Text Realization Image Steganography

Dr. Mohammed Nasser Hussein Al-Turfi *mohammed_alturfi@yahoo.com*
Department of Computer and Software Engineering
Al-Mustansiriyah University
Baghdad-Iraq

Abstract

In this paper the steganography strategy is going to be implemented but in a different way from a different scope since the important data will neither be hidden in an image nor transferred through the communication channel inside an image, but on the contrary, a well known image will be used exists on both sides of the channel and a text message contains important data will be transmitted. With the suitable operations, we can re-mix and re-make the source image.

MATLAB7 is the program where the algorithm implemented on it, where the algorithm shows high ability for achieving the task to different type and size of images. Perfect reconstruction was achieved on the receiving side. But the most interesting is that the algorithm that deals with secured image transmission transmits no images at all.

Key Words: Steganography, Text Stream, Secured Data

1. INTRODUCTION

The art of hiding information inside an image is called image steganography, where the ability of hiding information inside this image without noticeable distortion depends on the hiding algorithm and this decides the degree of successfulness [1]. The degree of successfulness may be specified by the amount of security and hardness that must be paid to extract features and important data. [9]

Steganography is a Greek word of two syllables, "stego" means "cover" while "grafia" means "writing" define it's a "cover writing" [10], therefore text file is the first impression about the meaning of the word. Hiding data in a text file is more difficult than hiding data in an image because it might change the text meaning or format or both [11]. Therefore two ways where followed either by not changing text meaning or by creating a rubbished, un-understandable, symbolic file keeping the file format. [12,14]

Early works depends upon using the Least Significant Bits (LSB) of the cover image for implementing data hiding since its effect is the minimum and the size effect is equi-probabile to other bits. Then certain operations are implemented on the source data in order to increase the rate of securing the data and then to be inserted inside the cover image and mostly in the LSB since the distortion rate will rise if the data inserted inside other bits. [2, 3]

DSP and convolutional techniques are used now a day for implementing image steganography to increase the rate of security and to reduce the rate of distortion but of course all these complications consume time, efforts, and processing powers. [2, 4]

Using some special functions that randomize the way of choosing the position was one of the alternatives where the processed data will be stored in. This will enlarge the plain of searching inside the cover image and reduces the rate of distortion since there is no linearity in the process especially if the image is RGB because at least this will triple the efforts. [3][5]

Some use the pixel-value differences in their researches inorder to increase the amount of embedded information where an Optimal Pixel Adjustment Process (OPAP) is used to enhance the stego-image quality. The confidential information can be extracted from the stego-image without the assistance of the original image [4].

Others search for certain places inside the image like a 2*2 block of pixels with high contrast where message bits can be embedded into these blocks. Mathematical functions like MOD-4 and a coding key is used to increase the amount of embedded data while maintaining the data fidelity and the process is as easy as possible since the process applied in the data hiding phase is the same as the one in the data extraction phase[10].

Some researches didn't apply the process from the computer point of view but from the communication point of view by applying Spread Spectrum Image Steganography (SSIS). SSIS conceals a message of substantial length with in digital imagery while maintaining the original image size and dynamic range. The hidden message can be recovered using the appropriate keys without any knowledge of the original image [13].

In this paper a new algorithm is proposed, the algorithm chooses the best cover image size exists on the system data base that matches the size of the source image. The difference between the cover and the source image is evaluated and transformed to text, then the text will be transferred as a text message, an E-Mail, or even as an SMS where this make the change as small as possible, hence reduce the need for a sophisticated channel and is more secured since the source image is not transferred at any way.

2. IMAGE VERSES TEXT

One of the most important principles that must be obeyed in image steganography is that the cover image must be larger in size than the source image so that we can hide the small inside the large where the larger cover the best secured data transmission and less distortion but this will be in need for larger, faster, and more powerful channel which means more cost and complex electronic circuitry, this will be insistence if the used images are colored. [4, 6]

Images may fall in different types depending upon their extensions, but in general they contains pixels, their representation starts from 8-bits of brightness in Black & White images (Gray Scale) where the brightness level may reaches up to 32-bits especially in true color image applications like Photoshop's and rendering (as in 3-D Max).[5]

Texts are quite different, they may fall in different type of files but the text itself inside these files is the same since the way of implementation (writing) is the same where here the ASCII code is the bible book and the representation is fixed given by 7-bits only and the variety is given by different re-arrangements but not by different types. [6, 7]

The important property that exists on text but not in images is that texts need no operational function for decoding while for images, there must be an operational de-coding and de-compression function depending upon the image extension "jpeg, tiff, gif, …etc" and the way of de-compression and operation "played with fox viewer or on media player, …etc" where this means that dealing with text is much easier than dealing with images. [6, 7]

In this paper we are going to use text to represent the difference between the source image and a pre-notified image stored and indexed in the system data-base exists on both sides of the channel where no noise will affect the images. The important useful data is delivered in different type and may be in different way " for example if the application is not urgent or channels are occupied or not secured then the difference may be sent in a message written ;where different types of watermarking and letter security algorithms may be applied; through the post office or by the person himself". [8]

3. PROPOSED ALGORITHM

One of the most important features of the proposed algorithm in this paper is that the cover image may be equal in size to the source image where data security and the hardness for extracting the important data maintained. Second, the algorithm doesn't transmit any image at all which is very important feature because the image stands for millions of words and any intrusion or system breaking may cause security alert.

Third, the easiness that stand for transporting the vital data from one end to the other by transmitting through the secured channel, or by mobile equipments like cellular mobiles or

PDA's as an SMS, or by letters either written and entered by hands or written and entered using OCR's.

Perfect reconstruction that appears at the receiving ends which is the fourth important property, since the system can stand strong and fight noise well in an easy ways and not in need for sophisticated equipments which rises the overall system cost.

4. ALGORITHM IMPLEMENTATION

The proposed algorithm may fall into two parts: - Forward (where the process applied at the transmitting end. The difference between source and cover image is generated), and Backward (Re-arrange the received text to extract the source image from the received text).

4.1 Forward Phase

The system must pass through the following steps to achieve data hiding where fig(1) shows the flow chart of the process explained as given:-

a) Image Comparative: - The cover image must be selected from the system data-base of images in such a way that the selected one must be as near as possible to the size of the source image so that we can maintain the channel resources. An important note must be taken in consideration is that it's not necessary they have the same extension or properties since both images are going to be changed to matrices but it's preferred that they are from the same color type (Both are RGB or Black & White).

b) Image Transforming: - In this step the algorithm will read the image using **imread(name)** function and transform each image into a matrix and each matrix into a vector where both vectors must be unified in length by adding zeros to the shorter one.

c) Difference Evaluation: -This step is performed by the ordinary mathematical subtraction. The difference is stored in a difference vector; its length equal to the larger one. An important tag is added for each operation performed which is very necessary at the receiving end to hold perfect reconstruction. If the operation output is positive then the tag of the operation is (0) and (1) if the output is negative.

d) Bit Traverse: - This part of the system is responsible for transforming the difference vector (which consists of 8-bits of brightness) from decimal to binary values in order to prepare these values to be changed into letters or texts (consists of 7-bits ASCII code).

e) Re-arranging: - This part of the algorithm is responsible for text generation where the system will begin to take seven bits at a time and transfer them into a letter (because the image pixels are represented in 8-bits) by applying the **char(number)** function. For example char (65) is "A" while char (97) is "a".

f) File Creation:- The file to be transferred is created and arranged as the following:-

1- The file is created for an open and append operations while the extension of the text file must be chosen (for example "txt", "doc" …etc).

2- The first part of contents of the file up to the first coma (,) represents the cover image index number (for example "173," this means the used cover image is numbered 173).

3- From the first to the second coma the data represents the image size (for example the image dimensions are 80*60*3).

4- From the second to the third coma the stream of (0's) and (1's) represents the status of the output of the subtraction operations where the number of bits in this stream is equal to the number of subtractions and is equal to the multiplication of the image dimensions (80*60*3=14400).

5- From the third coma to the EOF (End Of File) the text represents the difference between the pixels of the source and the cover images which is transformed to text.

At this point we can notify two things; the first is that the source image is not exchanged between the ends in any way or any how because the algorithm can extract the source image using the text file only and depending on the index value given which represents the cover image index. Second is that the text file and the source image are nearly of the same size which means no extra size is needed for the cover image as in the traditional steganography.

4.2 Backward Phase

At the receiving end the algorithm is in need for a text message contains the difference between the source and the cover images as a text, no matter how the letter is received

whether this letter may be an E-Mail, Secured encrypted message through secured channel, a letter entered by hand delivered by the post mail or even by a letter printed by a printer and entered using OCR scanner program, the letter must pass through the following steps:-

a) When the file is received the algorithm will start to read the data until it reaches the first coma (,) where this part represents the index number of the image used from the system images data base. At this point the stored vector (which represents the specified indexed image that is transformed from an image into a vector) is located and is ready to be used for comparison process.
b) The data stored between the 1st & 2nd coma represents the dimensions of the source image which is very important for image reassembling.
c) The data stored between the 2nd & 3rd coma (,) represents a stream of (0's) & (1's) which specifies the status of the difference operation where the number of bits is equal to the number of subtraction operations implemented and equal to the number of pixels in it.
d) Starting from the 3rd coma (,) to the EOF the text will be transferred to start processing which include:-

1- Reading the text and transfer it in a vector, letter by letter or symbol by symbol.
2- Apply the **abs('letter')** function to transfer the text in the vector to decimal values.
3- Transfer the decimal values in the text vector to binary values made of 7-bits.
4- Re-arrange the new train of bits into 8-bits values and convert it into new decimal values.

e) Applying the correspondence algebraic operations to re-create the source image from the cover image where if the status bit is (0) means the subtraction output is positive and the operation to be implemented now is subtraction to retrieve the original data. While if the status bit is (1) then the output is negative and the operation to be implemented is addition to retrieve original data.
f) The new decimal values obtained are arranged in a pre-defined matrix dimension where the function **imwrite(a,filename,fmt)** writes the matrix "a" of dimensions similar to the dimensions of the image to the file specified by "filename" in the format specified by "fmt". Matrix "a" can be an M-by-N (grayscale image) or M-by-N-by-3 (color image) array. If the format specified is "tiff" it can also accept an M-by-N-by-4 array containing color data that uses the CMYK color space.

5. RESULTS

Fig (2) clarifies parts of the file that will appear. The 1st line starts with the value 173 which represents the index code number of the cover image in the system image database that leads to the vector represents that cover image. In this vector the cover image is stored in a vector ready for subtraction to reduce the processing time and to increase the system speed.

Next value after the 1st coma represents the source image dimension which is important to prepare a matrix with these dimensions to store the results from the subtraction process between the received values and the stored one in the recieving side and to create the status vector, its size equal to the multiplication of the matrix dimension (as in our example 80*60*3=.14400) to store status bits.

After the 2nd coma the status bits will be stored in the status vector, each bit in a separate cell to decide whether the operation to be implemented between the received and the stored values are addition or subtraction.

The difference between the cover and the source image is stored as text after the 3rd coma. Part of the reconstruction process is demonstrated in fig (3). The demonstration shows how the real values are extracted from the received text applied on 20 characters as an example for the process flow.

The reconstruction was perfect; distortion rate was 0%, the process very fast because the algorithm is not complex especially at the receiving end where the cover images on both sides are transferred into vectors stored on the system data base ready to implement the subtraction process immediately.

From the brief shown in fig (2) it can be noticed that the status bits have high redundancy where a compression technique may be applied to reduce size, while a randomization

function may be added to randomize the sequence of data for mixing both status bits with difference texts to increase file security, on the same way data hiding or coding may be applied to cover the index, image size, …etc.

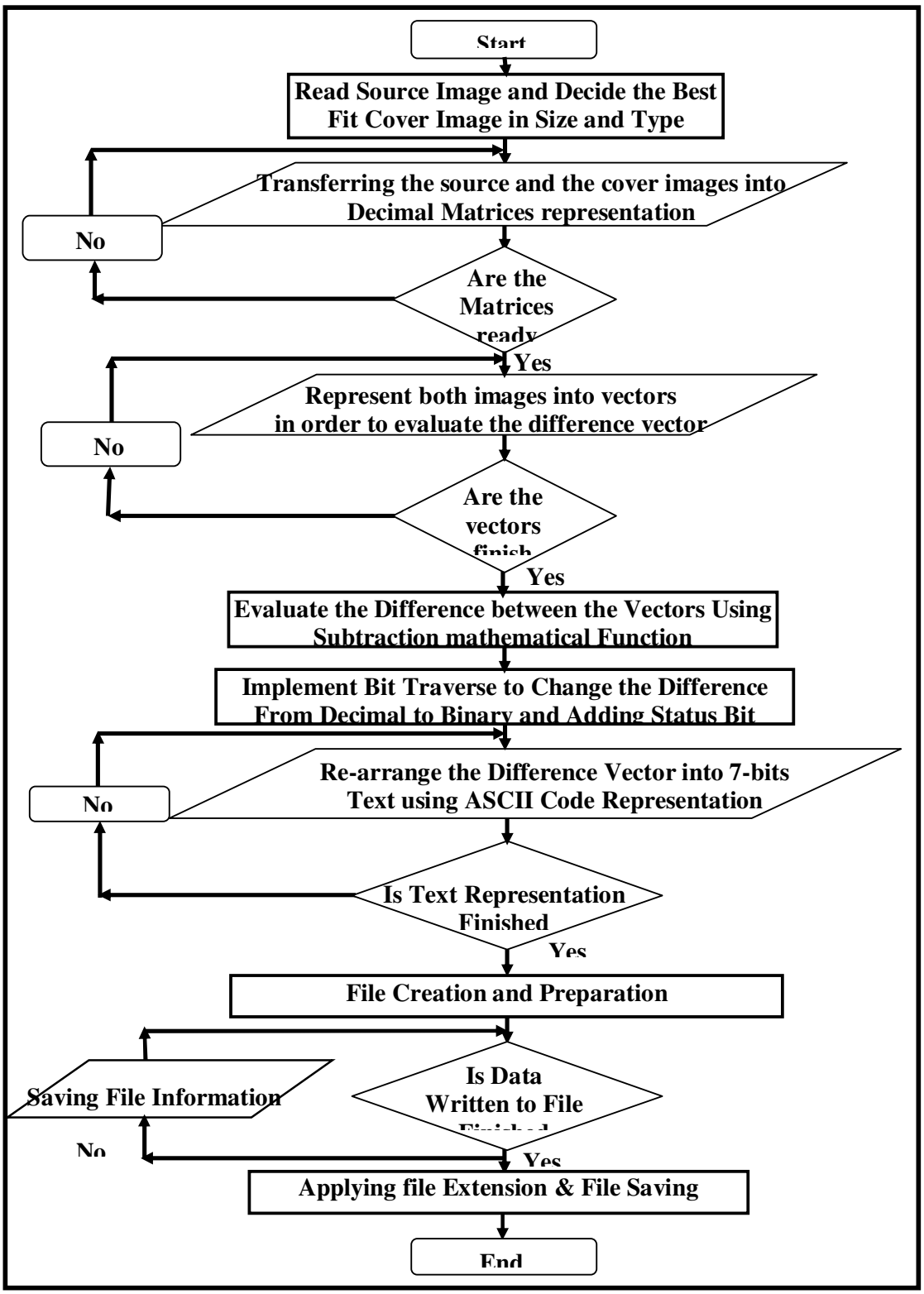

FIGURE 1: the flow chart of the forward phase

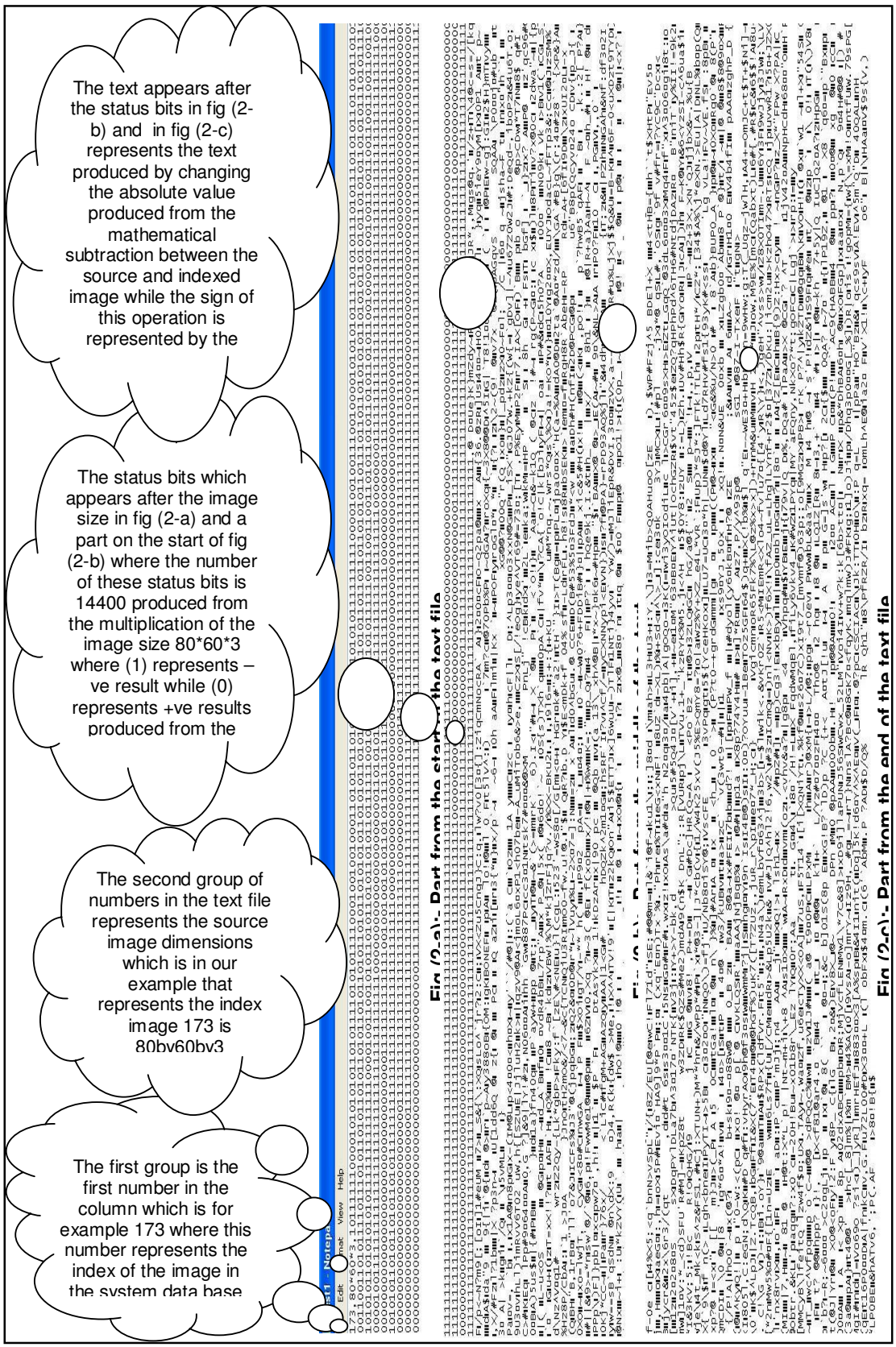

FIGURE 2: Part of the contents of the text file from Different places

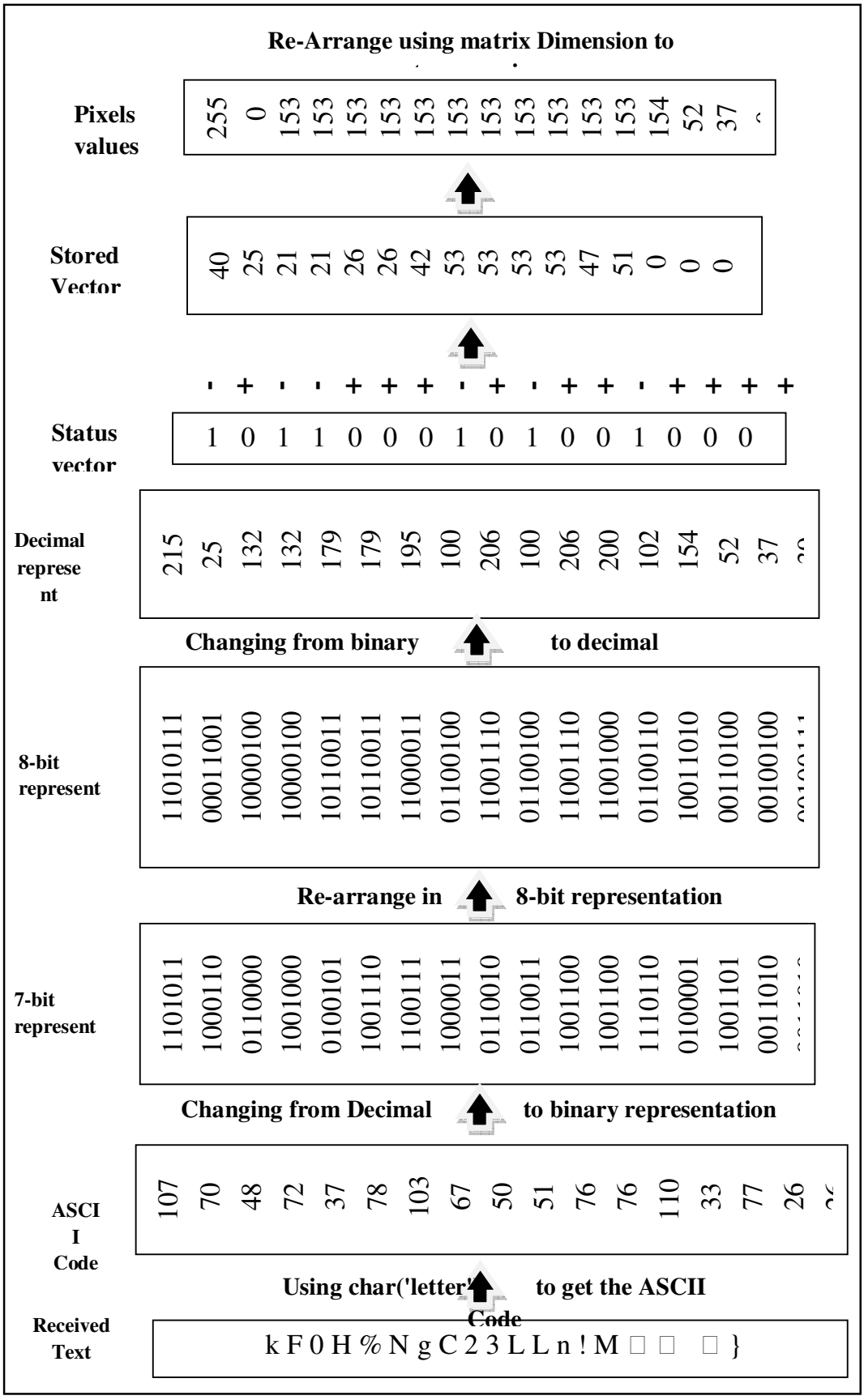

FIGURE 3: A demonstration for the process applied at the receiving end for 20 character

6. REFERENCES

[1] Juneja, M. --- Sandhu, P.S., "Designing of robust image steganography technique based on LSB insertion and encryption" International Conference on Advances in Recent Technologies in Communication and Computing Year: 2009 Pages: 302-305 Provider: IEEE Publisher: IEEE DOI: 10.1109/ARTCom.2009.228.

[2] Raja, K.B. --- Chowdary, C.R. --- Venugopal, K.R. --- Patnaik, L.M. "A secure image steganography using LSB, DCT and compression technique on raw image" Intelligent Sensing and Information Processing, 2005. ICISIP 2005. Third International Conference on ISBN: 0780395883 Year: 2005 Pages: 171-176 Provider: IEEE Publisher: IEEE.

[3] Neeta, D. --- Snehal, K. --- Jacobs, D., "Implementation of LSB steganography and its evaluation for various bits", 1st International Conference on Digital Information Management **ISBN**: 142440682X **Year**: 2007 **Pages**: 173-178 **Provider**: IEEE **Publisher**: IEEE **DOI**: 10.1109/ICDIM.2007.369349.

[4] Hanling Zhang --- Guangzhi Geng --- Caiqiong Xiong ,"Image steganography using pixel – value differencing" , Second International Symposium on Electronic Commerce and Security Year: 2009 Volume: 2 Pages: 109-112 Provider: IEEE Publisher: IEEE DOI: 10.1109/ISECS.2009.139.

[5] Nasser Hamad Faculty of Information Technology, Arab American University, Palestine, *Hiding Text Information in a Digital Image Based on Entropy Function* , The International Arab Journal of Information Technology, Vol. 7, No. 2, April 2010

[6] Muhalim Mohamed, Mohamed Amin, Subariah Ibrahim, Mazleen Salleh, Mohd Rozi Katmin. , "Information Hiding Using Steganography", Department of Computer System & Communication Faculty of Computer Science and Information system University Technology Malaysia, 2003.

[7] Sukhpreet Kaur , Department of Computer Science and Engineering Baba Farid College of Engineering and Technology Bathinda-India & Sumeet Kaur, Department of Computer Engineering Yadavindra College of Engineering Punjabi ,University Guru Kashi Campus Talwandi Sabo, *A Novel Approach for Hiding Text Using Image Steganography* , (IJCSIS) International Journal of Computer Science and Information Security ,Vol. 8, No. 7, Punjab, India October2010.

[8] H.B.Kekre, Archana Athawale, and Pallavi N.Halarnkar , *Increased Capacity of Information Hiding in LSB's Method for Text and Image,* International Journal of Electrical and Electronics Engineering 2:4:2008

[9] K Suresh Babu*, K B Raja*, Kiran Kumar K*, Manjula Devi T H*, Venugopal K R*, L M Patnaik** "Authentication of Secret Information in Image Steganography" , TENCON 2008 - 2008 IEEE Region 10 Conference Year: 2008 Pages: 1-6 Provider: IEEE Publisher: IEEE DOI: 10.1109/TENCON.2008.4766581

[10] *Department of Computer Science and Engineering University Visvesvaraya College of Engineering, Bangalore University, Bangalore 560 001

[11] ** Microprocessor Applications Laboratory, Indian Institute of Science, Bangalore ksureshbabu uvce@rediff.com

[12] K. Pramitha1, Dr. L.Padma Suresh2, K.L.Shunmuganathan3 " Image Steganography Using MOD-4 Embedding Algorithm Based on Image Contrast" TENCON 2008 - 2008 IEEE Region 10 Conference Year: 2008 Pages: 1-6 Provider: IEEE Publisher: IEEE DOI: 10.1109/TENCON.2008.4766581

[13] 1 Department of Electrical and Electronics Engineering, P.G. Scholar, NIU, Kumaracoil. E-mail:pramitha2007@gmail.com

[14] 2 Department of Electrical and Electronics Engineering, HOD, NIU, Kumaracoil. E-mail: suresh_lps@yahoo.co.in

[15] 3 Department of Computer Science Engineering, Professor, RMK Engineering College. E-mail:kls_nathan@yahoo.com

[16] Deshpande Neeta, Kamalapur Snehal Computer Science Dept. K.K.Wagh Institute of Engineering Education & Research, Nashik India , deshpande_neeta@yahoo.com, kamalapur_snehal@yahoo.com Daisy Jacobs School of Information Technology University of Pretoria, Pretoria 002 South Africa daisy.jacobs@up.ac.za, "Implementation of LSB Steganography and Its Evaluation for Various Bits", : 1st International Conference on Digital Information Management ISBN: 142440682X Year: 2007 Pages: 173-178 Provider: IEEE Publisher: IEEE DOI: 10.1109/ICDIM.2007.369349

[17] Hossein Malekmohamadi and Shahrokh Ghaemmaghami Sharif University of Technology, Tehran, Iran h_malekmohamadi@ee.sharif.ir, ghaemmag@sharif.edu "Steganalysis of LSB Based Image Steganography Using Spatial and Frequency Features", IEEE International Conference on Multimedia and Expo ISSN: 19457871 Year: 2009 Pages: 1744-1747 Provider: IEEE Publisher: IEEE DOI: 10.1109/ICME.2009.5202858

[18] Lisa M. Marvel and Charles T. Retter U.S. Army Research Laboratory Aberdeen Proving Ground, MD 21005 marvelQar1. Mil Charles G. Boncelet, Jr University of Delaware Newark, DE 19716 boncelet @ee.udel.edu, "Hiding Information in Images *" Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on ISBN: 08186882 Year: 1998 Volume: 2 Pages: 396-398 vol.2 Provider: IEEE Publisher: IEEE Comput. Soc

[19] 14) Mohammad Shirali-Shahreza, Sajad Shirali- Shahreza. d Shirali-Shahreza"Steganography in TeX Documents", 3rd International Conference on Intelligent System and Knowledge Engineering Year: 2008 Volume: 1 Pages: 1363-1366 Provider: IEEE Publisher: IEEE DOI: 10.1109/ISKE.2008.4731144

[20] 1.Computer Science Department Sharif University of Technology Tehran, IRAN

[21] 2.Computer Engineering Department Sharif University of Technology Tehran, IRAN shirali@cs.sharif.edu, shirali@ce.sharif.edu