Contemporary Engineering Sciences, Vol. 4, 2011, no. 7, 341 - 352

Difference Realization in Image Steganography

Mohammed Nasser Al-Turfi

Department of Computer and Software Engineering
Al-Mustansiriyha University, Baghdad, Iraq
mohammed_alturfi@yahoo.com

Abstract

The Steganography of images have been applied to propose a smooth efficient way depending upon the difference between the source and the cover image where the difference is to be hidden in the cover image. An algorithm has been built and designed on MATLAB7 to achieve different types and sizes of images where the reconstruction rate was 100 % and the data transferred difference was less that 15% hence image distortion was less than 0.05%

Keywords: Steganography, Data Hiding, Difference Vector.

I Introduction

Image Steganography is one of the most important branches that take its place in the modern science due to its efficiency in hiding data from intruders and hackers where data must be hidden in a very efficient, easy, non-expensive way. Therefore such ways must be available to achieve such tasks, and transmitting information using apparently innocent carrier without expose any suspicion. One of the most important principles in system and data security is how to achieve data hiding in a way that can

give maximum security, minimum cost, and can achieve best accuracy and reconstruction in a relatively less processing time. [1]

The LSB was one of the most effective ways in early projects where source image is stored in the cover image by transferring the image into bits and replace the LSB of the cover image by the bits of source image. Therefore the cover image must be at least 8 times larger than the source image (assuming grey image with 256 brightness levels) and the distortion level will be noticed if we use the Higher Bits. Some researches took other ways by using some convolutional techniques or key codes applied on the source image to do massive changes and then store it in the LSB of the source image where this leads to make the process more secured "and it will be more effective if it's used with large colored images" but more complex. [2]

Others mix between the two ways mentioned above with a new way for storing by using a special selective random coding function that store the data in a random way in the LSB in order to avoid the easiness to encode the relatively sorted data. In the later projects transforms like DCT & DWT with some of the previous applied techniques are used specially with watermarking needs to achieve higher security for the image embedded to guarantee large areas of uniform color, Internal structure of the image and providing maximum likelihood for the used fonts to avoid distortion as much as possible.[3]

In the present paper a new technique of evaluating the difference between the cover and the source image have been applied through the transferring difference between the cover and the source image instead of transferring the whole source image. Thus, the technique will lead to make the change as small as possible and hence reducing the rate of distortion which is more secured.

II Basic Concepts of Image Steganography

Steganography is the art of concealing information in ways that prevent the detection of hidden messages. Steganography include an array of secret communication methods that hide the message from being seen or discovered. Thus, hiding technique has recently become important in a number of application areas

especially in DSP categories, processing's must have certain structures and conveys properties that give the ability to avoid errors and to fight noise, therefore image steganography must satisfy the following:

1) **Robustness**

The ability to extract hidden information after common image processing operations like linear and nonlinear filters, loss compression, contrast adjustment, re-coloring, re-sampling, scaling, rotation, noise adding, cropping, printing / copying / scanning, D/A and A/D conversion, pixel permutation in small neighborhood, color quantization (as in palette images), skipping rows / columns, adding rows / columns, frame swapping, frame averaging (temporal averaging), etc. [4].

2) **Un-detect ability**

The ability to detect the presence does not automatically imply to read the hidden message. Un-detectability should not be mistaken for invisibility "a concept related to human perception"[4].

3) **Invisibility**

This concept is based on the properties of the human visual system (which is our concern in this paper) or the human audio system which depends on a very high sensitive instruments and it gets to its perfection if and only if the distortion rate is zero (which is tried to be reached but its un-reachable) [5].

4) **Security**

The embedded information cannot be removed beyond reliable detection by targeted attacks based on a full knowledge of the embedding algorithm and the detector (except a secret key), and the knowledge of at least one carrier with hidden message. Therefore special algorithms must be applied in order to increase the ambiguity and data imbroglio in a less consuming processing time and hence less processing power [5].

III Proposed Algorithm

Steganography process must pass through the steps shown in figure (1) below where we are in need for a source image larger than the cover image so that we can hide the part in the all. In early ones the cover image must be much larger than the cover image which is no longer needed. [6]

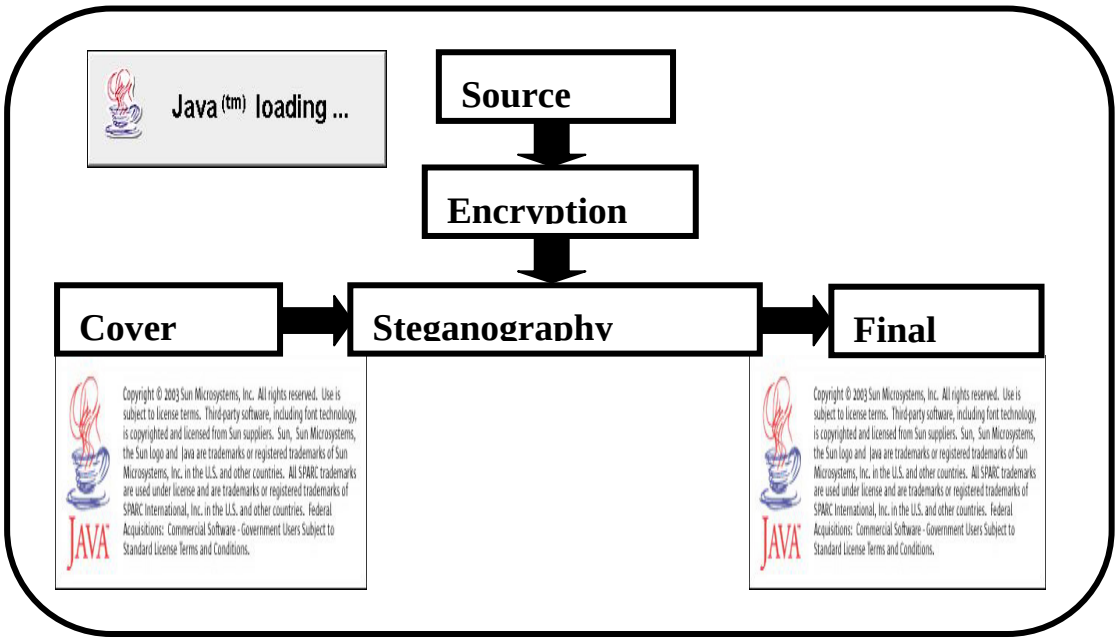

Figure (1): Image steganography process flow

Two major parts are represented by the Encryption Unit and the Steganography Algorithm. The Encryption unit is the parts that is responsible for implementing the way of hiding the data and make the way of covering the data as hard as possible. [6]

The proposed algorithm is the difference algorithm which depends upon the changes between the source and the cover image string of bits taking the whole source image in the consideration (not only the LSB) therefore one of the most important properties of this algorithm are:

First, it can handle a cover image nearly in the same size of the source image which means that large source image needs large cover image but not very large to handle the task this leads to keep the system resources and save the channel capacity.

Second, the algorithm will not jeopardize the cover image by sending it inside the source image; instead the algorithm will send the difference between the source and the cover image, which means,' and from first principle that we will not use the source image directly'.

Third, the algorithm bury the difference data inside the image using a special selective random coding function where the places are selected in such a way that minimize the distortion by selecting the places that contains information maximize the likelihood between the difference bits and the original cover image bits.

Fourth, the algorithm guarantee perfect reconstruction since the algorithm use algebraic functions so the results have no significant figures like some mathematical functions (for example sine and cosine) and hence the error produced by the processes in the forward (Encryption) and backward (Decryption) phases are eliminated.

IV Algorithm Implementation

The algorithm falls into two phases: - Forward phase (where the cover image hide the source image) and the Backward phase (where we can extract the source image from the received image)

1) Forward Phase

The system must pass through the following steps to achieve data hiding where figure (2) below shows the flow chart of the process explained as given:-

a) Data Acquisition: - Both the source and the cover image are acquired side by side with their extensions and their sizes since they are represented each one as a matrix.

b) Image Transfer: - In this step the system will represent each matrix as a vector where each vector length depend upon the size, type, color, and hence the number of bits per pixel.

c) Bit Difference: -The XOR logical function holds this job where the difference between the source and the cover vector gets one else gets zero.

d) Randomization: - the MOD function is used in order to achieve data hiding. For example if we have the data "a b c d e f g h i j "so by using Mod Function shifted by 3 then the output will be "a e i c g b f j d h" in the forward implementation or "a h e d i j c f g b" if reverse implementation is applied.

e) Re-arranging: - The randomized data is now ready to be hidden inside the cover image where we will create the new cover image that contains the embedded data with a new size and extension and representations.

2) Backward Phase

After hiding the data in the forward phase and the image being transferred the image will be received on the other side of the system and it contains the important information which must be extracted as accurate as fast as possible therefore the image must pass through the process explained as given below:-

a) Data Acquisition: - Both the received image and a similar copy of the cover image must be exist in order to have a reference that we must compare with and to identify the positions that we made changes in.

b) Image Transfer: - In this step the system will represent each matrix as a vector where both vectors will be of the same length (the addition in the size of the received one more than the original one represents the source image dimensions) so that we can achieve the comparison process one to one correspondence.

c) Bit Difference: -Since the XOR function achieves the job in the forward phase so the position in the received vector contains one then the real value is the opposite of the one in the real image. While if the received vector contains zero then the value is the same as the one in the real vector.

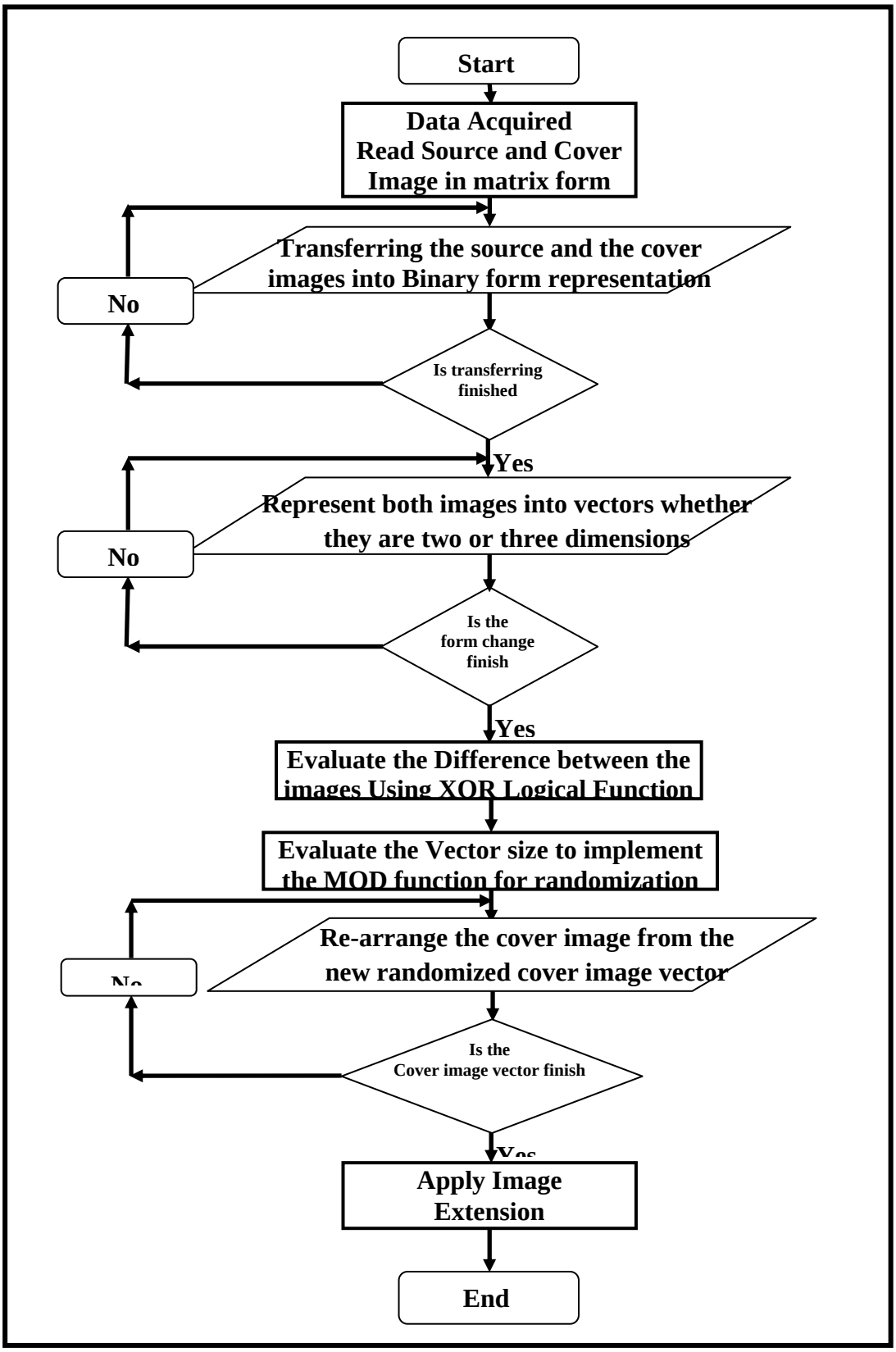

Figure (2):- Shows flow chart of the forward phase

d) Randomization: - in order to retrieve the same order of the data as the original one first we must identify the places that we had already made changes in, therefore we will use the same MOD function because this function will identify the real position. For example the function will produce the sequence "1 8 5 4 9 10 3 6 7 2" where they refer to the positions that occupied by the data after randomization. So we find that 2 is at the end of the sequence which means that the last symbol in the received sequence is t second one in the original sequence and so on.

e) Re-arranging: - By applying the traditional sorting process the system get its original vector sequence which represents the source image. In this step we are in need for the original dimensions of the source image so that we can transfer the vector into an image again where the system put these dimensions at the end of the difference vector on the transferred image where the system extract them to create the original image. For the image extension problem it is solved by applying that the source and the cover images must be of the same one.

As it can be seen that the processes of the forward and the backward phases are the same and the main parts of both phases are the same except some small details that depends upon the application itself and its requirements; which stats the easiness of the system mentioned above and hence the smoothness of the implementation environment which needs no sophisticated tools.

V Results and Conclusions

For the cover image (mnh5.jpg) shown its size (128*128*3) with maximum values of (176, 94, 60) and minimum values of (59, 21, 0) (each value represent the max or min value obtained in the Red-Green-Blue Matrixes).

For the source image (mnh3.jpg); its size (100*100*3) with maximum values of (255, 255, 255) and minimum values of (0, 0, 0). Where the difference between the two images is very clear and the size difference is not large with respect to their sizes.

The size of the vector of the cover image is 393216 while the size of the vector of the source image is 240000 so the cover is 1.64 times larger than the source image. By comparing both vectors for the same correspondence the number of one's (which means that there is a difference since we are using XOR logical function) that appears in the difference vector was 23387 which means that the rate was 9.75%. But when the difference vector is embedded inside the cover image the distortion rate was 0.0253% (the distortion measured by summing the weights of the different bits produced from the change process and divide it by the overall summations of weights of all bits).Figure (3) shows the new image production due to data hiding process.

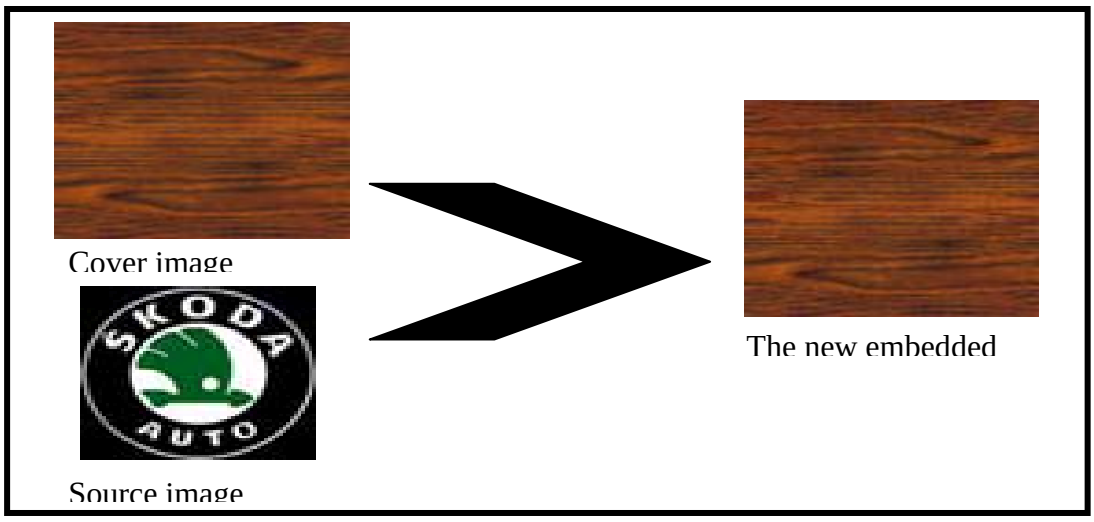

Figure (3): - The new image production from the forward phase

In the receiving side the backward phase will start by first converting the received image into a binary vector in order to prepare this vector for the comparison with the vector produced from the cover image saved in the received side as shown in figure (4-a). To locate the places that suffer from changes occurred through the processes in the forward side, this task will be applied by the XOR logical function as shown in figure (4-b) where the places with ones refers to changes while zeros refers to none where we can gather the data of the source image. Figure (4-c) shows the rearrangement for bits to retrieve its original state by applying MOD function where the original source image will appear again.

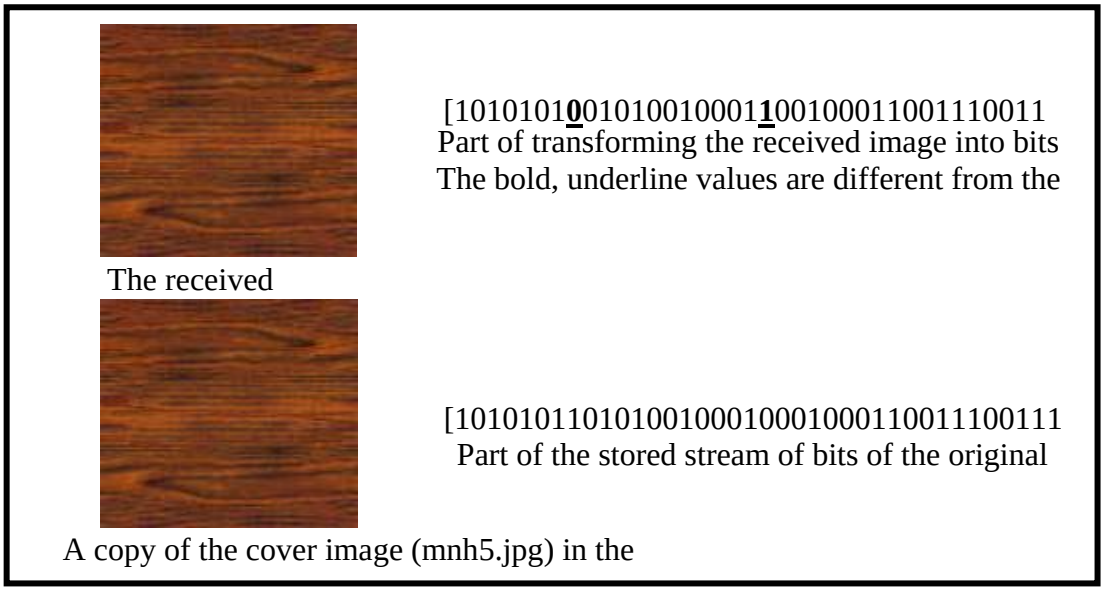

Figure (4-a): - The process of converting the image into a binary

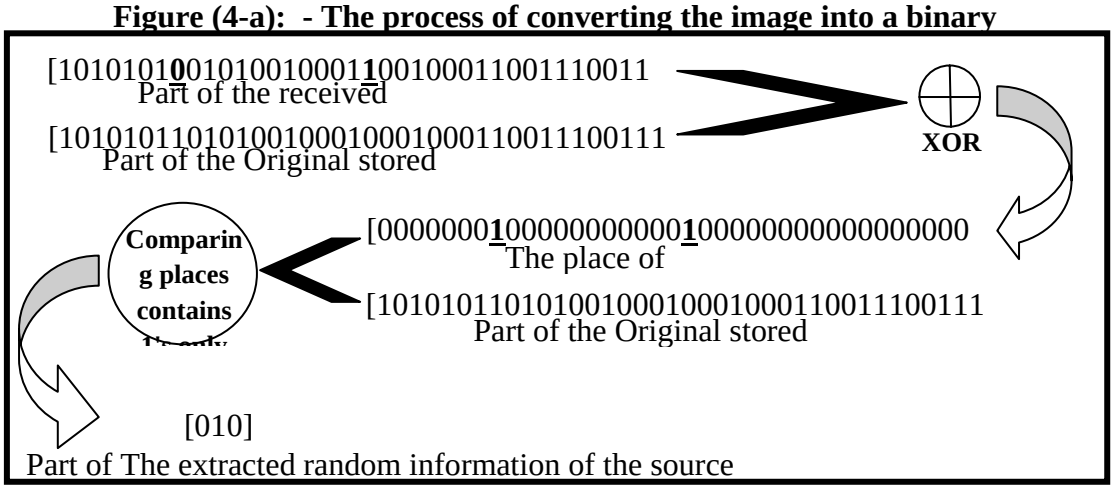

Figure (4-b): - The process of extracting the important information from the

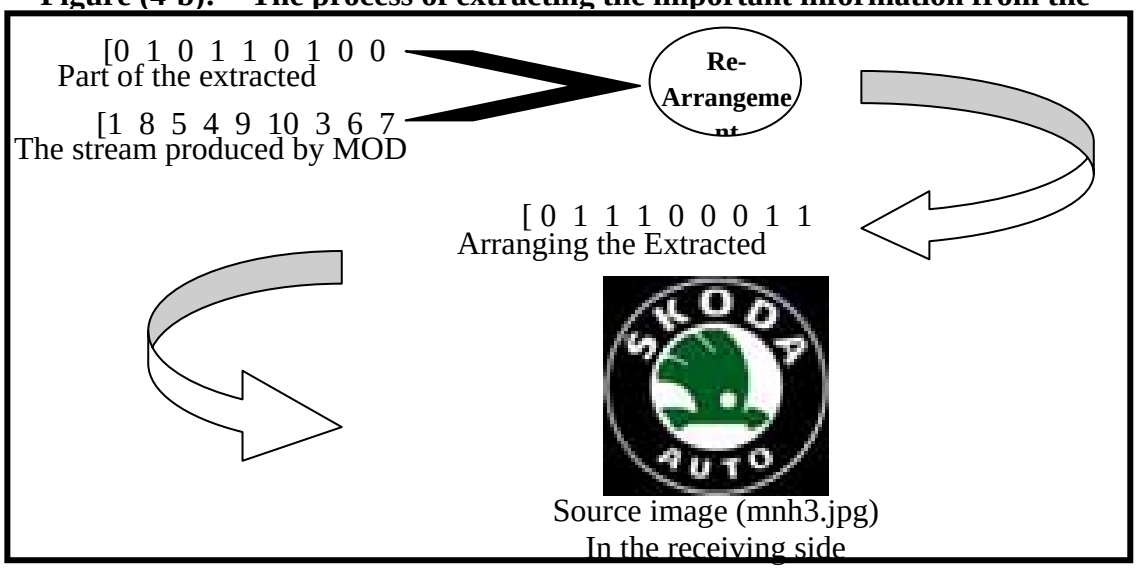

Figure (4-c): - The process of rearrangement to retrieve the source image

VI Conclusions

Some results appear very clear for images that have been applied for certain events. The cover and the source images have a close correlation factor for each data and can be hidden in a more efficient amount of information to each of a hide information inside due to rate of change will be distributed to all parts of the image nearly in the same rate therefore it will appear that no change occur. The images (mnh3.jpg) with (mnh2.jpg) and images (mnh4.jpg) with (mnh5.jpg) are very good examples of that situation as shown in figure (5) below.

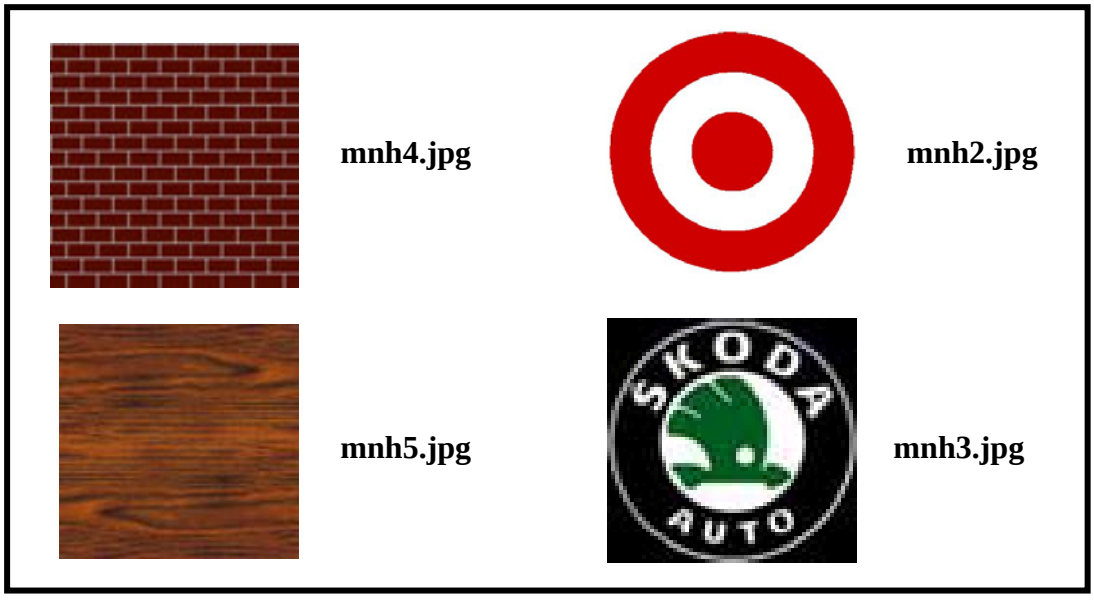

Figure (5): - Images that have a close value of correlation factors

Images that have similarity in their areas of colors, fonts, and rates of hiding are very high, with a noticeable decrimentation in the rate of distortion due to closeness of their correlation factors which is produced from the minimum amount of differences which can be presented in the figure 6 below.

Figure (6): - Images that have the same areas of design having better data hiding

References

1) Jessica Fridrich & SUNY Binghamton, Binghamton, NY 13902-6000, Center for Intelligent Systems U.S.A and Mission Research Corporation 1720 Randolph Rd. SE, Albuquerque, NM 87105, U.S.A, "Applications of Data Hiding in Digital Images" The ISSPA'2009, Brisbane, Australia August 22-25, 2009.

2) Irby Thompsonc & Mathew Monreo, "FragsFS: An Advanced Data Hiding Technique" , ATRC- Lockheed Martin, Blackhead Fediral, USA, January 2006

3) John E. Gillbert , "Steganography: Theory and Applications on FPGA" , September 2003.

4) Dusan Levicky, Emil Matus, Peter Kral , " Steganography Using Wavelet Transform analysis-synthesis-algorithms" , IEE 1996 Vol. 47, p.p281-286.

5) Vidyasagar M. Potdar, Song Han, Elizabith Chang "A Survey of Digital Image Watermarking Techniques" School of information system, Curtin University of Technology, Perth, Western Australia 2005.

6) *Peter FORIŠ*- XONOS Slovakia s.r.o., Štúrova 27, 040 01 Košice, Slovak Republic, *Dušan LEVICKÝ*- Dept. of Electronics and Multimedia Communications, Technical University of Košice, Park Komenského 13, 041 20 Košice, Slovak Republic "Adaptive Digital Image Watermarking Based on Combination of HVS Models", Radio engineering, Vol. 18, No. 3, September 2009.

Received: September, 2011