Enhancement in PSNR using Inverted LSB Mechanism

Stuti Patel
M.Tech (Final Year Student)
Kanpur Institute of Technology, Kanpur208001
Uttar Pradesh, India

***Abstract*— The word Steganography is derived from the word steganography of Greek language which means secured writing. It basically deals with the science of secret communication. The objective of steganography is to conceal the presence of the message from unapproved party. The present time secure image steganography exhibits a goal of exchanging the embedded information to the destination without being recognized by the attacker. Inverted Least Significant Bits (LSB) steganography is explained in this work. In this approach, the fundamental idea is of replacing the LSB of the cover image with the Bits of the messages to be hidden without destroying the property of the cover image fundamentally. This is one of the most challenging techniques because it is quite difficult to make a distinction between the stego-object and the cover-object if some LSB bits of the cover object are replaced. Generally, representation of 8 bits is used, subsequently 8 positions are accessible. Additionally, we have demonstrated that it is not important to put secret information at LSB only, the other bit positions can likewise be used to embed secret image. At some point it is conceivable to better PSNR at other bit positions.**

***Index Terms*—LSB, INVERTED LSB, PSNR**

1. INTRODUCTION

Steganography is the act of concealing sensitive or private information within something that seems, by all accounts, to be nothing out of the usual. Commonly, Steganography is frequently mistaken for cryptology on the grounds that both are utilized to secure critical information. The Steganography is different from cryptology as it includes concealing information so it gives the idea that no information is hidden at all. In the event that a man or persons sees the object that the information is concealed within he or she will have no clue of any hidden information, hence the person won't endeavor to decode the information.

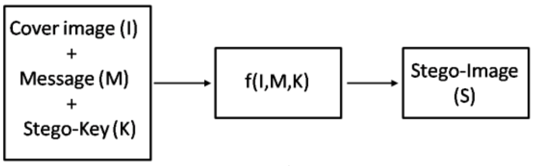


Fig. 1 Fundamental of Steganography

The basically working of Steganography is to exploit human perception, senses of humans are not prepared to search for documents that have information hidden inside them, even then there are programs accessible that can do what is known as Steganalysis (Detecting using Steganography.) The most widely recognized utilization of Steganography is to conceal a record inside another file. At the point a file or information is hidden inside a bearer file, the data is typically encrypted with a secret code or password.

2. PRESENT STATE OF ART

The multimedia objects such as audio, video, image etc are used as cover media by the greater part of today's steganography systems in light of the fact that people regularly transmit digital pictures over email along with the other Internet communication. Present day steganography utilizes the chance of concealing information into digital multimedia documents furthermore at the network packet level [1-4].

Concealing information into a medium needs the elements given below:
1. The cover medium generally an image (C) that will hold the secret message.
2. The secret message (M) may be simple text, image file, video file or any type of data.
3. The stegonographic techniques
4. A stego-key (K) which can be further used to hide and unhide the message.

In present day methodology, on the basis of the cover medium, steganography can be classified into five types:

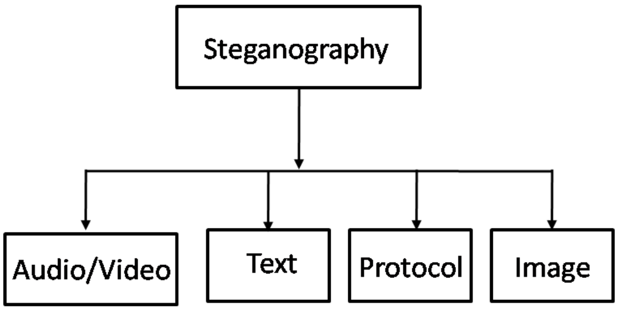


Fig. 2 Types of Steganography

Image steganography: For steganography, images are used as the prevalent cover medium. A message is implanted in a digital image utilizing an embedding algorithm, by making use of the secret key. The resultant stego-image is send to the receiver. While on the other hand, it's processing is done by the extraction algorithm using the same key. Unauthenticated persons can just notice the transmission of an image during the transmission of stego-image but cannot see the presence of the concealed message.

Image Definition

An image could be defined as a picture that has been copied or created and saved in electronic form. It can be described in terms of vector or raster graphics. An image that is stored in raster form is sometimes called a bitmap. An image map is a document that contains the information that associates different locations on a predefined image with hypertext links. An image could be better understood as a collection of numbers that comprise distinctive light intensities in the image‘s different areas. This numeric representation forms a grid and the particular points are detailed as pixels (picture element). For each pixel, Grayscale images uses 8 bits and have the capacity to illustrate 256 different colours or shades of grey. As far as Digital colour images are concerned, they are normally stored in 24-bit files and utilize the RGB colour model, also called true colour. Each variation of colour for the pixels of a 24-bit image is derived from green, red and blue as these three are basic colours and each of this primary colour is represented by 8 bits. Therefore, in one given pixel, there can be 256 distinct amounts of red, green and blue.

An image is an array of numbers for a computer that represent intensities of light at many different points (pixels). The image’s raster data is made up by these pixels. A typical size of an image is 640 x480 pixels and 256 colors (or 8 bits per pixel). An image of such kind could comprise about 300 kilobits of data. Digital images are generally stored in either 8-bit or 24-bit files. A 24-bit image gives the most space for concealing information; on the other hand, it can be quite large (with the exception of JPEG images). These three primary colors: red, green, and blue are responsible for all color variations for the pixels. 1 byte represents each primary color; 24-bit images use 3 bytes per pixel to represent a color value. These 3 bytes can be characterized as hexadecimal, decimal, and binary values. In numerous Web pages, a six-digit hexadecimal number represents the background color—actually three pairs representing red, blue and green. A white background represents the value FFFFFF: in RGB format, 100 percent red means (FF), 100 percent green as (FF), and 100 percent blue as (FF). Its decimal value is 255, 255, 255, and its binary value is 11111111, 11111111, 11111111, which are the three bytes making up white. This definition of a white background is analogous to the color definition of a single pixel in an image. Pixel representation contributes to file size. Let us take an example, assume that we have a 24-bit image 1,024 pixels wide by 768 pixels high—a typical resolution for high resolution graphics. An image of such kind has more than two million pixels, each having such a definition, which would deliver a file that exceeds 2 Mbytes. Since such 24-bit images are still relatively not common on the Internet, their size would draw in consideration during transmission. File compression would hence be advantageous, if not necessary, to transmit such a file.

Transform Domain Technique

The Frequency domain the message is embedded into transformed coefficients of image giving more information hiding capacity and more robustness against attacks. Transform domain embedding can be named as a domain of embedding techniques for which various algorithms have been recommended. In the present time a large portion of strong steganographic systems work inside the transform domain. The techniques of transform domain have a merit in comparison to the LSB techniques as the information in these techniques are concealed in the image areas that are less exposed to cropping, compression, and image processing. Some techniques of the transform domain do not appear to be dependent on the format of the image. These techniques may outrun lossless and format conversions that are lossy.

Transform domain techniques are of different types as given below:

1. Discrete Fourier transformation technique (DFT).
2. Discrete cosine transformation technique (DCT).
3. Discrete Wavelet transformation technique (DWT).

Discrete Cosine Transform

After color coordinate conversion the following step is of dividing the three image color components into numerous 8×8 block. The Forward DCT and the Inverse DCT could be mathematically defined as under:

Forward DCT

$$F(u,v)=\frac{2}{N}C(u)C(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right]$$

$$\text{for } u=0,\ldots,N-1 \text{ and } v=0,\ldots,N-1$$

$$\text{where } N=8 \text{ and } C(k)=\begin{cases}1/\sqrt{2} & \text{for } k=0\\ 1 & \text{otherwise}\end{cases} \qquad (1)$$

Inverse DCT

$$f(x,y)=\frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right]$$

$$\text{for } x=0,\ldots,N-1 \text{ and } y=0,\ldots,N-1 \text{ where } N=8 \qquad (2)$$

In the selected 8×8 block, the $f(x,y)$ is the value of each pixel and the $F(u,v)$ is the DCT coefficient after transformation. The transformation of the 8×8 block is also a 8×8 block composed of $F(u,v)$.

Least-Significant Bit (LSB) Technique

The bit of least significance (i.e. the 8th bit) of some or all of the bytes within an image is transformed into a bit of the secret message. 24 bit images and the 8 bit images are mainly of two types of Digital images [5]. We can install three bits of information in each pixel in 24 bit images, one in each LSB position of the three eight bit values. The LSB doesn’t change the appearance of the image with the increase or decrease of the value; much so the output stego image seems almost similar as the cover image. On the other side, one bit of information can be hidden in 8 bit images[8-9].

The inverse process is applied to remove the concealed image from the stego-image. In the case the LSB of the cover image pixel value $C(i, j)$ is equivalent to the message bit ‘*m*’ of secret massage to be embedded, $C(i, j)$ does not change; if not, set the LSB of $C(i, j)$ to m. The message embedding procedure is given below:

$S(i,j) = C(i, j) - 1$, if LSB $(C(i, j)) = 1$ and $m = 0$
$S(i,j) = C(i, j)$, if LSB$(C(i, j)) = m$
$S(i,j) = C(i, j) + 1$, if LSB $(C(i, j)) = 0$ and $m = 1$
Where LSB $(C(i, j))$ stands for the LSB of cover image $C(i, j)$ and m is the next message bit to be embedded. $S(i,j)$ is the stego image.

As already discussed, each pixel comprises three bytes consisting of either a 1 or a 0. Suppose one can hide a message in three pixels of an image (24-bit colors). Suppose the original 3 pixels are:
(11101010 11101000 11001011)
(01100110 11001010 11101000)
(11001001 00100101 11101001)
A steganographic program could hide the letter "*J*" which has a position 74 into ASCII character set and have a binary representation "01001010", by altering the channel bits of pixels.
(11101010 1110100**1** 1100101**0**)
(01100110 1100101**1** 11101000)
(11001001 0010010**0** 11101001)

We need to change only four bits in this case for the successfully insertion of the character. It is quite hard for a human eye to recognize the resulting changes that are made to the least significant bits as they are too small, hence the message is hidden in an effective manner [6]. The simplicity of LSB embedding is its advantage and many techniques use these methods.
Additionally, it also permits high perceptual transparency.

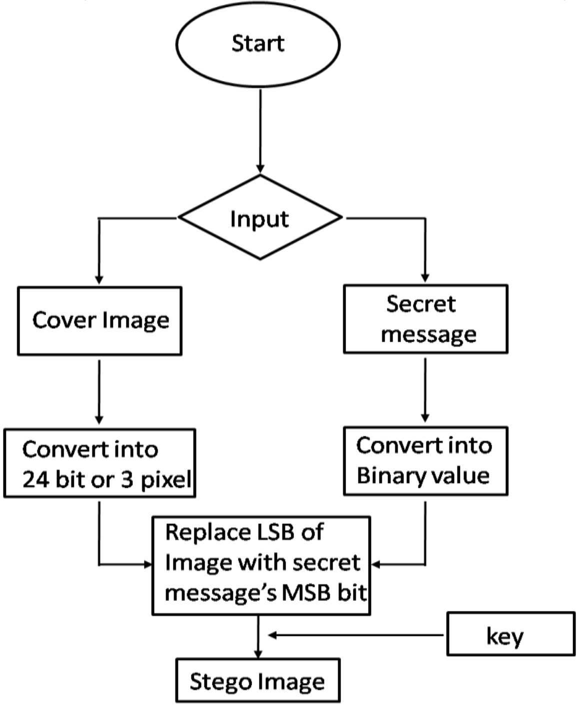

Start
Input
Cover Image
Secret message
Convert into 24 bit or 3 pixel
Convert into Binary value
Replace LSB of Image with secret message's MSB bit
key
Stego Image

Fig. 3 Flow chart description of LSB Steganography

The flow diagram of LSB steganography is shown in Figure 3 [7-10].
Above, we have discussed about the basic LSB scheme with its merits and demerits above. Therefore, it is feasible for invaders to attack images of such kind and extract the secret image or message. Generally, cover image size is much bigger than the secret image, hence it can be imbedded any place in the cover image consequently secret key is required to distinguish the start and end position of the message.
It is demonstrated in this work that it is not important to implant the secret message at LSB only. It could be done at any position staring form LSB to MSB. Moreover, it is also illustrated that sometimes proposed method additionally enhance PSNR. As bit position is not known it upgrade the security.

3. PROPOSED METHODOLOGY

Simple LSB
As considered above, the cover image
(11101010 11101000 11001011)
(01100110 11001010 11101000)
(11001001 00100101 11101001)
The letter "*J*" can be hiding (binary representation "01001010") by altering the channel bits of pixels.
(11101010 1110100**1** 1100101**0**)
(01100110 1100101**1** 11101000)
(11001001 0010010**0** 11101001)
As 4 bits are changed out of 72 bits, therefore stego image will appear as cover image.
Inverted LSB
In this case LSB of the pattern obtained above will be inverted.
(1110101**1** 11101000 11001011)
(0110011**1** 11001010 1110100**1**)
(1100100**0** 00100101 1110100**0**)
In this case 5 bits are altered, therefore MSE will be more.
In the third possible arrangement bits will be inverted only of the portion where secret image is inserted. Thus we have
(1110101**1** 11101000 11001011)
(0110011**1** 11001010 1110100**1**)
(1100100**0** 00100101 11101001)
In this case 4 bits are altered, so MSE will be same as simple LSB.
Cover Image Selection
In Inverted LSB steganography, selection of cover image is also very important, like if cover image would be
(11101010 11101001 11001010)
(01100110 11001011 11101000)
(11001001 00100100 11101001)
Then no bit would have changes and stego image would be same cover image, but this is not straight forward to search for such cover image.
The other possibility can be that the secret image binary representation could be search at any bit position like:
Let cover image is
(111**0**1010 1111**1**000 110**0**1011)
(011**0**0110 110**1**1010 111**0**1000)
(110**1**1001 001**0**0101 11101001)

Then at the fifth bit form LSB sequence is available. Again there will be no distortion and cover image would be exactly same as secret image.
Similarly the other positions can also be searched for inverted LSB, the only point is that MSE should be small either simple LSB or inverted LSB can be used.

The overall decimal representation of binary image would be like:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	0	1	1	0

This is equivalent to 166 in decimal representation, now if LSB is changed to 1, and then the new value would be 167, which is same as 166 in terms of intensity of pixel.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	0	1	1	1

Now let fifth bit form LSB is changed to 1, then the new value would be 182, which is much deviated from 166.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	1	0	1	1	0

Therefore, 4 LSB change is equivalent of 2 Changes bit next to LSB, 1 Change from 2 bit next to LSB.
Hence, in nutshell it can be concluded that bit other than LSB can only be selected if the total error created due to the bit change in other position should be smaller than total error due to LSB change. However, bit by bit scanning in place of LSB can produce better results in some cases.

Worst Case PSNR

In *k* bit representation the maximum error would be $2^k - 1$.

$$PSNR(dB) = 10\log\frac{(255)^2}{(2^k-1)^2} \qquad (3)$$

k	***PSNR*(dB)**
1	48.1308
2	38.5884
3	31.2288
4	24.6090
5	18.3036

It is clear from the table that as *k* increases beyond 3,

4. PERFORMANCE ANALYSIS

As a performance measure for image distortion due to hiding of message, the well-known peak-signal-to noise ratio (PSNR), which is categorized under difference distortion metrics, can be applied to stego images. It is defined as:

$$PSNR(dB) = 10\log\frac{(C_{\max})^2}{MSE} \qquad (4)$$

MSE = mean square error;
This is given as:

$$MSE = \frac{(S-C)^2}{MN} \qquad (5)$$

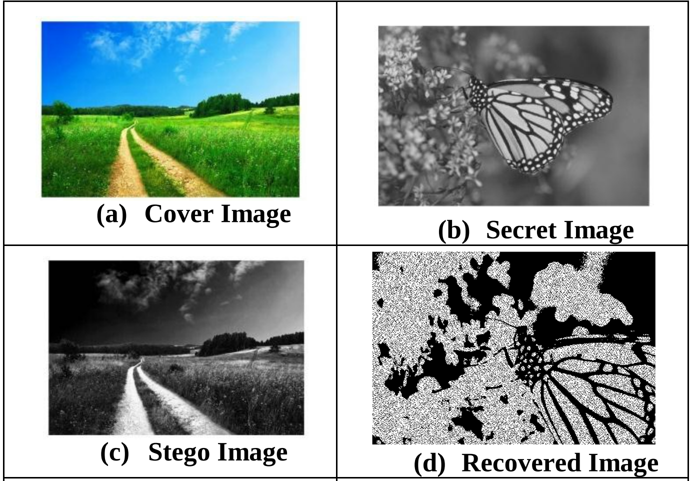
With $C_{\max} = 255$:
Where M and N are the dimensions of the image,
S is the resultant stego-image, and C is the cover image.

The PSNR for the retrieved image is defined as

$$PSNR(dB) = 10\log\frac{(C_{\max})^2}{MSE} \qquad (6)$$

$MSE = \frac{R^2}{MN}$, where *R* is retrieved image.

Experiment 1

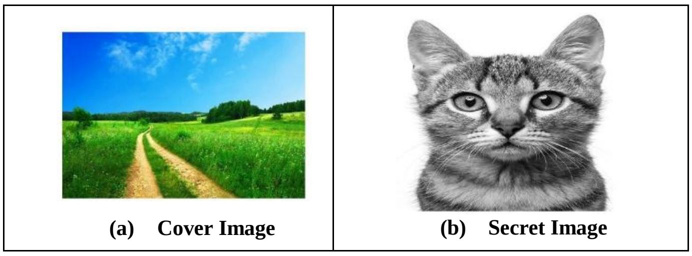

(a) Cover Image
(b) Secret Image
(c) Stego Image
(d) Recovered Image

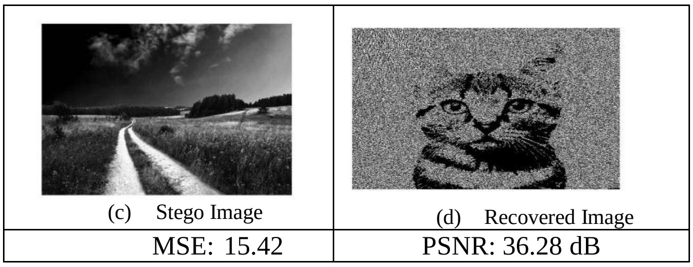
MSE: 72.57	PSNR: 29.55 dB

Figure 4 Cover image Scenery and Secret image as Monarch using LSB method

In figure 4, simple LSB based steganography technique is discussed. Here the cover image is scenery (a) and secret image is monarch (b). The stego image is shown in figure (c) which is very much similar to the cover image and in figure (d) recovered secret image is shown. In the simple LSB scheme MSE is found to be 72.57 and PSNR value as 29.55dB.

Experiment 2


(a) Cover Image
(b) Secret Image


(c) Stego Image

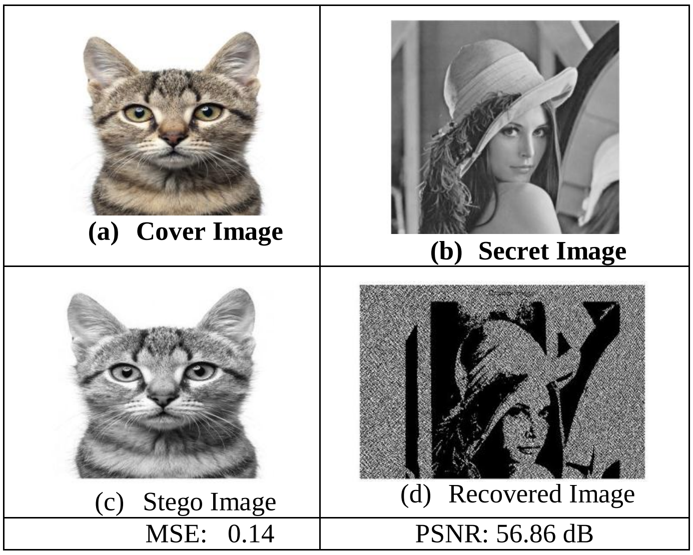
(d) Recovered Image

MSE: 15.42	PSNR: 36.28 dB

Figure 5 Cover image Scenery and Secret image as Cat using LSB method

In figure 5, again simple LSB based steganography technique is discussed. Here the cover image is scenery (a) and secret image is cat (b). The stego image is shown in figure (c) which is very much similar to the cover image and in figure (d) recovered secret image is shown. In the simple LSB scheme MSE is found to be 15.42 and PSNR value as 36.28dB

Experiment 3


(a) Cover Image

(b) Secret Image

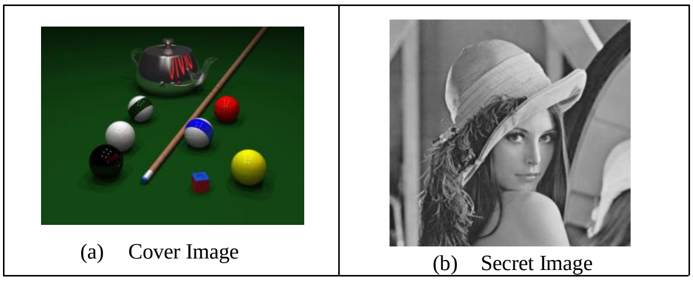
(c) Stego Image

(d) Recovered Image

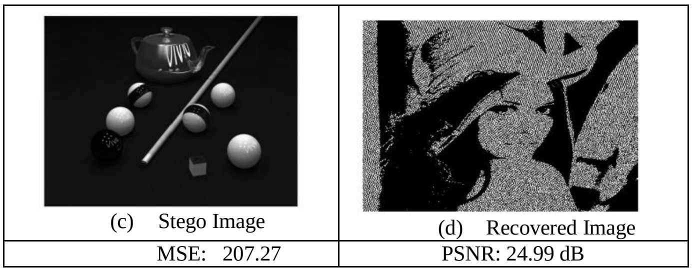
MSE: 0.14	PSNR: 56.86 dB

Figure 6 Cover image Cat and Secret image as Lena using LSB method

In figure 6, again simple LSB based steganography technique is discussed. Here the cover image is cat (a) and secret image is Lena (b). The stego image is shown in figure (c) which is very much similar to the cover image and in figure (d) recovered secret image is shown. In the simple LSB scheme MSE is found to be 0.14 and PSNR value as 56.86dB.


(a) Cover Image

(b) Secret Image

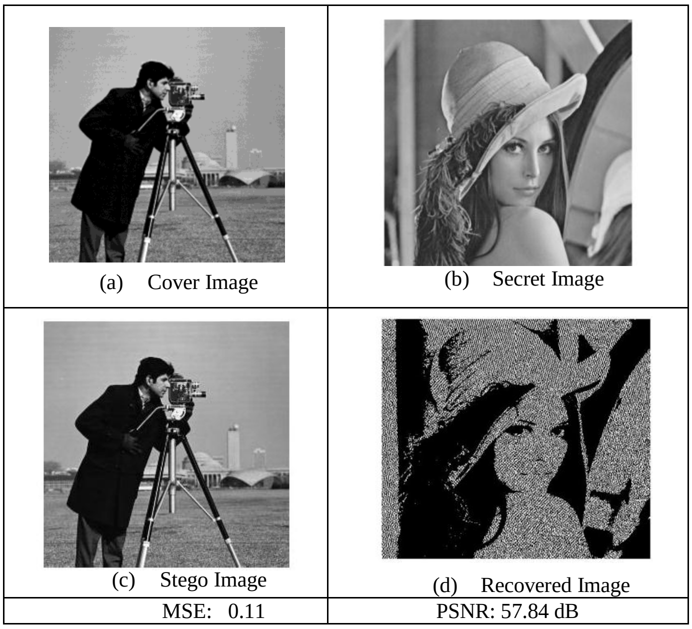

(c) Stego Image

(d) Recovered Image

MSE: 207.27	PSNR: 24.99 dB

Figure 7 Cover image Pool and Secret image as Lena using LSB method

In figure 7, again simple LSB based steganography technique is discussed. Here the cover image is pool (a) and secret image is Lena (b). The stego image is shown in figure (c) which is very much similar to the cover image and in figure (d) recovered secret image is shown. In the simple LSB scheme MSE is found to be 207.27 and PSNR value as 24.99 dB.


(a) Cover Image

(b) Secret Image

(c) Stego Image

(d) Recovered Image

MSE: 0.11	PSNR: 57.84 dB

Figure 8 Cover image as cameraman and Secret image as Lena using LSB method

In figure 5.5, again simple LSB based steganography technique is discussed. Here the cover image is cameraman (a) and secret image is Lena (b). The stego image is shown in figure (c) which is very much similar to the cover image and in figure (d) recovered secret image is shown. In the simple LSB scheme MSE is found to be 0.11 and PSNR value as 57.84 dB.

It is clear from above two figures (7-8) the secret image and covered image both plays in the overall PSNR. It is also noticeable that the transmitted image is stego image thus it should be of high quality so that it can be re-processed at the receiving end. However, the quality of the recovered image is also very important.

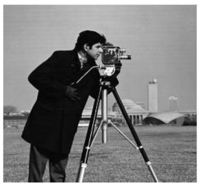
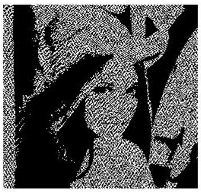
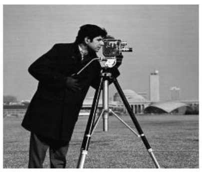
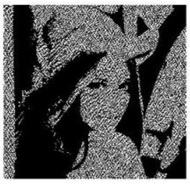
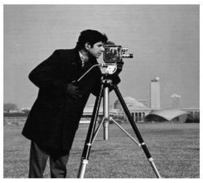
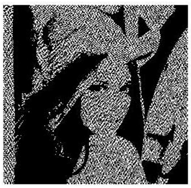
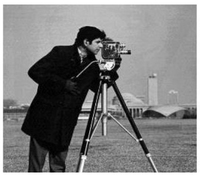
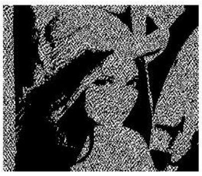
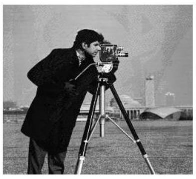
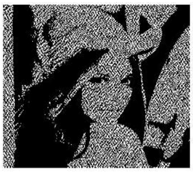
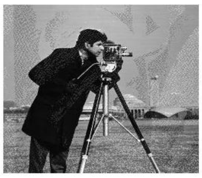
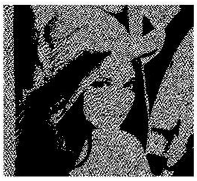
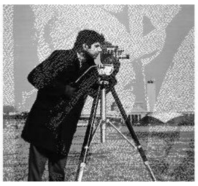
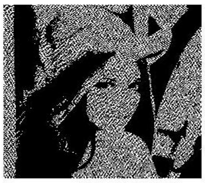
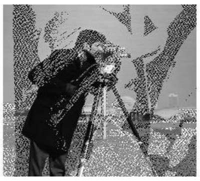
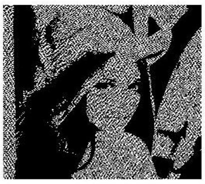
		57.84 (LSB)
		51.817 (LSB+1)
		45.675 (LSB+2)
		39.015 (LSB+3)
		35.150 (LSB+4)
		34.135 (LSB+5)
		38.826 (LSB+6)
		33.829 (LSB+7)

Figure 9 Cover image as cameraman and Secret image as Lena using proposed method

In figure 9, proposed technique is shown. Here the cover image is cameraman (a) and secret image is Lena (b). Here 8 images are shown, and secret image is inserted form LSB and shifts one bit right towards MSB. With this combination of cover and secret image PSNR degrades as we move from LSB to MSB. It is also noticeable that moving from left to right corrupt the secret image and for (LSB+3) a trace of Lena image can be found in the background. This effect increases from left to right and for (LSB+7) the Lena image is clearly visible in the background.

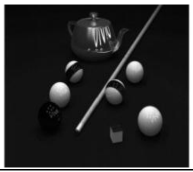
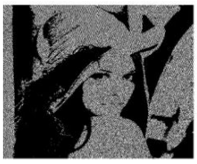
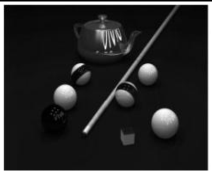
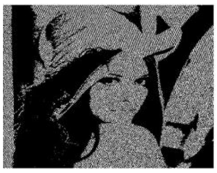
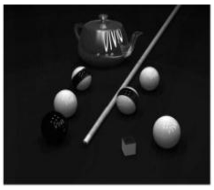
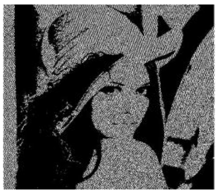
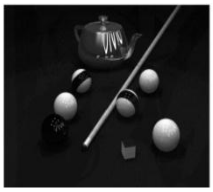
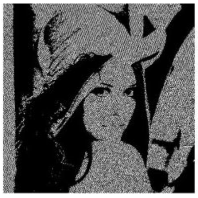
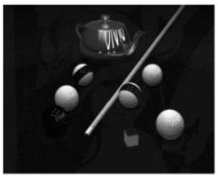
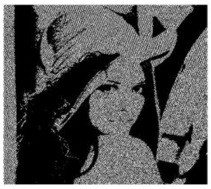
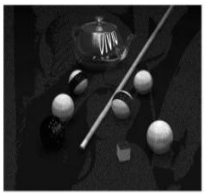
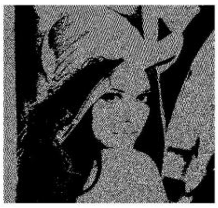
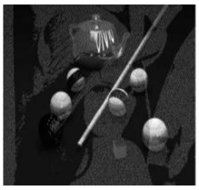
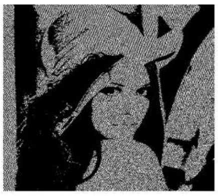
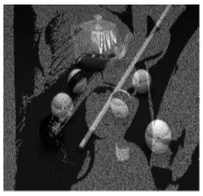
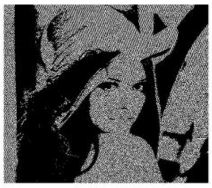
		24.999 (LSB)
		25.001 (LSB+1)
		25.033 (LSB+2)
		25.01 (LSB+3)
		25.386 (LSB+4)
		26.318 (LSB+5)
		26.330 (LSB+6)
		26.348 (LSB+7)

Figure 10 Cover image as pool and Secret image as Lena using proposed method

In figure 10, proposed technique is shown. Here the cover image is pool (a) and secret image is Lena (b). Here 8 images are shown, and secret image is inserted form LSB and shifts one bit right towards MSB. With this combination of cover and secret image PSNR improves as we move from LSB to MSB. Again it is also noticeable that moving from left to right corrupt the secret image and for (LSB+3) a trace of Lena image can be found in the background. This effect increases from left to right and for (LSB+7) the Lena image is clearly visible in the background. Therefore a possible choice can be (LSB +2) with PSNR level of 25.03 dB and good quality of stego image.

In the above two figures (9-10) we have shown that by inserting secret message other than LSB can be beneficial in some cases and better PSNR can be obtained but it should be kept in mind that the stego image should not superimposed with the secret image. It is also noticeable that in image processing no technique is ideal as some performs well on some images and some not.

5. CONCLUSIONS AND FUTURE WORKS

This paper presents a novel concept of bit shifting positioning of secret image insertion using Inverted LSB mechanism. Inverted LSB mechanism can very easily embed the secret information's. We have shown that in LSB both cover image and secret images are very important. For each cover and secret image combination PSNR varies. The proposed method can reduce distortion and thus increases PSNR.

Pixel adjustment method can be used to further enhance the PSNR. Pseudo random bit encoding can be used. Inverted LSB along with watermarking can be used to further enhance the security. However, the combination of more than one scheme can further reduce PSNR.

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

I would like to thank, first and foremost, my parents, who taught me how to learn, always encouraged me to pursue my interests and never failed to support me in any endeavor.

I wish to express my deepest gratitude to Mrs. Shubha Jain (HOD, CSE), [Kanpur Institute Of Technology,Kanpur] for providing me a platform to conduct research, and for supporting and guiding me throughout my studies.

I am also grateful to my supervisor Mrs. Preeti Chaudhari (Assistant Professor), [Kanpur Institute Of Technology,Kanpur] for her guidance, inspiration and constructive suggestions that helped me in the preparation and execution of this dissertation.

REFERENCES

[1] Chan, C.K. and L.M. Cheng, "Hiding data in images by simple LSB substitution, "Pattern Recognit., Vol. 37, pp. 469-474,2001.

[2] R. Anderson and F. Petitcolas, "On the limits of steganography" IEEE Journal of Selected Areas in Communications, Vol. 16, No. 4, 1998.

[3] Adrian G. Boris and Ioannis Pitas, "Image watermarking using block site selection and DCT domain constraints", Optics Express, Vol. 3, No. 12, pp.512-523,1998.

[4] Amirtharajan, R., D. Adharsh, V. Vignesh and R.J.B. Balaguru, "PVD blend with pixel indicator-OPAP composite for high fidelity steganography," Int. J. Comput. Applic., Vol.7, pp.31-37,2010.

[5] Amirtharajan, R., K. Nathella and J. Harish,. Info hide: A cluster cover approach. Int. J. Comput. Applic., Vol.3, pp.11-18,2010.

[6] Amirtharajan, R., S.K. Behera, M.A. Swarup, K.M. Ashfaaq and J.B.B. Rayappan, "Colour guided colour image steganography," Universal J. Comput. Sci. Eng. Technol., Vol.1, pp.16-23,

[7] Fridrich, J., M. Goljan and R. Du, "Detecting LSB steganography in color and gray-scale images," IEEE Multimedia, Vol.8: pp.22-28,2001.

[8] Hmood, A.K., H.A. Jalab, Z.M. Kasirun, B.B. Zaidan and A.A. Zaidan,, " On the capacity and security of steganography approaches: An overview., " J. Applied Sci., Vol.10, pp.1825-1833,2010.

[9] Provos, N. and P. Honeyman, "Hide and seek: An introduction to steganography,"IEEE Secur. Privacy, 1: 32-44, 2003.

[10] Qin, J., X. Xiang and M.X. Wang, "A review on detection of LSB matching steganography, "Inform. Technol. J., 9: 1725-1738,2010.