Enhancing the Least Significant Bit (LSB) Algorithm for Steganography

O. Osunade
Department of Computer Science
University of Ibadan
Ibadan

I. A. Ganiyu
Department of Computer Science
Oduduwa University
Ipetumodu

ABSTRACT
Various Steganography algorithms have been proposed and implemented for hiding the existence of data in a cover object starting from the algorithms that work in transform domain to the ones that work in spatial domain, such as Least Significant Bit (LSB), which uses the three colours (RED, GREEN and BLUE) present in an image. Three colours are present in the pixel of an image, therefore, this project proposed a new algorithm that chooses only the two colours (GREEN and BLUE) out of the three colours (RED, GREEN and BLUE) that made up of a pixel present in an image to hide data. This proposed algorithm successfully hides the data with the two colours (GREEN and BLUE) present in an image with no significant changes in the resulting colours of the image. The result of this experiment has shown the effectiveness of the proposed algorithm. This experimental result has shown that the algorithm strikes a balance between the security and the quality of the image. It should be noted that this research work only considers image as the cover object, other forms of cover object are not considered here. It should also be noted that the algorithm only hides data from 8 bytes to 1024 bytes using two different images of different size, which shows no effect on the effectiveness of the algorithm.

General Terms
Security, Algorithms

Keywords
Steganography, least significant bit, colour, data, algorithm

1. INTRODUCTION
Due to the continuous changing of global Technology trends, data is continuously moving from one host system to another system on the network or on the internet and thus the security of this data is highly important.

It is generally accepted that the security of the data can be achieved by using encryption and Steganography method. In Cryptography, the encrypted data is transmitted after the data is transformed to another form in order to hide the content of the data from unauthorized users. Steganography on the other hand, deals with hiding the existence of the data in a cover object such as texts, image, audio/video and protocol rather than transforming the data itself thereby making people unaware that communication is taking place.

The application of Steganography will continue to play a vital role in protecting data across several hosts due to its unsuspicious methodology. Various Steganography algorithms have been proposed and implemented but most of the algorithms do not hide the data effectively. Usually a slight distortion in the image used to hide the data gives it away. Therefore, there is a need to get an algorithm that gives only the slightest distortion. This is what led to the newly proposed algorithm.

1.1 An Overview of Ancient Steganography
Steganography can be traced back to ancient times. Early attempts at steganography made use of chemicals and even human bodies to convey information. In practice, modern steganography has gone beyond the use of physical bodies and chemicals but in principle, it is still the same as the ancient steganography. Some of the records are outlined below:

- Herodotus (484 BC – 425BC) is one of the earliest Greek historian. His great Work, The Histories, is the story of the war between the huge Persian Empire and the war between the huge Persian Empire and the much smaller Greek city–states. Herodotus recounts the story of Histaiaeus, who wanted to encourage Aristagoras of Miletus to revolt against the Persian King in other to secure convey his plan, Histaiateus shaved the head of his messenger, wrote the message on his scalp, and then waited for the hair to regrow. The messenger, apparently carrying nothing contentious, could travel freely. Arriving at his destination, he shaved his head and pointed it at the recipient.
- Pliny the Elder (23 AD – 79 AD) explained how the Milk of the thithymallus plant dried to transparency when applied to paper but darkened to brown when subsequently heated, thus recording one of the earliest recipes for invisible ink. The Ancient Chinese wrote notes on small pieces of silk that they then wadded into little balls and coated in wax, to be swallowed by a messenger and retrieved at the messenger's gastrointestinal convenience.

Giovanni Batista Porta (1535 - 1615) described how to conceal a message within a hardboiled egg by writing on the shell with an ounce of alum and a pint of vinegar. The solution penetrates the porous shell, leaving no visible trace, but the message is stained on the surface of the hardened egg albumen, so it can be read when the shell is removed.

2. RELATED WORK
Steganography is an art and science of hiding messages in such a way that no one apart from the intended recipient knows the existence of the message [3]. The term 'hiding' refer to the process of making the information imperceptible or keeping the existence of the information secret.

Steganography is derived from two Greek words 'steganos' which literally means 'covered' and 'graphy' means 'writing' i.e. covered writing. Steganography refers to the science of

'invisible' communication for hiding secret information in various file formats, there exist a large variety of Steganographic techniques. Some are more complex than others but all of them have respective strong and weak points [10]. Different applications have different requirement of the steganography techniques to be used.

Hiding data is the process of embedding information into digital content without causing perceptual degradation. In data hiding three famous techniques can be used. They are watermarking, steganography and cryptography. Steganography is defined as cover writing in Greek. It involves any process that deals with data or information within other. [15].

The main advantage of using Steganography over the remaining famous techniques is due to its simple security mechanism because steganographic message is integrated invisibly and covered inside other harmless sources.

The Steganography can be considered as a branch of Cryptography that tries to hide messages within others, avoiding the perception that there is some kind of message. To apply steganographic techniques, cover files of any kind can be used, although archives of image, sound or video files are the most used today. Similarly, information to hide can be texts, image, video, sound e.tc. There are two trends at the time to implement steganography algorithms: the method that work in the spatial domain (altering the desired characteristics on the file itself) and the methods that work in the transform domain (performing a series of changes to the cover image before hiding information) [9].

Different research carried out has proved the fact that the methods that work in the spatial domain are simpler and faster to implement than the ones that work in the transform domain which is more robust in term of resistance to attacks.

In Spatial Domain, message or data to be transferred is embedded directly into images to be used as cover object whereas, in transform domain as its name implies, images are first transformed before the data or message to be transferred is embedded into it.

Image steganography can be implemented using Transfer domain and Spatial domain which implements any of these three methods:

- Non-Filtering: This method deals with embedding the data into the cover object by starting from the first pixel of the images to be used as cover object.
- Randomized: In this method both the sender and receiver of the image use password denominated stego-key that is employed as the seed for pseudo-random number generator, which then creates sequence that is used as index to have access to the image pixel.
- Filtering: In this method, the algorithm filters the cover image by using a default filter and hides information in the areas that get a better rate [14].

2.1 Steganography Algorithms

Most of the algorithm that works in Spatial Domain use Least Significant Bits Algorithm (LSB) method or any of its derivatives as the algorithm for information hiding i.e., hiding one bit of information in the least significant bit of each colour of a pixel. However, this method cannot stand some types of statistical analysis (such as RS or Sample Pairs). The problem stems from the fact that modifying the three colours of a pixel produces a major distortion in the resulting colour. This distortion is not visible to the human eye, but detectable by statistical analysis [9].

Research carried out has proved the fact that the methods that work in the spatial domain is simpler and faster to implement than the one that work in the transform domain which is more robust in term of resistant to attacks. Therefore, this project focuses on Least Significant Bits Algorithm (LSB) method and its derivative Selected Least Significant Bit Algorithm (SLSB).

2.2 Least Significant Bit Algorithm

In Least Significant Bit Algorithm, both the data and the image to be used as cover object are converted from their pixel format to binary. And the Least Significant Bit of the image is substituted with the bit of the data to be transferred so as to reflect the message that needs to be hidden. The bits of the data replace each of the colours of the Least Significant Bit of the Image [10].

For instance, suppose the data 'AID' with the following property is to be stored in the first 8 pixels of 200 by 400 Pixels with 24 bits in a pixel that made up the image.

Table 1: Showing 3 letters with ASCII values and corresponding BINARY values

LETTER	ASCII VALUES	BINARY VALUES
A	065	01000001
I	105	01100100
D	100	01101001

To hide 'AID' with the Binary Code (01000001 01100100 01101001) using Least Significant Bit Algorithm, each bit with the least significant bit of each colour that made up the Pixel is flipped.

The affected Bits is half of the bits of the images, since there are 256 possible intensities of each primary colour, changing the LSB of a pixel results in small changes in the intensity of the colours. These changes cannot be perceived by the human eye - thus this makes the data to be successfully hidden.

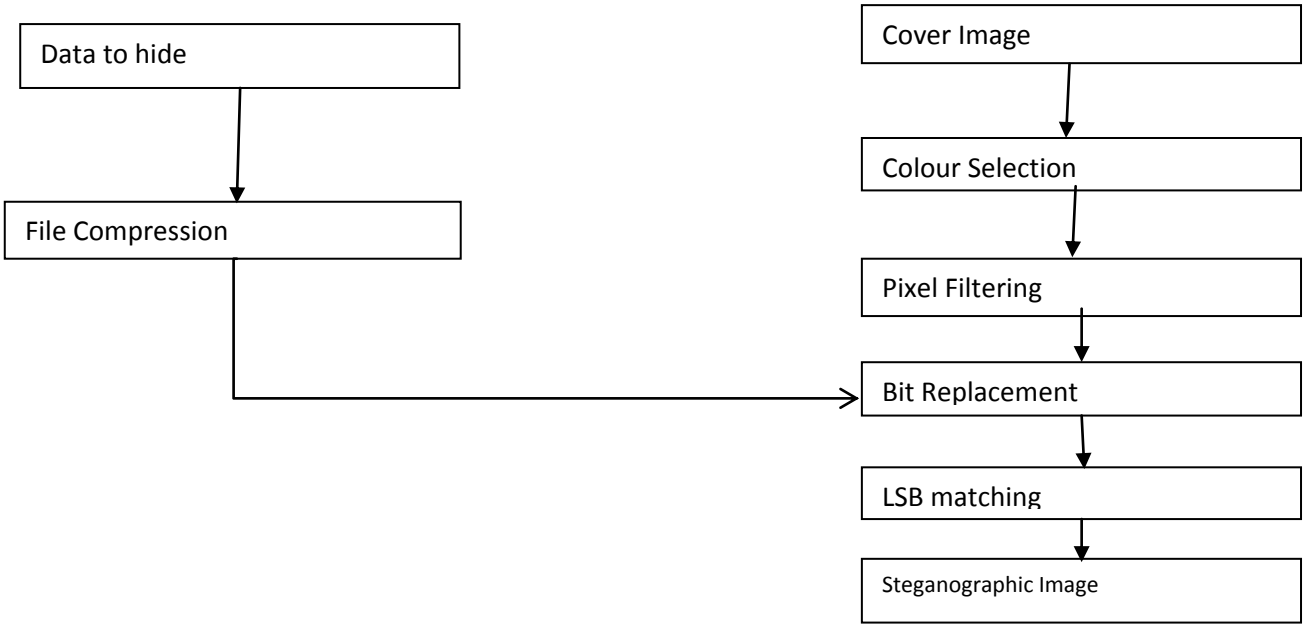


Figure 1: Least Significant Bit method adapted from [14]

2.3 Selected Least Significant Bit Algorithm

In Selected Least Significant Bit Algorithm, both the data and the image used as cover object are converted from pixel format to binary. The Least Significant Bit of one colour (BLUE) that made up a Pixel is substituted with the bit of the data to be transferred. This will reflect the message that needs to be hidden. Only the Least Significant Bit of one colour in a Pixel is flipped by the bits of the data to hide [9].

Only one-third (1/3) of the bits of the image is used. Hiding Data using Selected Least Significant Bit takes more pixels of images compared to the Least Significant Bits method of hidden data, since only the last colour of the Least Significant Bit is going to be replaced. As a result, the human eye cannot perceive the changes - thus this makes the Data to be successfully hidden and inconspicuous to the human eye.

3. PROPOSED ALGORITHM FOR NEW SELECTED LEAST SIGNIFICANT BIT

In this technique, a new steganography algorithm that is based on selecting the Least Significant Bit of the two colours (Green and Blue) in each pixel is proposed, since images in a computer system are represented as arrays of values. These values represent the intensities of the three colours R (Red), G (Green) and B (Blue), where the value for each of the three colours describes a pixel. Each pixel is combination of three components (Red, Green and Blue).

In this scheme, the bits of last two components (Green and Blue) of Pixels of image have been replaced with Data Bits. The blue colour is selected because of a research conducted by Hecht [7], which reveals that the visual perception of intensely BLUE objects is less distinct than the perception of objects of Red and Green. Green is chosen in combination with Blue because it gives more room for the length of the data to be embedded

3.1 Proposed Procedure for Embedding Phase

To embed data into images the following procedure is performed

Step 1: Extract the entire pixel in the image and store it in the array called Pixel-array

Step 2: Extract all the characters in the given text file and store it in the array called Character-array.

Step 3: Extract all the characters from the Stego-key and store it in the array called Key- array.

Step 4: Choose first pixel and pick characters from Key- array and place it in first and second component of pixel. If there are more characters in Key-array, then place rest in the first component of next pixels.

Step 5: Place some terminating symbol to indicate end of the key.

Step 6: Place characters of Character- Array in each first and second components (Blue and Green channel) of next pixels by replacing it.

Step 7: Repeat step 6 till all the characters has been embedded.

Step 8: Again place some terminating symbol to indicate end of data.

Step 9: Obtained image will hide all the characters that input.

3.2 Proposed Procedure for Extraction Phase

To extract data from Stego- image the following procedure should be performed

Step 1: Consider three arrays, Character-Array, Key-array and Pixel- array.

Step 2: Extract all the pixels in the given image and store it in the array called Pixel-array.

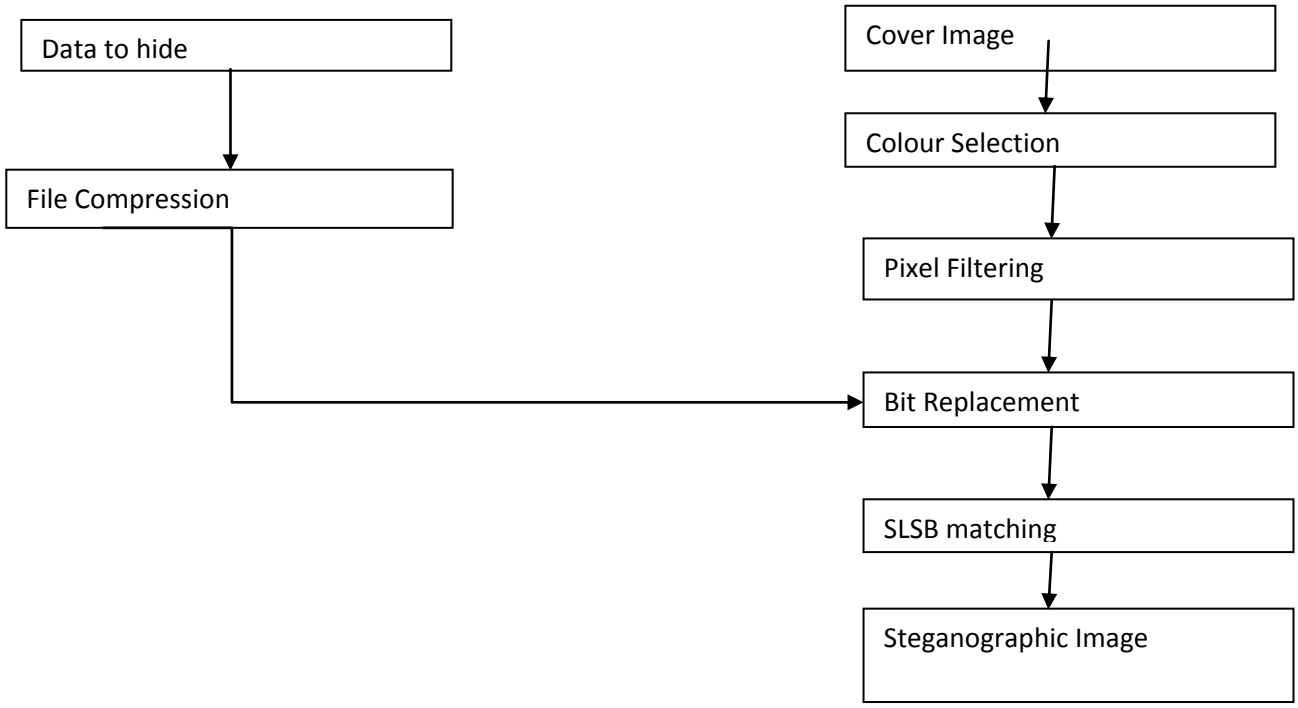


Fig 2: Proposed Selected Least Significant Bit method

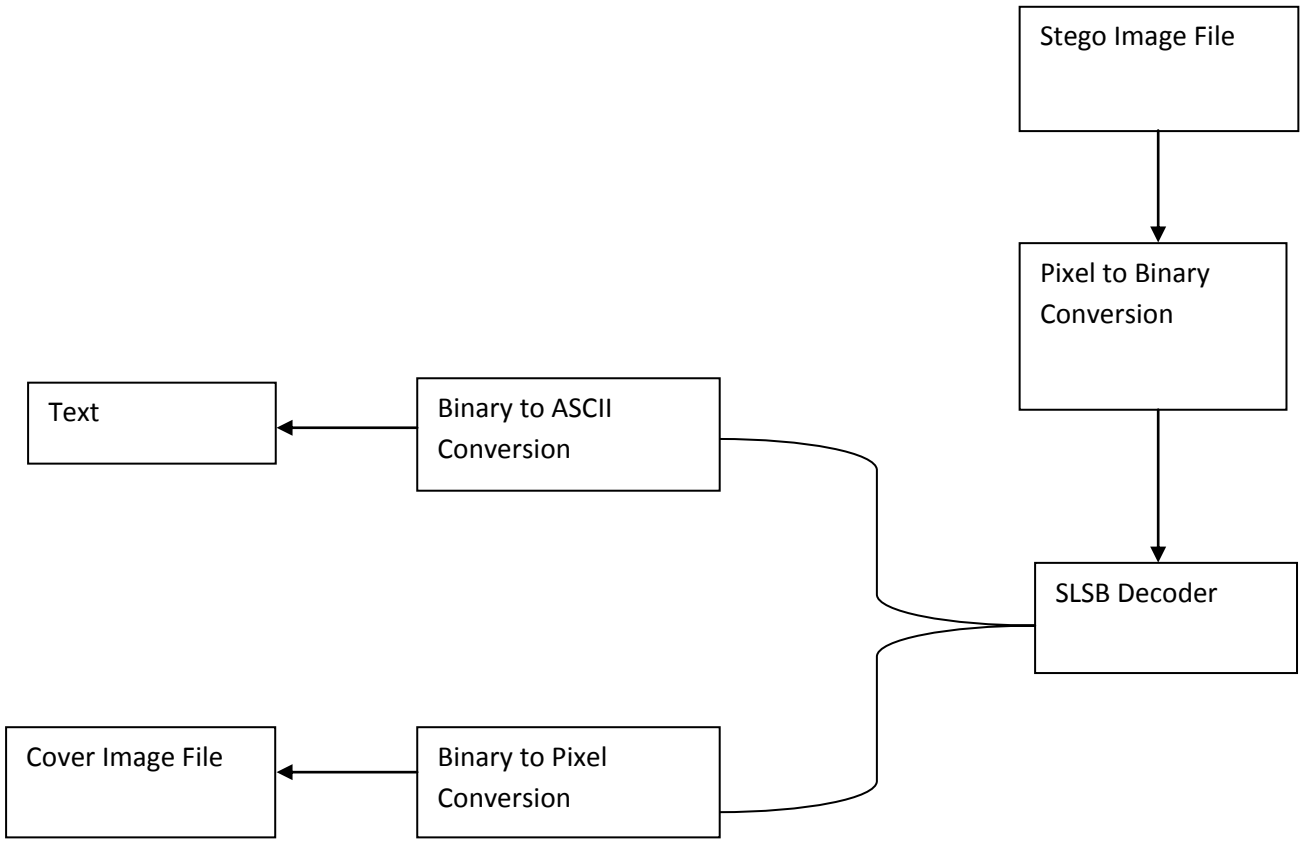


Fig 3: Steganography Mechanism Receiver

Step 3: start scanning pixels from first pixel and extract key characters from first and second (blue and green) components of the pixels and place it in Key-array. Follow

Step 3 up to terminating symbol, otherwise follow step 4.

Step 4: If this extracted key matches with the key entered by the receiver, then follow; otherwise, terminate the program by displaying message "Key is not correct"

Step 5: If the key is valid, then again start scanning next pixels and extract secret Message characters from first (Blue and Green) component of next pixels and place it in Character array. Follow Step 5 till up to terminating symbol, otherwise follow step 6.

Step 6: Extract secret message from Character-array.

3.3 Interface Design

The user interface is generally the means of communication between the user and the system i.e. to enable the user to access the system. It is important that this communication is as meaningful and friendly as possible.

Based on the proposed algorithm, we develop a simple interface using: Java Graphical User Interface i.e., Java Net Beans and Eclipse, since the system is implemented using Java Programming Language. It is a very simple interface to use with the following buttons:

- ENCODE: This Button when click will open a text box where user is asked to input the data to be hidden in the cover object.
- ENCODE NOW: This Button when clicked will open a dialog box for the user to browse for the preferred Cover Object (Image).
- DECODE: This Button when click will open a dialog Box for the user to browse for the Stego Image that has the data embedded in a cover object (Image).
- DECODE NOW: This Button when click will decode the Stego image.
- EXIT Button: This button is use to terminate the application programmed

4. RESULTS

Histograms are a very useful tools used to analyze and compare significant changes in the frequency of appearance of the colors of the cover image with steganographic images so as to be able to get a quick summary of the tonal range present in any given image.

It plots a graph of the tones in the image from black (on the left) to white (on the right). A histogram with lots of dark pixels will be skewed to the left and one with lots of lighter tones will be skewed to the right.

For efficient analysis and comparison, two different images are used and detailed analysis of the four component of any image: Brightness, Red, Green and Blue colors have been carried out.

Figure 4: Original Image

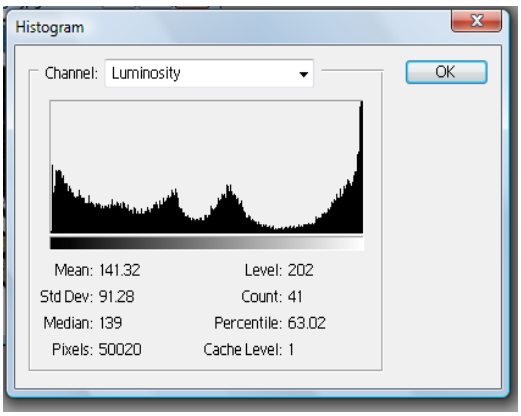


Figure 5: Original Image (Luminosity)

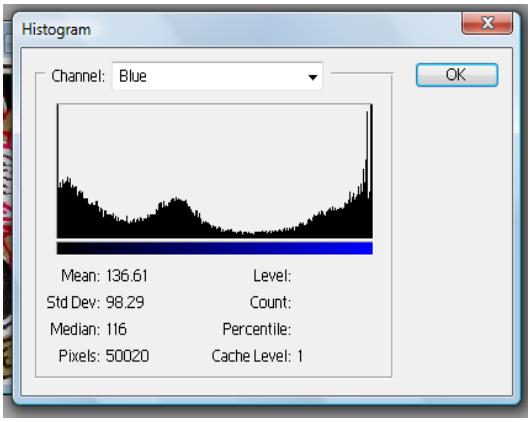


Figure 6: Original Image (Blue)

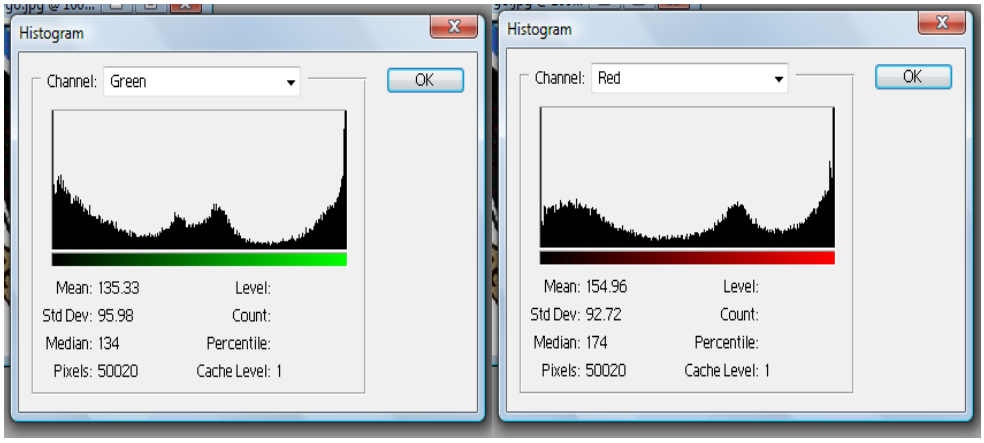
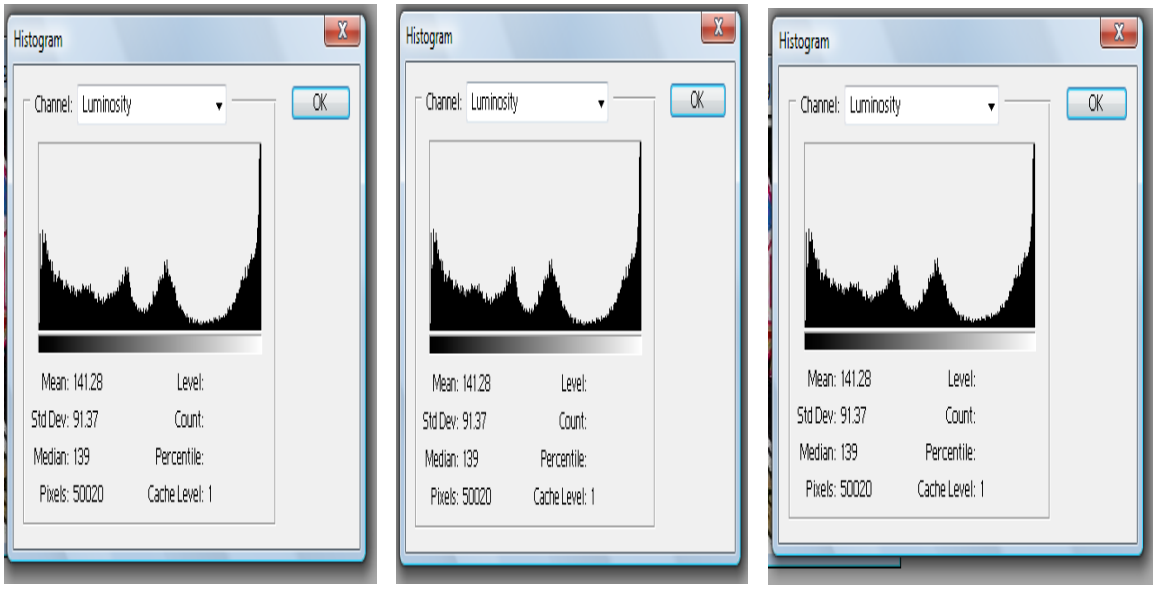


Figure 7: Original Image (Green) **Figure 8: Original Image (Red)**

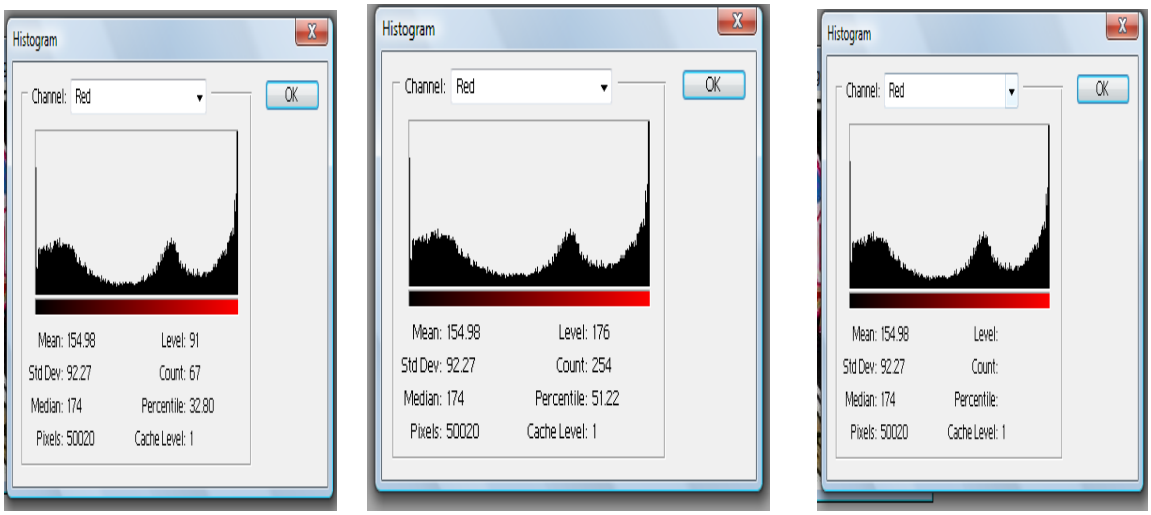
Stego Image using LSB **Stego Image using SLSB** **Stego Image using NEW SLSB**

Figure 9: Stego Images



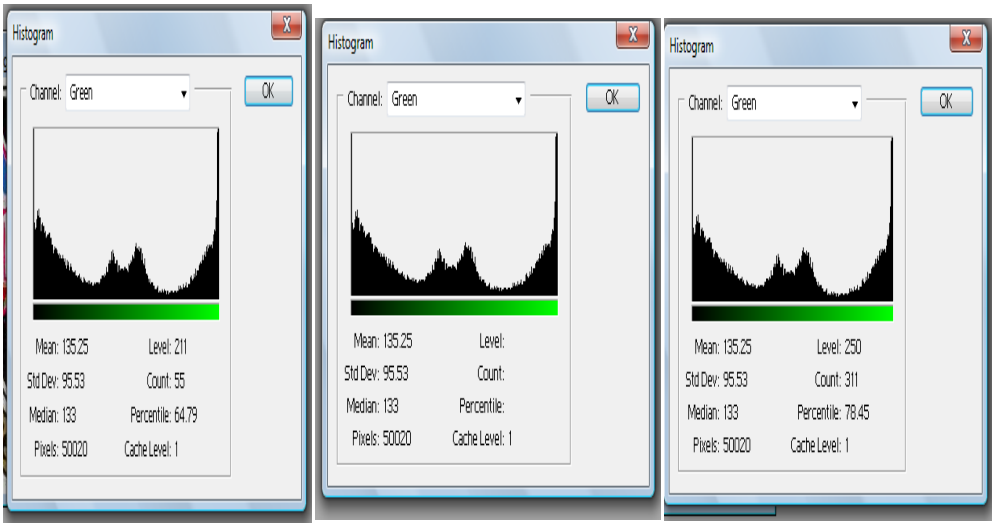
LSB Image SLSB Image NEW SLSB Image

Figure 10: Luminosity channel of Stego Images



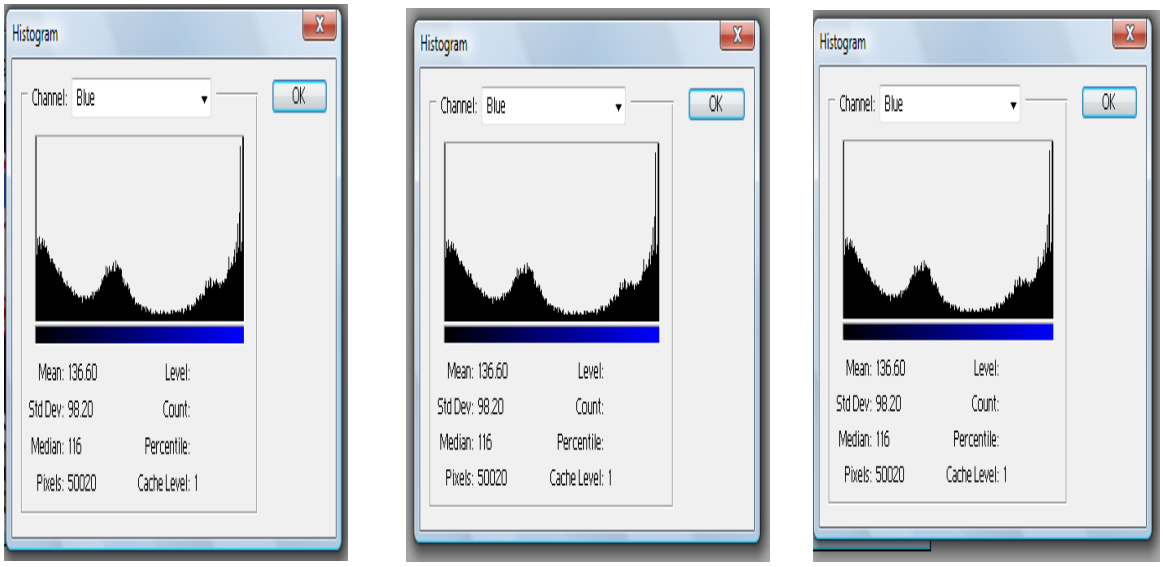
LSB Image SLSB Image NEW SLSB image

Figure 11: Red channel of Stego Images



LSB Image SLSB Image NEW SLSB image

Figure 12: Green channel of Stego Images



LSB Image SLSB Image NEW SLSB Image

As it can be seen from the Experimental result (Histogram analysis) above, the algorithms are tested using an image. The result has shown that all the algorithms successfully hides the image with no difference in the resulting frequency in the color of the images and sizes

Table 2: showing the image size before and after encoding using different algorithms

No	Original Image (Size)	Hidden Data File (Size)	LSB Algorithm (Size)	SLS Algorithm (Size)	NEW SLSB Algorithm (Size)
1	**11.97kb**	**32 bytes**	**111kb**	**111kb**	**111kb**
2	**5.97kb**	**32 bytes**	**77.4kb**	**77.4kb**	**77.4kb**

5. CONCLUSION

The result of the experiment perform has shown the effectiveness of the proposed algorithm. The experimental result has shown that the algorithm strikes a balance between Least Significant Bits Algorithm (LSB) and Selected Least Significant Bit (SLB) algorithm in such a way that achieved balance between the security and the quality of the image. There is no loss of the data hidden whatsoever and this new method retains the quality of the image.

This research work, only consider images as the cover object. Other forms of cover object are not considered here. The algorithm only hides data between 8 bytes and 1024 bytes. Future work will be how to use the algorithm with other forms of cover object i.e., Text, Video and also to hide data of bigger size.

6. ACKNOWLEDGMENTS

Thanks to the Omolola Olamide for his contributions towards development of the template.

7. REFERENCES

[1] Arvind K. and Kim P. (2010). "Steganography- A Data Hiding Technique" International Journal of Computer Applications ISSN 0975 – 8887, Volume 9– No.7, November 2010.

[2] Chen P. and Wu W. (2009). A modified side match scheme for image steganography, International Journal of Applied Science & Engineering 7 (2009) 53-60.

[3] Divya S.S and Ram M. (2012). Hiding text in audio using multiple lsb steganography and provide security using cryptography. International journal of scientific & technology research volume 1, issue 6, July 2012.

[4] El-Emam N. (2007) Hiding a large amount of data with high security using steganography algorithm, Journal of Computer Science 3 (2007) 223-232.

[5] Fridrich J, Du R and Meng L. (2000) "Steganalysis of LSB Encoding in Color Images," Proc. IEEE Int'l Conf. Multimedia and Expo, CD-ROM, IEEE Press, Piscataway, N.J., 2000.

[6] Gandharba S. and Saroj K.L. (2012). A Technique for Secret Communication Using a New Block Cipherwith Dynamic Steganograph. International Journal of Security and Its Applications Vol. 6, No. 2, April, 2012

[7] Hecht, E. 2006. Optics. Delhi, India: Pearson Education.

[8] John M. and Manimurugan S. (2012).A Survey on Various Encryption Techniques. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, 2(1), March 2012.

[9] Juan J. and Jesus M. (2009). SLSB: Improving the Steganographic Algorithm LSB. Universidad Nacional de Educación a Distancia (Spain).

[10] Lokeswara V., Subramanyam A. and Chenna P. (2011). Implementation of LSB Steganography and its Evaluation for Various File Formats. Int. Journal Advanced Networking and Application, 2(5), Pages: 868-872

[11] Lou D., Liu J. and Tso H. (2008)Evolution of information – hiding technology, in H. Nemati (Ed.), Premier Reference Source–Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, pp. 438-450.

[12] Mauro B., Franco B., Vito .C and Alessandro P. (1999). A DCT-domain system for robust image watermarking. Dipartimento di Ingegneria Elettronica, Universita di Firenze,via di S. Marta, 3, 50139 Firenze, Italy

[13] Morkel T., Eloff J. and Olivier M. (2005). An overview of image steganography.Information and Computer Security Architecture (ICSA) Research Group.Department of Computer Science University of Pretoria, 0002, Pretoria, South Africa.

[14] Roque, Juan J., and Jesús M. M., (2009) "SLSB: Improving the Steganographic Algorithm LSB." WOSIS.

[15] Roziati I. and Teoh (2011). Steganography Algorithm to Hide Secret Message inside an Image. Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia (UTHM), Batu Pahat 86400, Johor, Malaysia

[16] Stefan K. and Fabien A. (2000) "Information HidingTechniques for Steganography and Digital Watermarking". Boston,Artech House, pp. 43 – 82. 2000.

[17] Thomas A. (2005). Implementing Steganographic Algorithms: An Analysis and Comparison of Data Saturation

[18] Vijay k. and vishal S. (2005). A steganography algorithm for hiding image in image by improved lsb substitution by minimize detection .Journal of Theoretical and Applied Information Technology

[19] Wu P and Tsai W. (2003). A steganographic method for images by pixel-value differencing, Pattern Recognition Letters 24 (2003) 1613-1626.3.