Region Based Data Hiding For High Payload

Shreela Dash[#1], Mamatarani Das[*2], Kartik Chandra Jena[#3]

[#]*Department of Computer Science and Engineering*
CV Raman College of Engineering, Bhubaneswar, India

Abstract— **Steganography is the process of hiding a secret message within a larger one in such a way that someone cannot know the presence or contents of the hidden message. The purpose of Steganography is to protect the hidden information from the outside world and the security will increase if the secret information can't be retrieved even if the attacker has the idea of embedding method. The proposed technique is a new approach which is a combination of region based edge adaptive LSB steganography with Huffman coding. In this method some pixels of secret information are hidden in the edge pixels of RGB colour image in each of the three colour components by performing XOR on least significant two bits of edge pixels and other pixels of the secret information are hidden in the LSB of non-edge pixels. As a result it is difficult to retrieve the hidden information because the hidden information is kept in two different categories of pixels. It has been shown by the experimental results that the quality of image is not degraded even after the addition of secret data.**

Keywords— **Pixel, Edge Detection, LSB, Steganography, Stego Image, Huffman Coding.**

I. Introduction

Internet is the most commonly used communication channels prevalent today. But as often happens with any widely used communication channel, there is perceptible threat to security. To maintain this security, two most widely cyber experts are Cryptography and Steganography.

In cryptography, the data is first encrypted and sent to the receiver. The receiver has decrypted the message to decipher the data. But the limitations are obvious, as an intruder may try and guess to break it with different techniques.

In the other hand, steganography is the art and science of writing the hidden information in an multimedia carrier, that prevents the detection of hidden messages. Steganography differs from cryptography that, the secret message is hidden in the cover medium in such a way that only sender and receiver knows about it. In other words it is invisible to the others than the sender and receiver. The covering medium can be text, audio, video or still image.The hacker finds it difficult to differentiate between original image and stego image, so difficulty in guessing about secrecy. If, even anyone knows the presence of secret information inside stego image, then without knowing the embedding methodology, no one can extract the secret information.

Image steganography techniques are of two type's i.e. spatial domain steganography and frequency domain steganography. In spatial domain the information is hidden by directly manipulating the pixels of the image but in frequency domain secret data are hidden in the frequency components of the image by using Discrete Fourier Transformation and Discrete Cosine Transformation.

For any kind of steganography, either it may be spatial domain or frequency domain, security i.e. the ability of an eavesdropper to know the hidden information easily, capacity i.e. how much data that can be hidden inside a cover medium and robustness i.e. resistance power to modify or destroy the unseen data, are important aspects.

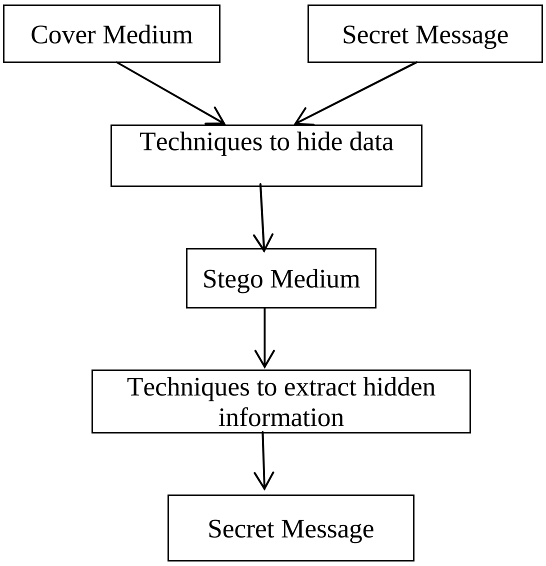


Fig 1: Basic Steganography Technique

Watermarking is the most important application of steganography.Watermarking can be treated as a protecting technique, which protects or claims the owner's property right for digital media (the digital media may be image, software, music or video etc.) by some hidden watermarks. When we compare steganography with digital watermarking, steganography wish to communicate the data in a completely unidentified manner, whereas the goal of digital watermarking is the cover object itself.

Many research works have been carried out on the area of steganography, like **a**. *Steganography by embedding message inside an image* **b**. *Steganography in frequency* *domain* **c**. *steganography in spatial domain* [1]. But the general steganography technique is shown in the figure 1.

In steganography, the cover medium may be audio file, video file or image. The hidden information may be text, audio, video or image. In our approach we have taken still image as cover medium, also another small still image as secret message. The stego medium is obtained from cover medium, by applying some steganography embedding techniques, is another still image, which is totally equivalent to cover image, that cannot normally be distinguished by human eye. Similarly, from the stego image we can recover the hidden information, by applying some steganography extraction techniques. Our proposed work focuses a novel data hiding and extraction technique that is robust, and provides high security with maximum embedding capacity.

The rest of the paper is organized as follows. Section II describes the related work, Section III explains the working of proposed algorithm, Section IV contains simulation results and Section V analyses the result of proposed algorithm, with other techniques and in Section VI conclusion is given.

II. Related Work

Many steganography methods have been proposed in spatial domain and LSB method is the most common replacing technique for hiding secret data as it gives high embedding capacity.

A LSB based image steganography is described in [3], which suggests the secret data is hidden in the least significant bit of RGB image. In this method secret data bit is hidden in the least significant bit of 3 colour planes Red, Green and Blue individually by directly substituting LSB. It gives more capacity for storing secret data.

Jarno Mielikainen described a steganography method in [4] for embedding message bits into a still image. In the Least Significant Bit Matching (LSBM), the choice of whether to add or subtract one from the cover image pixel is random. The new method uses a binary function of two cover pixels to the desired value. The embedding is performed using a pair of pixels as a unit, where the LSB of the first pixel carries one bit of information, and a function of the two pixel values carries another bit of information. Therefore, the modified method allows embedding the same payload as LSB matching but with fewer changes to the cover image. The experimental results of this method show better performance than traditional LSB matching in terms of distortion and resistance against existing steganalysis.

Another LSB based steganography is proposed in [5]. Here the LSB within the cover image of each byte is overwritten using the secret key. In this method the secretkey and the red component matrix of RGB image for decision making to replace the hidden information into either green matrix or blue matrix. Each bit of secret key is XOR with each LSB of red matrix. The result tells whether 1 bit will be placed in LSB of green or blue or blue matrix.

In Q.Huang [6], proposed the problem in LSB Matching Revisited (LSBMR) algorithm to make regions selection on images to find suitable area. By counting on each pixel we can decide if it should be protected. It can improve the visual imperceptibility and detectability of the LSB matching method. By adjusting the parameters of neighbour pixels, the max embedding capacity can be increased as needed.Object based steganography includes edge based steganography. Theedge based steganography schemes emphasizes on PVD i.e pixel value differencing [7] to differentiate between sharper edge and smooth pixels. If the secret bits are hidden in the sharper edges, then it is less sensitive to visual distortion.

In [8] chaos based edge adaptive steganography is given. The given approach uses Canny’s edge detection algorithm for locating edge pixels and data is hidden in the edge pixels. Canny’s edge detection can detect almost all edges including weak and strong edges [9].First the secret data is scrambled using chaotic cat map. The scrambled message is hidden in the edge pixels but not directly. More secrecy is incorporated by doing XOR operation. But the limitation is that only 2 secret bits are hidden in the first edge pixel of 3 colour components red, green and blue. So capacity to hide secret data is compromised.

Another technique is proposed in [10] which provide better embedding capacity with more security. The input text is converted to binary bits first and then it is fragmented into 2 parts.1st part is embedded in the edge pixels using 3-bit LSB technology and 2nd pair in the LSB of non-edge pixels It improves in security as some bits of secret message are in edge pixel and some bits are in non-edge pixel.

By Rig Das[1], the authors proposed a Huffman Encoding based image steganography. For better security, they used the technique called four tier storage procedure of “size of Huffman encoded bit stream”. By this scheme, the secret image is first encoded by Huffman encoding scheme. Then the number of bits is compressed up to 2 bits, in different tires. Now these two bits are embedded in last or first pixels in two LSB positions of a cover image. This is a better approach that provides more security as well as more robust.

A Nag[11], authors described a novel technique for image steganography in frequency domain, based on Block DCT, where DCT is used to transfer original image(cover image) blocks from spatial domain to frequency domain.

Here a grey level image is converted to number of blocks and 2D DCT is performed in each block. Huffman encoding is performed on secret image and each bit of Huffman code of secret image/message is embedded in frequency domain by altering the LSB of each of the DCT coefficients of cover image blocks. This algorithm provides maximum capacity due to Huffman coding, with good invisibility.

III. Proposed Work

Least significant bit (LSB) based steganography is a well-known and simple approach for embedding information in a cover image. In LSB technique, the secret message bits replace the bits in cover image. To increase the capacity, we can use two or more LSBs for embedding the data but this will affect the quality of the stego image. But, there is a trade-off between embedding capacity and quality of stego image. So researchers have combined LSB with other characteristics to make stego image more resilient towards attacks and detectability. Embedding secret data into specific region within an image is used now a day to provide better security. The specific region which provides least distortion after embedding can be used. Edge pixel, skin pixel, corner pixel are the different regions of interest (ROI).If edge pixels are taken to hide information; the technique is called Edge adaptive steganography. In our proposed technique edge region is chosen for data hiding. Here only the prominent edges are taken as they are less sensitive to distortion compared to other edges and also provide random pixel position for data hiding.

In this paper we proposed a spatial domain technique to hide large number data. The method uses region based edge adaptive data hiding that provides high security and good invisibility, with Huffman coding for better capacity and no loss of data of secret message. Before going to discuss proposed method in detail, let us discuss some techniques, which we had taken for comparison with our work.

A. Edge Adaptive steganography [8]

The question here is how much secret data is communicated through cover medium, without distorting cover medium (with high PSNR we can achieve this). Maximum embedding capacity indicates better the algorithm. Here the authors embed secret bits at the rate of two bits per pixel. As a result, the capability of this method to hide data is very less compared to other technique. The embedding capacity is shown in table 1:

TABLE 1
EMBEDDING CAPACITY OF EDGE ADAPTIVE STEGANOGRAPHY

Cover Image	Secret Image	Data Hiding Possible?
Leena (512 X 512)	Kid(35 X 50)	Yes (PSNR 74.7)
	Pout(76 X 79)	No
	Rabbit(214 X 214)	No
	Cameraman(310 X 310)	No

B. Region Based Edge Adaptive Data Hiding[12]

Here the RGB image is fragmented into two sections. First section consists of edge pixels and second section consists of non-edge pixels. For finding the prominent edges, Canny's edge detector algorithm is used, with supplying some threshold. The secret data bits are hidden in all edge pixels and also in prominent or strong edges. The strong edges are chosen, as any changes done in these edges it produces less distortion.

The authors suggest that more size of data can be embedded by this approach, because secret bits of three bits per pixel are embedded. The PSNR calculated is also high. As security is important in steganography, so this method gives more security, as the secret data is kept in different regions of image i.e. some of the pixels are hidden in edge region and some of the pixels are hidden in non-edge region.Since this method is better compared to edge adaptive steganography, but it fail in some cases as shown in the table 2:

TABLE 2
EMBEDDING CAPACITY OF REGION BASED DATA HIDING

Cover Image	Secret Image	Data Hiding Possible?
Leena (512 X 512)	Kid(35 X 50)	Yes (PSNR 73.5)
	Pout(76 X 79)	Yes (PSNR 68.1)
	Rabbit(214 X 214)	Yes (PSNR 59.3)
	Cameraman(310 X 310)	No

So authors propose a method, where the embedding capacity is high and communication through internet is more secure. It uses the concept of compressing the secret data (audio, video or image) before passing it for embedding. In table2 we see that the secret image of size 310 x 310, cannot be hidden in cover image of size 512 x 512.To overcome this difficulty,at sender end, we compress our secret image first (by Huffman encoding technique), and then the compressed data is embedded in cover image with region based edge adaptive data hiding technique (figure 2). At receiver end, by extraction technique, we extract the hidden information from stego image and this extracted information is decoded by Huffman decoding technique to recover the secret data(figure 3).

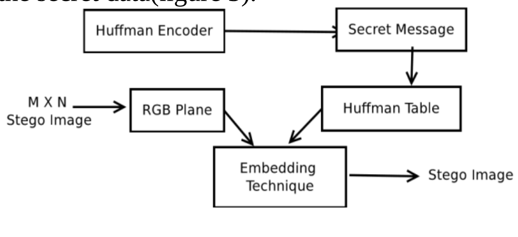


Fig 2: Embedding Technique

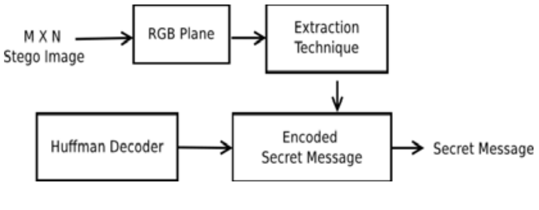


Fig 3: Extraction Technique

C. Huffman Encoding and Decoding

Huffman coding is an entropy encoding algorithm used for lossless data compression in the field of computer science and information theory. It is based on the frequency of occurrence of a data item (pixel in images).Huffman codes are optimal codes that maps one symbol to one code word. Codes are stored in a Code Book (CB) can be constructed for each image or a set of images. In all cases the code book plus encoded data must be transmitted to enable decoding[13]

D. Technique

In this method, RGB image of M x N is fragmented into 2 sections. 1st section consists of only edge pixels and 2nd section consists of non-edge pixels. For finding edge pixels Canny's edge detection algorithm is used as it can detect more number of edges. Huffman encoding is used for lossless compression of secret message. Consider we have a grey scale secret image of size 35 x 50 of size 14000 bits, after compression, the size of this image is 12794bits and it is shown in table 4. Then this encoded secret data bits are hidden not in all edge pixels but only in the prominent /strong edges. Any change done in the stronger edges will produce less distortion. Rest of the encoded secret bits are hidden in the LSB of non-edge pixel.

The proposed steganography algorithm consists of 2 phases. Phase I describes the embedding algorithm and phase II describes the extraction algorithm. The proposed work uses the following condition table.(Table 3).

Let P_i be the ith pixel from set of prominent edge pixels. P_{iR}, P_{iG} ,P_{iB} be the Red(R), Green(G) and Blue(B) components for the pixel P_i. Take 3 bits of secret message at a time , say B_1,B_2,B_3. The embedding technique can be implemented as per the following listing and it represents the condition of embedding 3 bits of secret message in the Red, Green and Blue component of the cover image.

TABLE 3
EMBEDDING CONDITION LIST

Condition	Action
$B_1=P_{iR7}\oplus P_{iR8}$	No Change
$B_1\neq P_{iR7}\oplus P_{iR8}$	Invert P_{iR8}
$B_2=P_{iG7}\oplus P_{iG8}$	No Change
$B_2\neq P_{iG7}\oplus P_{iG8}$	Invert P_{iG8}
$B_3=P_{iB7}\oplus P_{iB8}$	No Change
$B_3\neq P_{iB7}\oplus P_{iB8}$	Invert P_{iB8}

1) Embedding Algorithm

The embedding algorithm uses Huffman encoding technique in RGB image, with region based edge adaptive data hiding, along with the combination 3-bit LSB technique. The generated stego image by the embedding algorithm is then sent to the receiver for extraction of secret message.

Embedding Algorithm

```
Input: Cover Image C, Secret Image M
Output:Stego Image S

Step 1: Extract Red(R), Green(G), and Blue(B) components from the cover image C.
Step 2: Extract edge pixels from the cover image by taking Canny Edge detector.
Step 3: The secret image M is converted to grey scale image and stored in 1D array GM.
Step 4: Huffman encoding technique is applied to 1D array GM to find code book CB.
Step 5: Strong edges are selected for hiding secret information.
Step 6: Let the prominent edges be Pe.
Step 7: Assign count:=1.
Step 8: Repeat Step i to iv for Pe/3 times.
      i.   Set Pi := i^th pixel value from the set of pixels selected in step 5.
      ii.  Let $a_1$ := Red plane LSB of Pi.
           a2 := Green plane LSB of Pi.
           a3:= Blue plane LSB of Pi.
      iii. Embedding on the bits $a_1$, $a_2$ and $a_3$ are performed according to the condition list given in Table 3.
      iv.  Set count := count + 3.
Step 9: Rest of the secret bits from GM are hidden in the LSB of the non-edge pixels.
Step 10: Save the stego image S in either png or bmp file or send it to the receiver.
```

A. Working principle of Embedding

Consider the secret message M and after applying Huffman encoding to M, we get the encoded secret message GM. The Red, Green and Blue component of Pixel 1 and Pixel 2 is shown below:

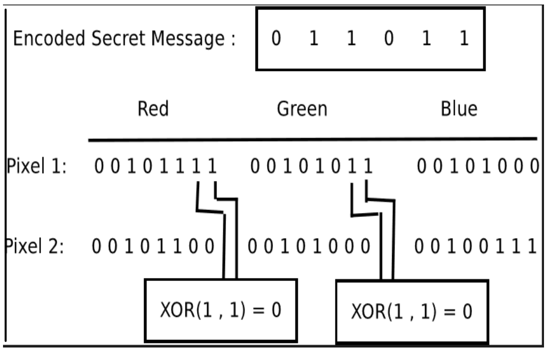


Fig 4: Embedding Example

For Red component, XOR of two LSB's is 0(as shown in figure4), and most significant bit of secret message is 0. So there is no change and the corresponding Red component of pixel 1 is 0 0 1 0 1 1 1 1. For Green component, XOR of two LSB's is 0(as shown above in the box) and the second most significant bit of secret message is 1. As it is a change, the green componentof pixel 1's LSB is changed from 0 to 1 and corresponding pixel is changed from 0 0 1 0 1 0 1 1 to 0 0 1 0 1 0 1 0. Applying the above procedure, we get the pixel 1 and pixel 2 of stego image S, after embedding the secret message GM is :

Pixel 1 : 00101111 00101010 00101001

Pixel 2 : 00101100 00101001 00100110

2) *Extraction Algorithm*

To recover the hidden image from stego image, first the stego image S is divided into three matrices Red(R), Green (G), Blue (B). Using the reverse process of embedding, encoded secret image GM is generated. Huffman decoding scheme is applied to GM to get our original secret image M.

Extraction Algorithm

```
Input: Stego Image S.
Output: Secret Image M.

Step 1: Extract Red(R), Green(G), and Blue(B) components from the encoded stego image GM.
Step 2: Extract edge pixels from GM by taking Canny's edge detector.
Step 3: Let the prominent edges be Pe.
Step 4: Set count := 1.
Step 5: Repeat steps i to iii for Pe/3 times.
    i.   Set Pi:=i^th pixel value from the set of pixels selected in step 3.
    ii.  Perform XOR operation on the two LSB's of Red(R), Green(G) and Blue(B) matrix to generate the secret bits B1, B2 and B3.
    iii. Store the bits in one dimensional array SA and set count:=count+3.
Step 6: The next secret bits are taken from the least significant bits of non-edge pixels.
Step 7: Take the 1D array SA and apply Huffman decoding scheme to SA and we get 1D array and it is arranged properly to generate the secret image.
```

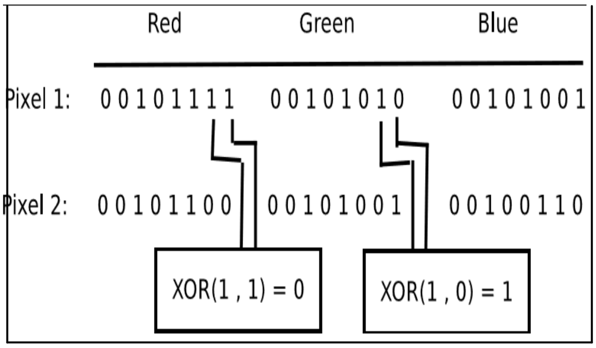
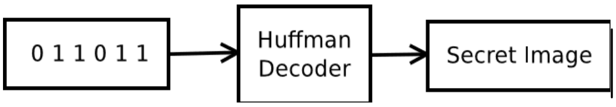

Red
Green
Blue
Pixel 1: 0 0 1 0 1 1 1 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 1
Pixel 2: 0 0 1 0 1 1 0 0 0 0 1 0 1 0 0 1 0 0 1 0 0 1 1 0
XOR(1 , 1) = 0
XOR(1 , 0) = 1

Fig 5: Extraction Example

B. *Working principle of Extraction*

Consider the encoded stego image GM and the Red, Green and Blue components of pixel 1 and pixel 2 is shown in figure 5. By XOR-ing the bits of LSBi and LSB_{i+1}, we successfully recover our secret image as 0 1 1 0 1 1 and this secret image is the encoded secret image. By applying Huffman decoding scheme, we get our original secret image M.



IV. Experimental Results

The proposed algorithms in Phase I and II of the preceding section have been implemented in MATLAB7.0 and tested on a 32-bit, 2.94 GHz dual core processor computer. The embedding algorithm has been tested for the level of visual distortion produced and time taken to embed, security of the secret message and capacity of cover image to hold the amount of secret information. So for the comparison of original image and stego image 3 parameters are taken i.e i) *PSNR* ii) *Capacity* iii) *Execution time to embed*. The quality between the cover image and stego image is calculated using PSNR (Peak Signal To Noise Ratio) value and PSNR is used to get the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale(dB) [14]. PSNR is defined as:

$$\boldsymbol{PSNR = 10log_{10}(255^2 /MSE)\ dB}$$

Where MSE is defined as:

$$MSE = 1/(M * N) \sum_{i=1}^{M} \sum_{j=1}^{N} (xij - yij)\ (xij - yij)$$

Where, M and N are the horizontal and vertical pixel dimensions of the cover image, x_{ij} and y_{ij} are the pixel values in the cover image and stego image. The runtime behaviour of the algorithm is measured in terms of the time required to embed the secret message. The proposed algorithm is implemented with the several standard cover images. Here we show only the result of four pictures Leena, Fruits, Butterfly and Car. The secret images used in our technique is Kid, Pout, ,Rabbit and Cameraman. In our method, the output of the edge detection depends on threshold value supplied to it. To find prominent edges, the edge detector is running with threshold value between 0.70 to 0.85. For different cover images the threshold value varies.The threshold value was already specified by the sender have to be communicated with the receiver prior to starting the secret communication. Many solutions are there, one of which is sending them via email, or handing them over the phone, or injecting them at the end of the carrier image, or embedding them into some predefined pixels locations in the carrier image using the LSB technique. In other techniques of edge based steganography complete edge pixel information need to be provided but in our proposed technique only threshold value is sent ,so it is more secure and takes less time to execute. The secret images used in our technique is shown in figure 7.

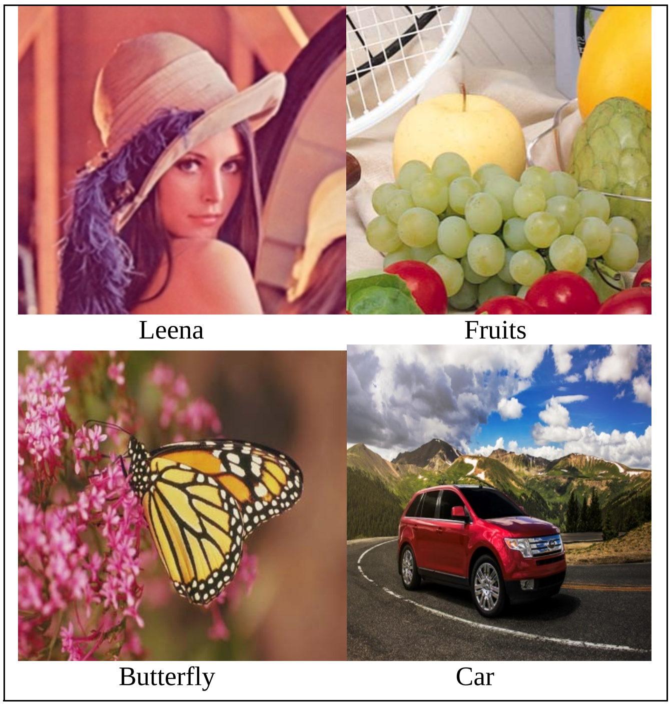

Leena Fruits
Butterfly Car

Fig. 6: Cover Images

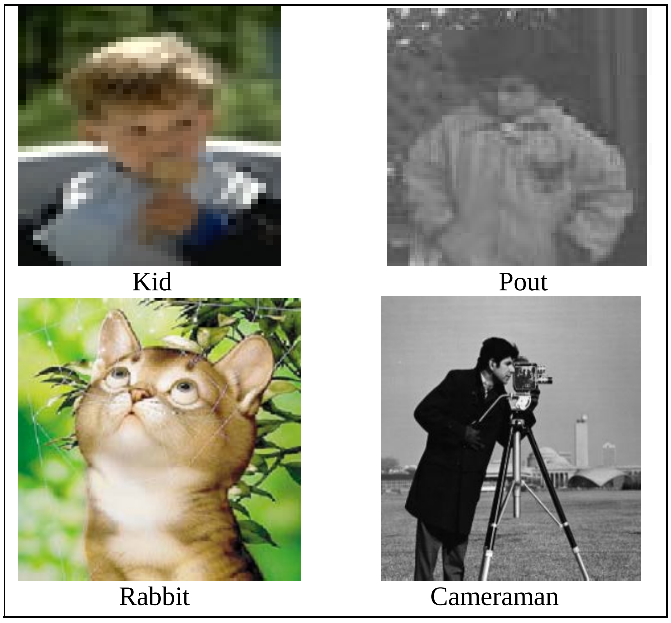

Kid Pout
Rabbit Cameraman

Fig.7 Secret Images

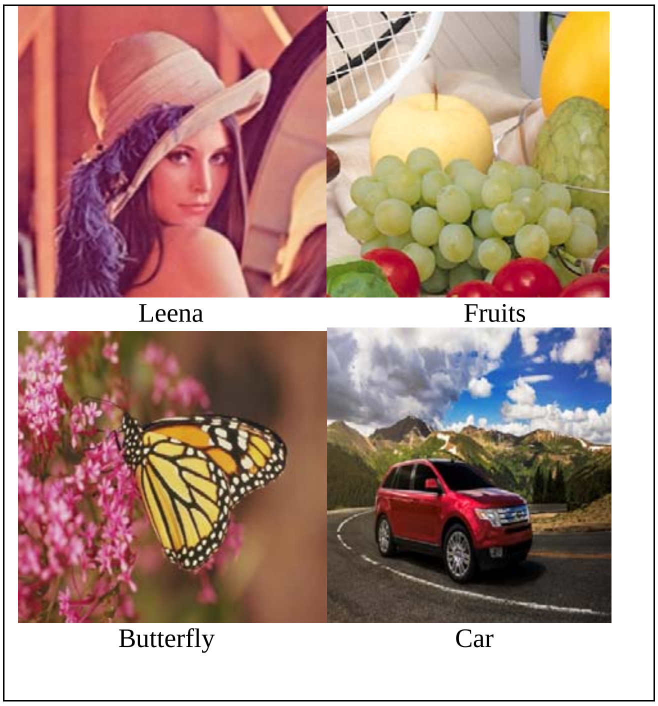
Here we can observe no visual distortion is produced. All the original and corresponding stego images are identical. The algorithm is executed for cover image and secret image of different size, then PSNR is calculated and shown in Table 5.

TABLE 4
IMAGE SIZES BEFORE AND AFTER HUFFMAN ENCODING

Secret Image Name	Size (Height X Width)	Number of Bits to Hide	Number of Bits after Huffman Encoding
Kid	35 X 50	14000	12794
Pout	76 X 79	48032	38239
Rabbit	214 X 214	366368	347895
Cameraman	310 X 310	768800	682401

TABLE 5
PSNR COMPARISON WITH EXISTING METHOD

Cover Image	Secret Image	Peak Signal To Noise Ratio(PSNR)	
		Region Based Edge Adaptive Data Hiding(12)	Proposed Method for Data Hiding
Leena (512 X 512)	Kid	73.5	74.0
	Pout	68.1	69.1
	Rabbit	59.3	59.5
	Cameraman	**Not Possible**	56.6
Fruit (629 X 794)	Kid	76.2	76.7
	Pout	70.9	72.0
	Rabbit	62.0	62.3
	Cameraman	58.6	59.3
Butterfly (768 X 512)	Kid	75.2	75.5
	Pout	69.7	70.9
	Rabbit	60.9	61.2
	Cameraman	57.8	58.2
Car (512 X 512)	Kid	77.4	78.0
	Pout	72.1	73.1
	Rabbit	63.4	63.6
	Cameraman	60.1	60.6



Leena

Fruits

Butterfly

Car

Fig.8 : Stego Images

V. Result Analysis

The proposed steganography algorithm is compared with Region Based Data Hiding [12]. The comparison is done on the basis of PSNR and capacity to embed. From the Table 5, it is noticeable that in all the approaches, PSNR varies directly with the size of the image. This is due to the fact that, with decrease in the size of the image the number of pixels is also decreased. Therefore more number of pixels are used to store the secret message as the message size is constant.

We observe that the proposed method has advantage over Region based and Edge adaptive steganography that is its capability of hiding more data bits. In this approach we are embedding secret bits 3 bits per pixel after Huffman encoding of the secret message. In region based edge adaptive steganography, secret message is embedded only three bits per pixel. In Edge based steganography secret bits are embedded at the rate of 2 bits per pixel. So it is possible to embed more secret bitsbut with nearly equal PSNR value. Another important requirement of steganography is security. The proposed method gives more security as compared to others because the secret data is kept different regions of image. Here some of the pixels are hidden in the edge region and rest of the pixels is hidden in the non-edge region. Therefore it is difficult for the hacker to retrieve the secret message. Therefore the proposed method can be considered as a better approach over the existing method when bigger size images are to be hidden.

VI. Conclusion

Use of encoding schemes provides the proposed work to increase the efficiency in terms of security and embedding capacity. The proposed method is an efficient way to embed hidden information without significant visible distortion. It is difficult for the unauthorized users to identify any change in stego image. This proposed method gives more security as secret data is divided into two parts and first part is hidden in the edge pixel and second part is hidden in the non-edge pixels, so it is difficult for the intruder to retrieve the original secret message. When compared with other method the proposed method achieves higher embedding capacity with high PSNR value.

References

[1] Rig Das, ThemrichonTuitthung, " *A Novel Steganography Method For Image Based on Huffman Encoding*", IEEE, 2012.

[2] Abbas Cheddad, oan Condell, Kevin Curran, Paul McKevitt, F. Hartung and M. Kutte "*Information hiding-a survey*" Proceedings of the IEEE: Special Issue on Identification and Protection of Multimedia Content, Volume: 87 Issue:7, pp. I062-I078, July. 1999.

[3] Chin-Chen Chang, Min-Hui Lin, Yu-Chen Hu "*A fast and secure image hiding scheme based on LSB substitution*" International Journal of Pattern Recognition and Artificial Intelligence, 2002, vol. 16, no. 4, pp. 399-416.

[4] JarnoMielikainen, "LSB matching revisited," IEEE signal processing letters, vol. 13, no. 5, May 2006.

[5] S. M. MasudKarim, Md. SaifurRahman, Md. Ismail Hossain "*A New Approach for LSB Based Image Steganography using Secret Key*" Proceedings of 14th International Conference on Computer and Information Technology (ICCIT) 2011.

[6] Qinhua Huang and WeiminOuyang, "Protect Fragile Regions in Steganography LSB Embedding", 3rd International Symposium on Knowledge Acquisition and Modelling, 2010.

[7] G.KarthigaiSeivi, Leon Mariadhasan, K. L. Shunmuganathan "*Steganography Using Edge Adaptive Image"* International Conference on Computing, Electronics and Electrical Technologies [ICCEET} 2012,pp 1023-1027.

[8] Ratnakirti Roy, AnirbanSarkar, SuvamoyChangder "*Chaos based Edge Adaptive Image Steganography"* international Conference on Computational Intelligence (CIMTA),2013, vol 10,pp.138-146.

[9] John Canny, "*A Computational Approach to Edge Detection*" IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. -8, No. 6, November 1986.

[10] Nidhi Grover, A.K. Mohapatra "*Digital Image Authentication Model Based on Edge Adaptive Steganography*" Second International Conference on Advanced Computing, Networking and Security 2013,pp. 238-242

[11] A Nag, S Biswas, D Sarkar, P P Biswas,"A Novel Technique for image steganography based on block DCT and Huffman Encoding", IJCSIT, 2010.

[12] Shreela Dash, Mamatarani Das, Nilamadhab Dash" *A New Approach For Region Based Edge Adaptive Data Hiding*" International Journal of Scientific & Engineering Research, Volume 5, Issue 3, March-2014

[13] Mamta Sharma,"*Compression Using Huffman Coding*" IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.5, May 2010,page 133-141 .

[14] http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio