object are “marked” in the same way. The kind of information hidden in objects when using watermarking is usually a signature to signify origin or ownership for the purpose of copyright protection [5]. With fingerprinting on the other hand, different, unique marks are embedded in distinct copies of the carrier object that are supplied to different customers. This enables the intellectual property owner to identify customers who break their licensing agreement by supplying the property to third parties [4]

In watermarking and fingerprinting the fact that information is hidden inside the files may be public knowledge –sometimes it may even be visible – while in steganography the imperceptibility of the information is crucial [3].

The information communicated comes in numerous forms and is used in many applications. In a large number of these applications, it is desired that the communication to be done in secrete. Such secrete communication ranges from the obvious cases of bank transfers, corporate communications and credit card purchases, on down to a large percentage of everyday emails [6].

There has been a rapid growth of interest in this subject over the last ten years and for two main reasons[7].

1. The publishing and broadcasting industries have become interested in techniques for hiding encrypted, copyright marks and serial numbers in digital films audio recordings, books and multimedia products; an appreciation of new market opportunities created by digital distribution is coupled with a fear that digital works could be too easy to copy.
2. Moves by various governments to restrict the availability of encryption services have motivated people to study methods by which private messages can be embedded in seemingly innocuous cover messages. The ease with which this can be done may be an argument against imposing restrictions.

Capacity, security and robustness, are the three main aspects affecting steganography and its usefulness [8].

- Capacity refers to the amount of data bits that can be hidden in the cover medium.
- Security relates to the ability of an eavesdropper to figure the hidden information easily.
- Robustness is concerned about the resist possibility of modifying or destroying the unseen data.

This paper will focus on hiding information in images in the next sections.

There are lots of techniques available that implement steganography on a variety of different electronic mediums.

We are using the technique depend on the digital image as digital images often have a large amount of redundant data and this is what steganography uses to hide the message.

The data can hide in the image by changing the image content i.e. by changing the color of the pixels. By this technique we can hide a large volume of data inside the image. Once implemented, it is not necessarily perceptible to a human eye that the image has been changed, but to a computer simple statistical analysis can pinpoint a changed image from original one. It is so easy for a computer to notice these changes are.

Images that are used for inserting and hiding secure data are called 'cover image' and the image where secret bits are inserted is called 'stego image'. There are many different Steganographic algorithms. Some of them are in spatial domain and others are in transform domain. LSB (Least Significant Bit) replacement steganography is a popular and simple technique that can hide message bits in LSB planes of image pixels. LSB based methods can be divided into 2 main groups: LSB replacement, which simply replaces LSB bits of cover image with secret bits, and LSB matching where pixels are randomly incremented or decremented. In contrary; steganalysis methods attempt to detect Stego-image and extract it. Inserting secret bits in image changes some statistics of image; this opens some roads to detect Stego-image. So the changes made by Steganographic are a key performance metric; lower change: more robust algorithm. It is evident that the changes in cover image are related to the volume of inserted bit, so Stego-images with higher insertion rate are detected more easily.

Steganalysis methods generally are divided in two main groups: active and passive methods. In passive methods only presence or absence of hidden data is considered, while in active methods a version of inserted data is extracted, too. Furthermore, different steganalysis methods, depending on steganography algorithms they target, can be classified in two groups: Model-based (Specific) and Universal Steganalysis.

The goal of model-based methods is attacking to Specific-Steganographic algorithm but in Universal methods attack is performed not considering any prior assumption on Steganographic algorithm and so can be used for several Steganographic algorithms [8]. Universal methods usually are preferred because of their versatility but, their performances are inferior to specific steganalysis [9],[10].

Universal methods that targets different Steganographic algorithms, usually contain two main steps: feature extraction and classification. Firstly, in extraction phase analyzer must find features that have been changed significantly as a result of hiding process and can suitably used as separating characteristics for inserted and non-inserted images. In classification phase, classifier that can be a neural network, Support Vector Machine (SVM), a similarity measure and etc, must be trained on feature vectors from both inserted and non-inserted images, which were extracted in the first step. Universal methods usually use features that are sensitive to wide variety of embedding algorithms [8]. Otherwise, they must extract features for every specific insertion algorithm separately [9].

II. CONCEPT OF IMAGE FILES

The image is combination of pixels. Each pixel shows a color and specified whit a number. Thus the computer an image is a collection of numbers that constitute different light intensities in different areas of the image. This numeric representation forms a grid and the individual points are referred to as pixels [10].

Each pixel set of multi-bit. The number of bits in a color scheme, called the bit depth, refers to the number of bits used for each pixel. The smallest bit depth in current color schemes is 8, meaning that there are 8 bits used to describe the color of each pixel. Monochrome and grey scale images use 8 bits for each pixel and are able to display 256 different colors or shades of grey. Digital color images are typically stored in 24-bit files and use the RGB color model, also known as true color [12].

All color variations for the pixels of a 24-bit image are derived from three primary colors: red, green and blue. Three colors in each pixel create the 24 bit binary number, 8 bit of it belong to red color, 8 bit belong to blue color, 8 bit belong to green color.

Thus in one given pixel, there can be 256 different quantities of red, green and blue, adding up to more than 16-million combinations, resulting in more than 16-million colors [11]. Not surprisingly the larger amount of colors that can be displayed, the larger the file size [12]. in image steganography it works by changing a few pixel color value.

III. IMAGE STEGANOGRAPHY

Images are the most popular cover objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications. For these different image file formats, different steganographic algorithm.

Image steganography techniques can be divided into two groups: those in the Image Domain and those in the Transform

Domain. Image – also known as spatial – domain techniques embed messages in the intensity of the pixels directly, while for transform – also known as frequency – domain, images are first transformed and then the message is embedded in the image [14].

Image domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation and are sometimes characterised as "simple systems". The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format [15].

Steganography in the transform domain involves the manipulation of algorithms and image transforms. These methods hide messages in more significant areas of the cover image, making it more robust [20]. Many transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression [14]. In the next sections steganographic algorithms will be explained in categories according to image file formats and the domain in which they are performed. In figure 1 is showen classification of steganography techniques

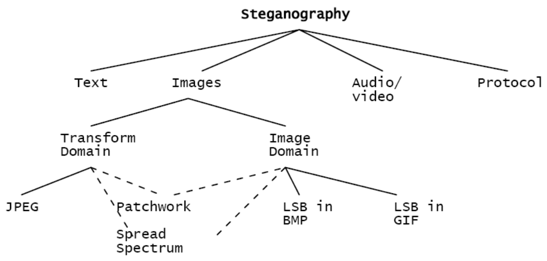


Figure 1. Classification of steganography techniques

A. *LEAST SIGNIFICANT BIT INSERTION*

Human visual system (HVS) is not sensitive to very small color variations. LSB steganography methods benefit this human eye's weakness in distinguishing small detail variations, and insert message information in least significant bits as noise. This method is exactly what it sounds like; the least significant bits of the cover-image are altered so that they form the embedded information. The following example shows how the letter A can be hidden in the first eight bytes of three pixels in a 24-bit image.

Pixels: (00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
A: 01000001
Result: (00100110 11101001 11001000)
(00100110 11001000 11101000)
(11001000 00100111 11101001)

The three underlined bits are the only three bits that were actually altered. LSB insertion requires on average that only half the bits in an image be changed. Since the 8-bit letter A only requires eight bytes to hide it in, the ninth byte of the three pixels can be used to begin hiding the next character of the hidden message.

This approach is very easy to detect. A slightly more secure system is for the sender and receiver to share a secret key that specifies only certain pixels to be changed. Should an adversary suspect that LSB steganography has been used, he has no way of knowing which pixels to target without the secret key.

In its simplest form, LSB makes use of BMP images, since they use lossless compression. Unfortunately to be able to hide a secret message inside a BMP file, one would require a very large cover image. Nowadays, BMP images of 800 × 600 pixels are not often used on the Internet and might arouse suspicion. For this reason, LSB steganography has also been developed for use with other image file formats.

IV. CURRENT WORK

A. *convert text to byte*

Data is converted into the bytes that are each character in message is converted into its ASCII equivalent.

Moreover if message is password protected, then while retrieving message, the retriever has to enter the correct password for viewing the message. For an example if we are taking the character "a" in the message then "a=" 01100001 is stored in byte array. Because ASCII value for "a" is 97 and binary equivalent is 01100001.

At 8 bit of the color number, if we change 2 least significant bit, our sighted system can detect changes in pixel. In this case, leas significant bits have 4 state, which is shown in Table 1.

Table 1

1	1	0	0
1	0	1	0

If we want to store information in 2 bit, at the worst situation, 2 bit are changed, for example if the red color number is a 10111011 pixel, and we want to store the information in 2 least significant bit, at the worst situation the red color number is change to 10111000, examinations shows that HVS can not distinguish this alteration. So we save our information into least significant bits of color.

B. *Message Embedding In Digital Image*

Hiding image involves embedding the message in to the digital image. Each pixel typically has three numbers associated with it, one each for red, green, and blue intensities, and these value often range from 0-255. In order to hide the message, And data is first converted into byte format and stored in a byte array. the message is then encrypted and then embeds each bit into the LSB position of each pixel position. It uses the first pixel to hide the length of message (number of character).Suppose, we only change the last two that determine the "one place", and the "two place". We can only alter the original pixel color value by 3degre.

We use four bytes in two pixel to store 8 bits character.
The first color in first pixel : r7 r6 r5 r4 r3 r2 r1 r0
The second color in first pixel :g7 g6 g5 g4 g3 g2 g3g2
The third color in first pixel: b7 b6 b5 b4 b3 b2 b5b4
The first color in second pixel: r7 r6 r5 r4 r3 r2 r7 r6

My character have (c7 c6 c5 c4 c3 c2 c1 c0) bits. Then we can place two

of these character bits in the lowest red pixel, tow more in the lowest green pixel, the two in the lowest blue pixel nda the two in the lowest red other pixel as follows.

The first color in first pixel: r7 r6 r5 r4 r3 r2 c1 c0
The second color in first pixel:g7 g6 g5 g4 g3 g2 c3c2
The third color in first pixel: b7 b6 b5 b4 b3 b2 c5c4
The first color in second pixel: r7 r6 r5 r4 r3 r2 c7c6

If we take an example of pixel (255, 64, 64) with character "a", then we can obtain:

Originl pixel=(11111111,01000000,0100000)
"a" = 01100001
New pixel = (11111101, 01000000,0100000)
New pixel =(253,64,64)

Here we can notice that the new pixel of (253,64,64) is almost the same value as the old pixel of (255, 64, 64). So there will not be noticeable color difference in the image.

C. Algorithm to find the pixel

We suggeste new algoritm in this paper. For more efficient and find pixel of image that have a certain complexity, we divide image to bolck n*n. To perform this operation, and to find higher intensity pixel, we put n^2 color data element of block n *n in matrix. The average color of this block obtains. The number is a boundary to determine the elements whit to implement a security level, we want to creat more scattere in selected pixels. Untill understaning the implemented algorithm would be more difficult and the discovering of information may not be possible without the algorithm. Therefore, we use matrix multiplication. Acording to Figure 2 data elements in each block are read in form of row and column and put them in two matrices.

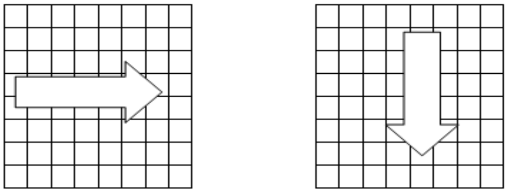
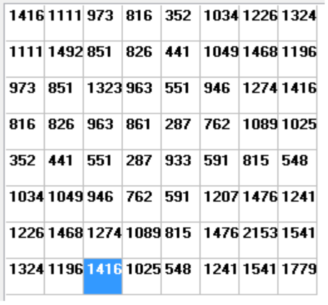


Figure 2: How to read data in each bloc

Then these two matrices are multiplied together. For this task we use strassen multiply algorithm with complexity $O(n^{2.81})$.

We examine this method to store the text "image steganogarphy" in figure 8. If N = 8, we check the informyation which are stored in the first block whit this method. If the strassen beat is implement in this block And we compute the average color and obtained average by strassen multiplying , 10 pixels are selected to store the data. We doing this algorithm for green and blue color in block. 11 percent of colors in block are changed.

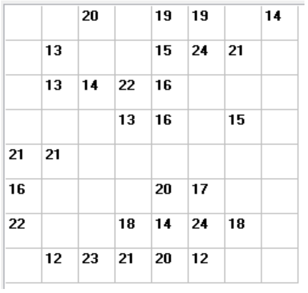
Then these two matrices are multiplied together. For this task we use strassen multiply algorithm with complexity $O(n^{2.81})$. Matrix multiplied is processed that we get elements mean the number is a measure to select elements and we choose elements larger than the mean. For example, in the case before, matrix multiplied is shown in figure3.



1416	1111	973	816	352	1034	1226	1324
1111	1492	851	826	441	1049	1468	1196
973	851	1323	963	551	946	1274	1416
816	826	963	861	287	762	1089	1025
352	441	551	287	933	591	815	548
1034	1049	946	762	591	1207	1476	1241
1226	1468	1274	1089	815	1476	2153	1541
1324	1196	1416	1025	548	1241	1541	1779

Figure3: matrix multiplied

After taking the mean of the block and selected elements of a larger average, elements that are selected is shown in figure4.



		20		19	19		14
	13			15	24	21	
	13	14	22	16			
			13	16		15	
21	21						
16				20	17		
22			18	14	24	18	
	12	23	21	20	12		

Figure4:selected pixels

In order to perform this algorithm the career picture Is shown figure 5.

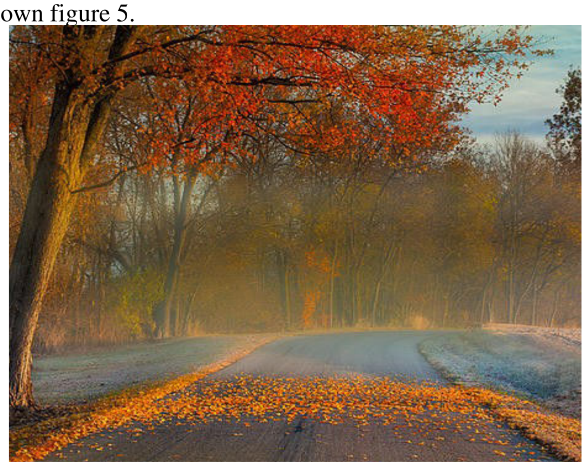


Figure 5:A career picture

In figure 6, selected pixels are greater average number are marked with black color

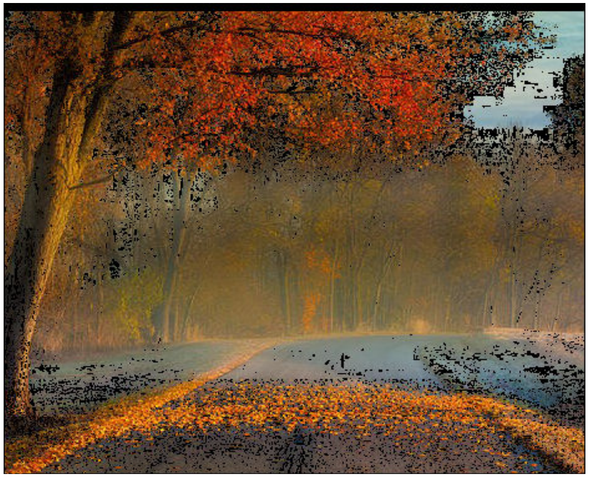


Figure 6: selected pixels whit N=8

Changes in the amount of red pixels selected in block is shown in Figure 7

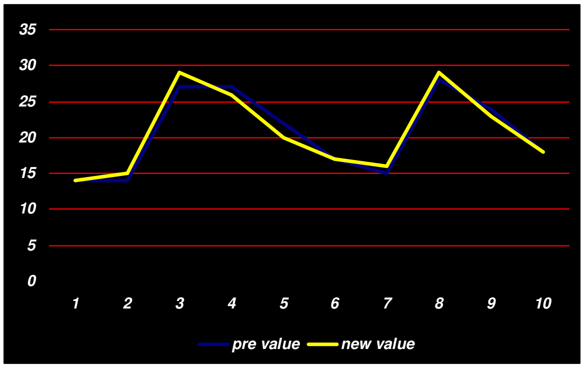


Figure 7:changes red color

Changes in the amount of green pixels selected in block is shown in Figure 8.

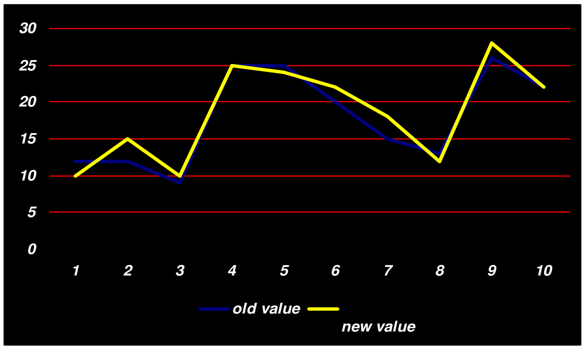


Figure 8: changes green color

Changes in the amount of blue pixels selected in block is shown in Figure 9

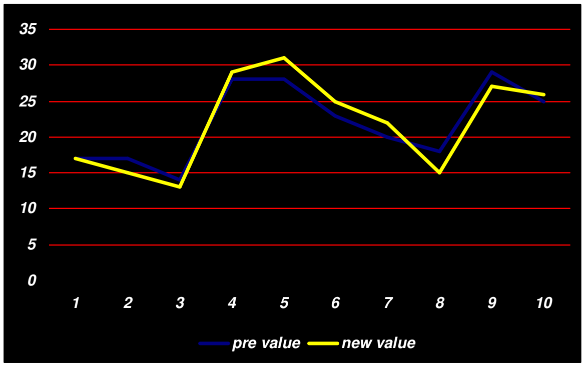


Figure 9: changes blue color

We examine this method to store the text "image steganogarphy" in figure 10. If N = 8,we check the informyation which are stored in the first block whit this method. If the strassen beat is implement in this block And we compute the average color and obtained average by strassen multiplying , 10 pixels are selected to store the data. We doing this algorithm for green and blue color in block. 11 percent of colors in block are changed.

Figure 10:Image with hide text

So, we use this algorithm for embedding the massage text

1) First, we chose the image and massage text, that we should use them on the picture.

2) We covert massage text to binary code.

3) Image divided into n blocks.

4) Data elements in each block are read in form of row and column and put them in two matrices. Two matrix multiply whit strassen Multiplying and determinde the average matrix. Elements are greater average, are selected.

5) We estimate the least significant bits in marked pixels.

8) Embed the text into the LSB.

D. *Retrive Maessage*

In this section we will discuss the retrieving the message from the image independent of the file format. Once a message has been retrieved it has to be converted in to the original message. This process can be done by reading the embedded data from the file. The read data will be in bytes format. This can be done by extract the selected pixels of output image in one array. Decoding in same manner as the reversal of encoding process i.e. first pixel value gives number of character in the message. After every pixel gives the message character’s ASCII value, which then stored in byte array.

To presente the stored information in the image, we use this algorithm.

1) First, we chose the image, that the text embedded into it.

2) We retrieve the LSB.

3) We combine 8 bit and convert them into one character

Retrieve text stored in the image is done in two main stages.

✓ The first step to find pixel that information is stored in them

✓ The second phase includes putting together bits and retrieve the text is hidden

In the first step to find the pixel, we must to choose selected pixels.

As in the HIOP algorithm referred, first, we divide the image into blocks (n*n), and then the average color of each block in three dimensions (red, green, blue) obtain and we create two matrices, the elements of each block read row by row and column by column and put in each matrix and using Strassen's matrix multiplication, these two matrices have the multiply, we obtain the average matrix’s elements multiplied

Now, we choose elements that is larger than average color of blocks and the average result of matrix multiplication.

Now we should note that we have access to the selected pixel of Row or Column.

After access to the pixel should note that we have LSB bits, how characters are stored in bits; if two bits character are C0 and C1 and tow bit LSB are L0 and L1, two modes for data storage are:

1) C0 in L0 and C1 in L1

2) C0 in L1 and C1 in L0

We must also consider the character bits from place more value to place less value or contrast.

After considering, how to access pixels, LSB bits and how to save text characters, we start extracting LSB bits and we put them in a buffer and using the bits that are placed in buffer, hidden text obtained.

There are two methods for this

1. A buffer length of 8 bits to create, each 8-bit LSB of pixels selected to become characters and we put together the characters and get the hidden text.
2. A buffer of length count bits hidden text created, we put all bits LSB of pixels selected in buffer. Each 8-bit of buffer to become characters and we put together the characters and get the hidden text.

These methods are correct, in the first method, the length of buffer is fixed and reading process bits and process to obtain the text are concurrently, but in the second method, the length of buffer is not fixed and depends on the length of the text and firstly, reading process bits is performed, then hide text is obtained.

In general, how to obtain secret information, the processes in how to store text is used, as obtaining the mean color, consists of two matrices, beat them, taking the mean of the matrix multiplication and pixel selection is repeated and addition to these steps, bits obtained during the process are converted to text. This is causing repeated processes and is reduced speed.

One way to avoid repeating process and increasing the speed of storage, we save address of selected pixel to save. Each pixel is used for storing images is three characteristics:

1. What color of the pixels are selected for storage
2. Pixel columns
3. Pixel row

In this case, we must put address of selected pixels in specific location in the image. Usually, changes in the initial row are not cause attention, therefore, the initial row pixels to save the pixel address have selected. In each pixel we have three different colors, so to specify the type of color we need two bits. Each pixel is 24 bits, as shown in figure 3-25, two bits to determine the pixel color, 11 bits to determine column of selected pixel and 11 bits to determine row of selected pixel. How to save the address in pixel is showmen in figure 12.

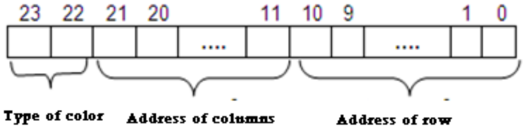


Figure 12-25:How to save the address in pixel

Since we need 4 bytes for each character, so to save the text whit length N characters, we reserve N*4 initial pixel to save address of selected pixel and we do block scheme from the last row of the data of address is located and each pixel is selected, we save its data in pixels reservation.

In this case, when you get the hide text in the image, according to the pixel address, easily, we obtain hide text in image.

E. *Conclusion*

As the result we can find the out come of the paper is to create across platform that can effectively hide a message inside a digital image file. As there are many application of image steganography like it allows for two parties to communicate secretly and covertly.

One of the other main uses for image steganography is for the transportation of high level or top secret documents between international governments also it allows for copyright protection on digital files using the message as a digital watermark. Image steganography has many legitimate uses as it can be used by hackers to send viruses and Trojans to compromise machines. So in conclusion, as more emphasis is placed on the areas of copyright protection, privacy protection, and surveillance, we believe that steganography will continue to grow in importance as a protection mechanism.

This paper has investigated whether taking the image as the cover into account increases the security of the message by creating cross platform self evaluating tool. Also describe the benefits from the approach like the security of message increases and distortion rate has reduced.

IV.REFERENCE

[1] Moerland, T., “Steganography and Steganalysis”, Leiden Institute of Advanced Computing Science,www.liacs.nl/home/ tmoerl/privtech.pdf

[2] Saurabh Singh, Gaurav Agarwal," Use of image to secure text message with the help of LSB replacement",Invertis Institute of Engineering and Technology, Bareilly, India,2010

[3] Wang, H & Wang, S, “Cyber warfare: Steganography vs. Steganalysis”, Communications of the ACM,47:10, October 2004

[4] Anderson, R.J. & Petitcolas, F.A.P., “On the limits of steganography”, IEEE Journal of selected Areas in Communications, May 1998

[5] Marvel, L.M., Boncelet Jr., C.G. & Retter, C., “Spread Spectrum Steganography”, IEEE Transactions on image processing, 8:08, 1999

[6] Image Steganography by Mapping Pixels to Letters, Mohammed A.F. Al-Husainy Department of Computer Science, Faculty of Sciences and IT, Al-Zaytoonah University of Jordan ,2009

[7] Wolfgang, R.B. and E.J. Delp, 1996. Watermark for digital images. Proceeding of the IEEE International Conference on Image Processing,Sep. 16-19, IEEE Computer Society, Washington DC., USA., pp: 219-222. DOI

[8] Chen, B. and G.W. Wornell, 2001. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding.IEEE Trans. Inform. Theor., 47: 1423-1443. DOI:10.1109/18.923725

[9] Mehdi Kharrazi, Husrev T. Sencar, Nasir Memon,” Performance study of common image steganography and steganalysis techniques”. Journal of Electronic Imaging 15(4), 041104(Oct–Dec 2006)

[10] Harmsen, J.J., Pearlman, W.A; "Steganalysis of additive noise modelable information hiding".Rensselaer Polytechnic Institute, Troy, New York, May 2003.

[11] Johnson, N.F. & Jajodia, S., “Exploring Steganography: Seeing the Unseen”, Computer Journal,February 1998

[12] Owens, M., “A discussion of covert channels and steganography”, SANS Institute, 2002

[13] “Reference guide: Graphics Technical Options and Decisions”,

[14] Lee, Y.K. & Chen, L.H., “High capacity image steganographic model”, *Visual Image SignalProcessing*, 147:03, June 2000

[15] Venkatraman, S., Abraham, A. & Paprzycki, M., “Significance of Steganography on Data Security”, Proceedings of the International Conference on Information Technology: Coding and Computing, 2004

[16] Wang, H & Wang, S, “Cyber warfare: Steganography vs. Steganalysis”, *Communications of the ACM*, 47:10, October 2004