Implementation of Video Steganography Using Hash Function in LSB Technique

S. Chitra, M.Tech Student, Department of Electronics and Communication Engg, Kuppam Engineering College, Kuppam, Chittor district, Andra Pradesh - 517 425,

Narasimhalu Thoti,Assistant professor, Department of Electronics and Communication Engg, Kuppam Engineering College, Kuppam, Chittor district, Andra Pradesh-517 425

Abstract

Data hiding is the process of embedding information in a data source without changing its perceptual quality. There are various techniques used for data hiding namely cryptography, watermarking and steganography. Cryptography deals with the study of encoding techniques for secret data transmission through communication channels under condition that the third party should not be able to read and interpret the data. Watermarking is a technique where a recognizable image or the pattern in paper that appears as various shades of lightness or darkness when viewed by reflected light. Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message. This project is based on video Steganography implemented by using hash LSB method. Video steganography deals with hiding secret data or information within a video. A hash function is used to select the position of insertion in LSB bits. The proposed technique is evaluated in terms of PSNR and MSE which is found to be better than LSB method.

1. Introduction

Steganography is the science that deals with hiding of secret data in some carrier media which may be image, audio, formatted text or video. The main idea behind this is to conceal the very existence of data. It derives from the Greek word steganos, meaning covered or secret, and graphy (writing or drawing).The medium where the secret data is hidden is called as cover medium, this can be image, video or an audio file. The information to be hidden in the cover data is known as the "embedded" data. The "stego" object or data is the data containing both the cover signal and the "embedded" information. Logically, the process of putting the hidden or embedded data, into the cover data, is sometimes known as embedding. In decoding process, Stego object is decoded with the help of the public or private key. Depending upon the encoding technique, sometimes the original cover image is also needed in decoding process. After successful decoding, the embedding secret information can be extracted and viewed.

1.1 Block diagram of steganographic operation.

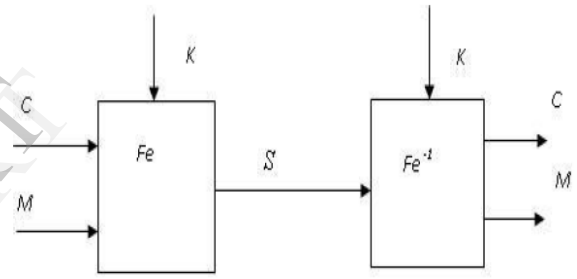


Figure 1.1: Block diagram of steganographic operation.

Application of Steganography varies from military, industrial applications to copyright and Intellectual Property Rights (IPR). By using lossless steganography techniques messages can be sent and received securely. Traditionally, steganography was based on hiding secret information in image files. But modern work suggests that there has been growing interest among research fraternity in applying steganographic techniques to video files as well. The advantage of using video files in hiding information is the added security against the attack of hacker due to the relative complexity of the structure of video compared to image files.

2. Literature survey

2.1 The past of steganography

Steganography has been used in various forms for 2500 years. It has found use in military, diplomatic, personal and intellectual property applications. Briefly stated, steganography is the term applied to any number of processes that will hide a message within an object, where the hidden message will

not be apparent to an observer. Several stenographic methods were used during World War II .Microdots developed by the Nazis are essentially microfilm chips created at high magnification. The microfilm chips are the size of periods on a standard typewriter .These dots could contain pages of information, drawings, etc. More recent uses of stenographic techniques involved a photograph of the captured crew of the U.S.S. Pueblo where the crewmembers spelled the word "snowjob" using various hand positions.

2.2 Present

Currently, the emphasis has been on various forms of digital steganography. Commonly there are a number of digital technologies that the community is concerned with, namely text files, still images, movie images and audio. The majority of other organizations using steganographic techniques involve individuals or corporations interested in protecting intellectual property.

2.3 Future of Steganography

Steganographic techniques have obvious uses, some legitimate and some less so. The business case for protection of property, real and intellectual is strong. Individuals or organizations may decide to place personal, private or sensitive information in steganographic carriers. With advances in steganography, it is possible that this medium could serve as a relatively secure storage and transmission method. Steganographic techniques can also be used in the application of digital watermarks. Using a variety of techniques, images, music, movies can be imprinted with digital watermarks.

2.4 Steganalysis

The process of detecting steganography is called as steganalysis. The discovery of digital files that may contain secret messages may be broadly divided into two types, images and textual. In the matter of images, one method is creating a statistical profile of compressed data files that make up natural or undisturbed images, and then checking a given image against the profile. Another method is to use a tool such as "Stegdetect" which scans JPEG images to examine the presence of secret message. Detection of real time steganography (e.g. ISDN or TCP/IP steganographic techniques) will require either real time steganalysis, or capture of packets for later analysis.

2.5 Steganographic Techniques

The steganographic systems are classified based on the stego-key used in the embedding process in to three types as follows.

2.5.1 Pure Steganography. Pure Steganography is defined as a steganographic system that does not require the exchange of a cipher such as a stego-key. This method of Steganography is the least secure means by which to communicate secretly because the sender and receiver can rely only upon the presumption that no other parties are aware of this secret message.

2.5.2 Secret Key Steganography. Secret Key Steganography is defined as a Steganographic system that requires the exchange of a secret key (stego-key) prior to communication. Secret Key Steganography takes a cover message and embeds the secret message inside it using a secret key (stego-key). Only the parties who know the secret key can reverse the process and read the secret message. Unlike Pure Steganography where a perceived invisible communication channel is present, Secret Key Steganography exchanges a stego-key, which makes it more secure.

2.5.3 Public Key Steganography. Public Key Steganography is defined as a steganographic system that uses a public key and a private key to secure the communication between the parties wanting to communicate secretly. Public key is used during the encoding process and private is used during the decoding process. The sender will use the public key during the encoding process and only the private key, which has a direst mathematical relationship with the public key, can decipher the secret message. Public Key Steganography provides a more robust way of implementing a steganographic system because it can utilize a much more robust and researched technology.

2.6 Classification of steganography based on cover media.

Based on the cover medium used, steganography can be classified into the following types

2.6.1 Plaintext steganography. In this technique the message is hidden within a plain text file using different schemes like use of selected characters, extra white spaces of the cover text etc.

2.6.2 IP datagram steganography. This is another approach of steganography, which employs hiding data in the network datagram level in a TCP/IP

based network like Internet. Network covert channel is the synonym of network steganography. Overall goal of this approach to make the stego datagram is undetectable by Network watchers like sniffer, Intrusion Detection System (IDS) etc.

2.6.3 Image steganography. The most widely used technique today is hiding of secret messages into a digital image. This steganography technique exploits the weakness of the human visual system (HVS). HVS cannot detect the variation in luminance of color vectors at higher frequency side of the visual spectrum. A picture can be represented by a collection of color pixels.

2.6.4 Audio Steganography. In audio steganography, secret message is embedded into digitized audio signal which result slight altering of binary sequence of the corresponding audio file. There are several methods are available for audio steganography like LSB Coding, Phase Coding: Human Auditory System (HAS) etc

2.6.5 Video Steganography. Video Steganography is a technique where the secret message is hidden within a video file. The original video file is known as cover file and the video obtained after embedding the secret data is known as stego – video. This project proposes a steganographic model which utilizes cover video files to conceal the presence of other sensitive data regardless of its format. The model presented is based on pixel-wise manipulation of colored raw video files to embed the secret data.

2.7 Techniques for implementing steganography

There are mainly two approaches for implementing steganography.

2.7.1 Spatial Domain or substitution Techniques. Spatial domain techniques either operate on pixel wise or block wise bases. It Substitutes redundant parts of a cover with a secret message. (E.g. LSB method, matrix embedding etc)

2.7.2 Transform or frequency Domain Techniques. In this technique, images are first transformed to frequency domain by using FFT, DCT or DWT and then the messages are embedded in some or all of the transformed coefficients.

2.8 Least Significant Method

Sometimes abbreviated as LSB, the least significant bit is the lowest bit in a series of numbers in binary; the LSB is located at the far right of a string. For example, in the binary number: 10111001, the least significant bit is the far right 1. The lsb is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position. It is common to assign each bit a position number, ranging from zero to N-1, where N is the number of bits in the binary representation used. Normally, this is simply the exponent for the corresponding bit weight in base-2 (such as in $2^{31}...2^{0}$). Although a few CPU manufacturers assign bit numbers the opposite way (which is not the same as different endianness), the term lsb (of course) remains unambiguous as an alias for the unit bit. By extension, the least significant bits (plural) are the bits of the number closest to, and including, the lsb. LSB insertion usually has a 50% chance to change a LSB every 8 bits, thus adding very little noise to the original frame. For 24-bit images the modification can be extended sometimes to the second or even the third LSBs without being visible. 8-bit images instead have a much more limited space where to choose colors, so it's usually possible to change only the LSBs without the modification being detectable. The cover medium first of all must seem casual, so it must be chosen between a set of subjects that can have a reason to be exchanged between the source and the receiver. The LSB technique can be used for implementing different types of steganography like image steganography, audio steganography, video steganography etc.

3. Video Steganography

In video steganography, the video file is converted into frames. About 1 to 50 frames can be generated per second. A digital frame consists of a matrix of colour and intensity values. The images or the frames in a video are divided into three types: binary (Black- White), Gray scale and Red-Green-Blue (RGB) images. The binary image has one bit value per pixel represent by 0 for black and 1 for white pixels. While the gray scale image has 8 bits value per pixel represent from 00000000 for black and 11111111 for white pixels. The RGB image has 24 bits values per pixel represent by (00000000, 00000000 and 00000000) for black and (11111111, 1111111 and 11111111) for white pixels. The RGB image is the most suitable because it contains a lot of information that help in hiding the secret information with a bit change in the image resolution which does not affect the image quality and make the message more secure. Modulating the least significant bit does not result in a human-perceptible difference because the amplitude of the change is small other techniques

"process" the message with a pseudo-random noise sequence before or during insertion into the cover image. The advantage of LSB embedding is its simplicity. In a 24 bit image, one can store 3 bits in each pixel by changing a bit of each of the red, green and blue colour components, since they are each represented by a byte. A 600x500 pixel image, can thus store a total amount of 9, 00,000 bits or 112,500 bits of embedded data. LSB embedding also allows high perceptual transparency. We opt for LSB techniques to hide the secret message as the design is simple and payload capacity is better. But the main disadvantage is that attacker can easily remove the message by removing (zeroing) the entire LSB plane with very little change in the perceptual quality of modified stego file. Therefore to improve the security level of hidden information, we go for Hash LSB technique which is an enhanced version of LSB technique. In Hash based LSB method, the secret information is hidden in the LSB of RGB pixel value of the carrier frames. The embedding position is decided by the hash function.

3.1 Hash Function

A typical hash function is a well-defined procedure or mathematical function that uses a single integer that may serve as an index to an array and returns a number of values called hash codes or values. A hash function is also defined as a process that takes an input of arbitrary size and returns a fixed size output, which is called hash value or message digest. A hash function should be referentially transparent or stable, i.e., if called twice on input that is "equal" (for example, strings that consist of the same sequence of characters), it should give the same result. Hash functions provide a means for integrity checking which can mostly detect and correct errors introduced by the network.

3.2 Implementation of video steganography by using hash function in LSB technique

In this project, a hash based LSB technique is proposed in spatial domain. There are several steganographic methods and most of which are performed in pixel domain. The existing methods are mainly based on LSB where LSBs of the cover file are directly changed with message bits. But in the existing LSB method, only one bit can be hidden in a 8 bit pixel value which limits the capacity of the frame to hide the secret message and it is also prone to attacks where the LSBs of the pixels are manipulated to obtain the secret message. A video stream (AVI) consists of collection of frames and the secret data is embedded in these frames as payload. The information of the cover video (AVI) such as number of frames (n), frame speed (fp/sec), frame height (H) and width (W) are extracted from the header. The cover video is then broken down into frames. Now the proposed LSB based technique has been applied to conceal the data in the carrier frames. The size of the message does not matter in video steganography as the message can be embedded in multiple frames. The proposed technique takes eight bits of secret data at a time and conceal them in LSB of RGB(Red, Green and Blue) pixel value of the carrier frames in 3, 3, 2 order respectively. Such that out of eight bits of message six bits are inserted in R and G pixel and remaining two bits are inserted in B pixel. The embedding positions of the eight bits out of the four available bits of LSB is obtained using a hash function of the form,

$$k = p\%n$$

Where, k is LSB bit position within the pixel, p represents the position of each hidden image pixel and n is number of bits of LSB. The bits are distributed randomly during fabrication which increases the robustness of the technique compared to other LSB based techniques. After concealing data in multiple frames of the carrier video, frames are then grouped together to form a stego video, which is now an embedded video to be, used as normal sequence of streaming. The intended user follows the reverse steps to decode the secret data. During decoding the stego video is again broken into frames after reading the header information. Using the same hash function which is known to the intended user, the data of the secret message is regenerated. The extracted stream of the secret information is used to authenticate the video.

3.3 Illustration of HASH LSB process

Consider a RGB pixel value of the cover frame as below

R: 10110111
G: 10010100
B: 11001001

And a byte of message to be inserted in LSB as:
10001001
LSB is lowest bit in a series of binary numbers, so in this case for R it will be 1, 0 for G and 1 for B. The proposed technique is applied in four lowest LSBs in each pixel value. So the LSBs for the above RGB values are:

R: 0111
G: 0100
B: 1001

The message is embedded in groups of 3, 3 and 2 in the respective RGB LSBs positions. The positions are obtained from the hash function given by the equation k=p%n. The value of n number of bits of LSB is 4. Using the hash function let the position of insertion k returned for a particular iteration are,

k = 1, 2, 3 for R.
k = 4, 1, 2 for G
k = 3, 4 for B

Considering the above positions of insertion, the bits from the message are inserted in four LSB positions and resulting RGB pixel value are as given below:

R: 1011**100**1
G: 1001**100****0**
B: 11001010**1**

Thus all the eight bits of the message are embedded in three bytes and number of bits actually changed is five out of twenty four bits. Further these five bits are randomly distributed which increases the robustness of the scheme. On decoding the message, the valid user follows the reverse step. As the hash function is known to the intended the user, it calculates the k values to get the position of insertion. Taking the same embedded RGB value as above,

R: 10111001
G: 10011000
B: 11001001

The hash function will return the following k values for this particular iteration.

k = 1,2,3 for R.
k = 4,1,2 for G
k = 3,4 for B

Using these k values which represent the four LSB positions, the data of the secret message is found as below,10001001,Which is same as the data of secret message.

4. ALGORITHM OF HLSB TECHNIQUE

4.1 Algorithm of Encoding

Step 1: Input cover video file or stream.
Step 2: Read required information of the cover video.
Step 3: Break the video into frames.
Step 4: Find 4 LSB bits of each RGB pixels of the cover frame.
Step 5: Obtain the position for embedding the secret data using hash function given by the equation k = p%n.
Step 6: The secret message is converted to binary format and is segmented into blocks of eight bits each, that is one block has eight bits of the secret message .
Step 6: Embed the eight bits of the secret image into 4 bits of LSB of RGB pixels of the cover frame in the order of 3, 3, 2 respectively using the position obtained from step 5.
Step 7: Regenerate video frames.
Step 8: Obtain stego video file or stream.

4.2 BLOCK DIAGRAM OF ENCODING

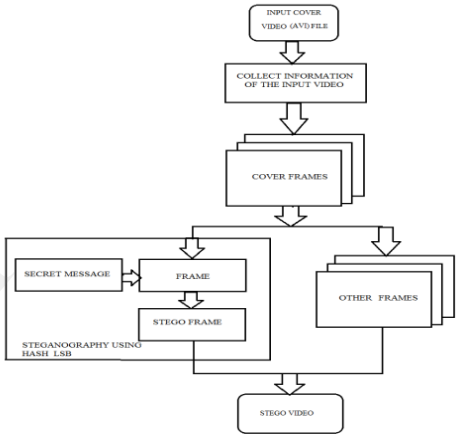


Figure 4.1Block diagram of encoding process.

During the encoding process, the input cover video file of .avi format is first processed to obtain all information about the video file regarding the length of the input video and the capacity of the input video. And then sampling is done which is followed by quantization to covert the analog signal into binary sequence. And then the input cover video is divided into frames, about 25 to 50 frames can be generated per second from the input video file. Each frame is taken individually to hide the secret message. In this project, only RGB pixels of the images are considered. The RGB image has 24 bits values per pixel represent by (00000000, 00000000 and 00000000) for black and (11111111, 1111111 and 11111111) for white pixels. We consider the last four LSBs in the RGB pixel values of the cover video where the secret message is embedded. The position of embedding the secret message in the last four LSBs of cover file is decided by the hash function. The secret message is converted to binary format and is segmented into blocks prior to being embedded in the cover video. These blocks are then embedded in the cover video. Depending on the capacity of the secret message

the number of frames required is decided. Thus the resulting video consists of the secret message embedded within and it is called as stego video. This stego video is same as the input cover video. By using hash function the security of the secret message is improved.

4.3 Algorithm of Decoding

Step 1: Input stego video file or stream.
Step 2: Read required information from the stego video.
Step 3: Break the video into frames.
Step 4: Find 4 LSB bits of each RGB pixels of the stego frame.
Step 5: Obtain the position of embedded of the secret data using hash function ,k=p%n.
Step 6: Retrieve the bits using these positions in the order of 3, 3, 2 respectively.
Step 7: Reconstruct the secret information.
Step 8: Regenerate video frames.

4.4 Block Diagram of Decoding Process:

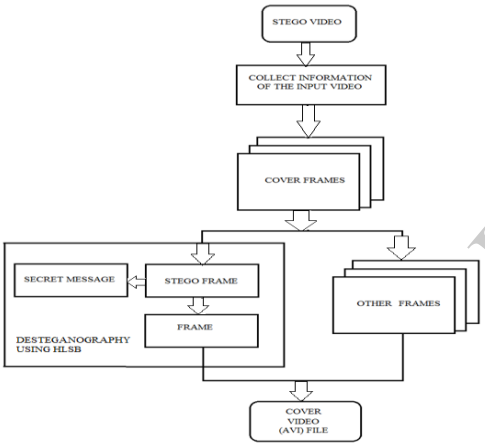


Figure 4.2: Block diagram of decoding process

During decoding the reverse process of encoding is performed. Information is collected from stego video and it is divided into frames. And then stego frame is identified and using the same hash function used in encoding process the secret message is retrieved back.

4.5 Specific Requirement

4.5.1.Hardware Requirement specification. Intel Pentium III Processor, 2 GB Hardisk, 20 GB HDD,CD-ROM.

4.5.2. Software Requirement Specification. Operating System: Windows XP/Vista-Professional Version, Programming Tool: MAT Lab 7.10.

5. Results and Performance Evaluation

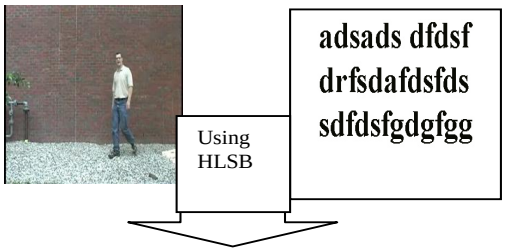
The proposed method is implemented on MATLAB. Any Steganography technique is characterized mainly by two attributes, imperceptibility and capacity. Imperceptibility means the embedded data must be imperceptible to the observer (perceptual invisibility) and computer analysis (statistical invisibility). The performance of the proposed technique is evaluated using two different video streams (man.avi, car.avi and video.avi) and one secret data. The perceptual imperceptibility of the embedded data is indicated by comparing the original image or video to its stego counterpart so that their visual differences, if any, can be determined. Additionally, as an objective measure, the Mean squared Error (MSE), Peak Signal to Noise Ratio (PSNR) between the stego frame and its corresponding cover frame are studied.

$$MSE = \frac{1}{H*W} \sum_{i=1}^{H} (P(i, j) - S(i, j))^2$$

where, MSE is Mean Square error, H and W are height width and P(i,j) represents original frame and S(i,j) represents corresponding stego frame.

$$PSNR = 10 \log_{10} \frac{L^2}{MSE}$$

where, PSNR is peak signal to noise ratio, L is peak signal level for a grey scale image it is taken as 255.Maximum payload (bits per byte/bpb) for the technique has also been obtained i.e. maximum amount of data that can be embedded into the cover image without losing the fidelity of the original image. In the proposed scheme eight bits of data are embedded in 3 pixels of the cover frame. Figure 5.1 shows the input video file[Man.avi] and the secret message [1.jpg]and after embedding the secret message into the input it results in the stego-video[Man1.avi]. Figure 5.2 shows the input video file [car.avi] and the secret message [1.jpg]and after embedding the secret message into the input it results in the stego-video[car1.avi].

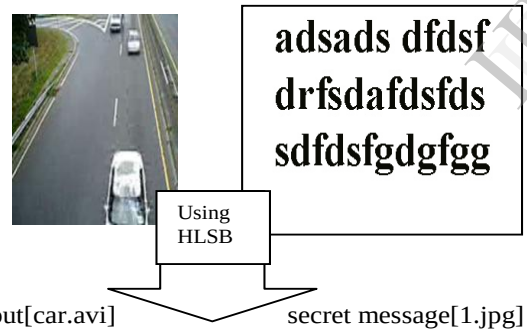


[Man.avi] Secret message[1.jpg]

Stego-video [Man1.avi]

(For the above video PSNR=47.6909, MSE=1.1066)

Figure 5.1 Example of HLSB



Input[car.avi] secret message[1.jpg]

Stego-video [car1.avi]

(For the above video PSNR=47.6909, MSE=1.1066)

Figure 5.2 Example of HLSB.

Table 5.1 gives the details of resolution and the no of frames generated. Table 5.2 gives the comparison between Hash LSB and LSB in terms of PSNR, MSE and payload. The results show that in Hash LSB method, the PSNR, MSE and payload is better than LSB method

Table 5.1: Cover video file details

Name of video	Resolution (W*H)	Frame/ sec	No. of frames	Secret message resolution
Man.avi	256*256	3	25	256*256
Car.avi	256*256	5	25	256*256

Table 5.2: Results obtained from Hash LSB and LSB method

Name of video	PSNR	MSE	Payload	PSNR	MSE	Payload
Man. avi	47.69	1.106	2.66	51.23	1.32	1
Car. avi	53.04	0.322	2.66	58.56	0.42	1

5.1 Steganalysis of video

Several image Steganalysis technique exist in literature. In recent times researchers have developed some video steganalysis techniques. In a technique for video steganalysis by using the redundant information present in the temporal domain as a deterrent against secret messages embedded by spread spectrum steganography has been proposed. A video steganalysis method using neural networks and support vector machines to detect hidden information by exploring the spatial and temporal redundancies. Literature survey suggests that when temporal redundancies are used as video steganalysis then performance is more satisfactory than in spatial domain. Where as in steganalysis algorithm has been proposed that uses the correlation between adjacent frames to detect a special distribution mode across the frames. This is considered to work well with AVI file formats. However every carrier media is supposed to have its own special characteristics and thus it behaves differently when a message is embedded in it. The existing video steganalysis technique may not work very well to detect the presence of secret message in HLSB technique.

6. Conclusion

A secured hash based LSB technique for video steganography has been proposed in this paper. This technique utilizes cover video files in spatial domain to conceal the presence of sensitive data regardless of its format. Performance analysis of the proposed technique after comparison with LSB technique is quite encouraging. The proposed technique is applied to AVI files, however it can work with any other formats with minor procedural modification. For compressed video files like MPEG the video needs to first decompress then the technique can be applied to the uncompressed video. Whereas for Flash Video FLV files the technique can be applied with little modification. Software based steganographic Engine for video steganography is the future scope of the technique

7. References

[1] Kousik Dasgupta, J.K. Mandal and Paramartha Dutta "Hash based least significant bit technique for video steganography(HLSB)", International Journal of Security, Privacy and Trust Management (IJSPTM), Vol. 1, No 2, April 2012.

[2] Mritha Ramalingam, Stego Machine Video Steganography using Modified LSB Algorithm, in World Academy of Science, Engineering and Technology 74, pp. 502-505, 2011.

[3] Atallah M. Al-Shatnawi, 'A N,ew Method in Image Steganography with Improved Image Quality", Applied Mathematical Sciences, Vol. 6, no. 79, 3907 – 3915, 2012.

[4] Manish Mahajan, Dr. Navdeep Kaur "Adaptive Steganography: A survey of Recent Statistical Aware Steganography Techniques", I. J. Computer Network and Information Security, pp. 76-92, September 2012.

[5] Amit Singh, Susheel Jain, Anurag Jain,"Digital watermarking method using replacement of second LSB with inverse of LSB", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 2, February 2013.

[6] Mrs. Kavitha, Kavita Kadam, Ashwini Koshti, Priya Dunghav, " Steganography Using Least Signicant Bit Algorithm", International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 3, pp. 338-341, May-Jun 2012.

[7] Dr. Ekta Walia a, Payal Jainb, Navdeep," An Analysis of LSB & DCT based Steganography", Global Journal of Computer Science and Technology, Vol. 10 Issue 1 (Ver 1.0), April 2010.

[8] Pritish Bhautmage, Prof. Amutha Jeyakumar, Ashish Dahatonde, " Advanced Video Steganography Algorithm', International Journal of Engineering Research and Applications (IJERA), Vol. 3, Issue 1, , pp.1641-1644, January -February 2013.

[9] H. B. Kekre, Dhirendra Mishra, Rhea khanna, Sakshi Khanna & Aadil Hussaini, " Comparison between the basic LSB Replacement Technique and Increased Capacity of Information Hiding in LSB's Method for Images", International Journal of Computer Applications (0975 – 8887), Volume 45– No.1, May 2012.

[10] Zenon Hrytskiv, Sviatoslav Voloshynovskiy and Yuriy Rytsar, "Cryptography and steganography of video information in modern communications" Facta university,series: electronics and energetics vol. 11, no.1, 1998.

[11] Gabriel Macharia Kamau1* Stephen Kimani2 Waweru Mwangi2," An enhanced Least Significant Bit Steganographic Method for Information Hiding", Journal of Information Engineering and Applications ,ISSN 2224-5782 (print) ISSN 2225-0506 ,Vol 2, No.9, 2012.

[12] Namita Tiwari, Dr.Madhu Shandilya," Evaluation of Various LSB based Methods of Image Steganography on GIF File Format", International Journal of Computer Applications (0975 – 8887) Volume 6– No.2, September 2010.

[13] B. Sharmila, R. Shanthakumari, "Efficient adaptive steganography for color images based on LSBMR algorithm", ICTACT journal on image and video processing, volume 02, issue 03, February 2012.

[14] S.Shanmuga Priya ,K.Mahesh, Dr.K.Kuppusamy," Efficient Steganography Method to Implement Selected Lease Significant Bits in Spatial Domain (SLSB – SD)", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, pp.2632-2637, May-Jun 2012.

[15] Soumyendu Das, Subhendu Das, Bijoy Bandyopadhyay, Institute of Radio physics & Electronics,University of Calcutta, Kolkata, India," Steganography and Steganalysis: Different Approaches", 2004.

[16] James C. Judge, "Steganography: Past, Present, Future", SANS Institute InfoSec Reading Room, SANS Institute 2001.