A New Approach for Canny Edge Detection Method to Hiding Data in Digital Images using HSI Color Model

Avneesh Chandra Chack
Department of Computer Science
& Engineering
Shri Shankaracharya Technical Campus
Junwani, Bhilai (C.G.)
ranu1in@gmail.com

*Prof. Rajesh Kr. Tiwari
Department of Computer Science& Engineering
Shri Shankaracharya Group Of Institution
Shri Shankaracharya Technical Campus
Junwani, Bhilai (C.G.)
raj_tiwari_in@yahoo.com

Abstract

The objective of this work is to use new steganography techniques in HSI color digital images, which conceals secret data on the edges of the carrier images using the 2-bit LSB substitution to embed and calculate runtime and PSNR. The subjected (canny edge detection) technique has been used for obtain real edges. The amount of data that will be integrated plays very significant role in the selection of the edges. The main advantage of using the HSI color mode is that it produces an image with a significantly huge size of file so the technique used by us have better performance and its capability have better embedding because it hides huge amount of secret message as per result of this experiment.

Keywords: *HSI color model, True-Edge, 2-bit LSB, Canny Edge Detection, Hide data.*

1. INTRODUCTION :

Steganography is a word which is derived from the words ancient Greek, the cover of importance of the stegos and the meaning of spelling that compose both characterize it as insured composition. Image steganography is just to hide the data in the images. The science and art of secret communication is steganography. The secret communication is ended by coding or incorporating such secret data, so much so that the invisibility of the presence of data is there. The original files are called the cover image, cover page text, or cover page audio. After the insertion of secret messages it is alluded as stego-medio. By using stego-key, the process of hiding or encoding to confine the location or extraction of embedded data. Today, digital communication needs has increased tremendously and this helps to come on the web has turned out to be an extremely critical means for more effective and faster communication to digital communication. Meanwhile, the data that are accessible on the web have proved to be more powerless to make you copyright infringement, protection, espionage, and so on. That requires secret communication. Therefore, a new domain dedicated to data security has evolved and this is also called data hiding. The novel idea steganográfica in the field of hidden information that goes back its history previous to its origin.

At present, steganography categorize into different media such as audio, video, images or text files to secretly hide any information in it therefore does not illustrate any interest and therefore looks like a safe medium. Digital image, video, audio and photo become the first option as a means of coverage. Stego is the medium that helps to contain secret data while the cover media is the plain file. These days, images become a more common decision as a method to cover the medium, mainly because of their over-talking and the ability to overcome applications in our day to day. In the last couple of years, numerous calculations are as a research topic. In our work, we have built another system for steganography in RGB images. The data are covered in the images in the vector space. The cover image is selected and the secret message is inserted.

The information is covered secretly at the edges that are chosen progressively in light of the size of the content. The proposed strategy is free of ancillary drawings as it uses the 2LSB procedure. The 2LSB strategy is free of basic assault. The three most vital classes of steganography strategies are space-area strategy, space-of-recurrence technique, and versatile strategies.

The adaptive strategy can be used both in the spatial area and in the space of recurrence and is also considered as an extraordinary area. The two broad groups in image steganography are spatial space and space of recurrence. In space space, hidden data are used to directly incorporate into the power of the pixels found in the space-area technique, with the help of a stenographer made the alteration of the hidden secret message and the cover document, which demonstrates the change in the least significant bit phase.

This approach is less convolute and is also better when contrasted with the other two types of techniques, whereas because of the area of recurrence, the image is first changed in area of recurrence or swap space and then the covered message is secretly embedded in the coefficients of change. There are different types of numerous applications for information hiding.

Data concealment methods can not be easily classified into any of the watermark or steganography categories, so there are so many similarities between watermark or steganography terms. Hence the different application of these two terms based on the application of the algorithm. Therefore, instead of sorting between them, the most common data that hide the applications are: cover communication, fingerprints, copyright protection, secure storage and secret communication.

1.1 CLASSIFICATION

In image steganography systems, the following classifications are frequently used independently of the algorithm by which they are executed.

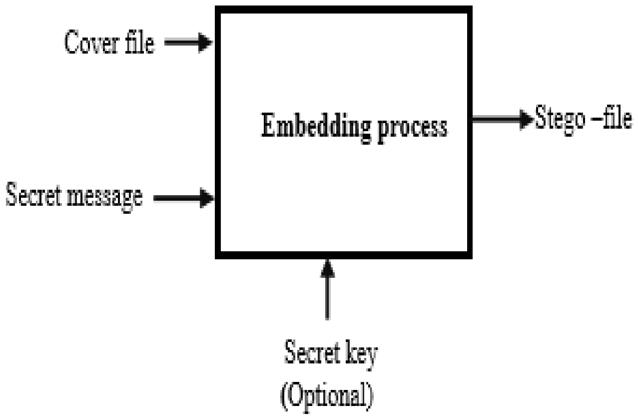

Fig. 1.1. A Generic Scheme of Steganography Embedding Model

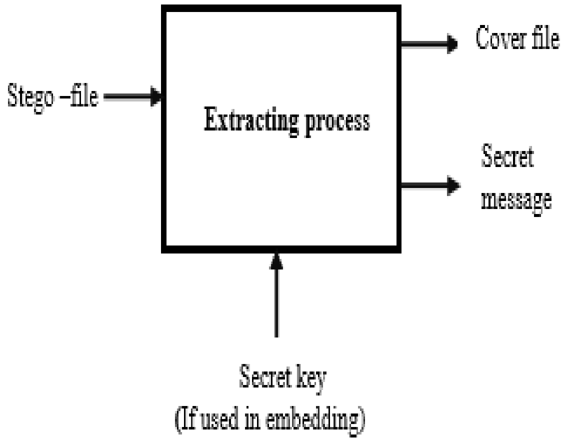

Fig. 1.2. A Generic Scheme of Steganography Recovery Model

2. EDGE DETECTION METHOD

2.1 INTRODUCTION

A border is illustrated as the points in an image where the brightness changes rapidly. Edges are substantial local modi fi cation in the intensity of an image. They are treated as the boundaries between several image segments. Image processing, computer vision and machine vision generally require the edge detection process as a very important tool, mainly in the area of feature detection and feature extraction, since the edges are the main characteristics for the analysis of the information contained more necessary in an image. The methodology for acquiring significant transitions in an image is known as edge detection. Stains where abrupt changes in brightness usually occur from the boundaries between different separate objects. Many classic edge operators are now available in the image processing literature. As

1. Sobel Edge Detector

2. Prewitt edge detector

3. Edge Detector Robert

4. Gaussian laplacian edge detector (Log)

5. Canny Edge Detector

6. Fuzzy Edge Detector

2.2 CANNY EDGE DETECTOR

The main objectives of the Canny Edge Detector are:

2.1 Good detection - There must be a lower probability of not marking the actual points of the edge, and also less likely to falsely mark non-marginal points. Therefore, both probabilities are decreasing functions of predictability of the output signal-to-noise ratio, this criterion means maximizing the signal-to-noise ratio. Therefore, we need to mark as many real borders as possible.

2.2 Good location - The points marked as edge points by the operator should be as close as possible to the center of the true edge. In essence, the marked edges should be as close to the edges at the actual edges as possible.

2.3 Single response: only one response to a given edge. This is implicitly captured in the first criteria, therefore, when there are two main responses to the same edge, one of them should be considered as false. Therefore, the idea is that a certain border should be marked only once, and image noise should not create a false border detection.

The most important thing about the canny border detector is that it specifically has three features for which it is primarily deployed in machine vision, computer vision and image processing to find a sharp intensity conversion and object boundaries in an image. They are:

All the most necessary edges are conserved, false edges are not taken into account and in the mean time the magnitude of the error detection must be low.

A minimum distance must be maintained between the actual position and the actual edge position. There is only one response to a given edge.

In the case of a canny edge detector operator, a pixel is taken to be an edge pixel, if the magnitude of a particular pixel's gradient is more important than the pixels on both sides thereof and in the direction of maximum intensity modification. The procedure for implementing the Canny Edge Detector is summarized in the following steps:

First, the image is smoothed by applying a Gaussian filter with a fixed standard deviation, to reduce noise. (ρ).

The magnitude of the gradient g2 x + g2 and y the edge direction tan-1 (gx gy) must be calculated at each point. A point whose force is close to the maximum in the direction of the gradient is defined as an edge point.

3. IDENTIFICATION OF PROBLEMS

This paper provides the description of the existing approach available for image steganography and later in this section, the problem identification is discussed.

3.1 Existing system: This section describes the current and existing system for image-based steganography.

3.1.1 Quick Stego: It is a software that allows us to hide text in images. The same software must be available at the other end to decode the text of the images. The image can be saved, uploaded to the website or can be sent by email. It is the best security software in the USA. One of the main disadvantages of this software is that it is less accurate. While decoding text messages, the retrieved content may be different from what is stored.

3.1.2 Hide image (HIP): It also allows storing text in images. But it only uses bitmap images. You can set your password to hide text in it.

3.1.3 Chameleon: It is also a steganography software that allows to hide confidential information in images. Especially used to communicate over the Internet.

TABLE. I. Comparison of several stenographic programs

S/W	Algorithm	Usability	Speed	Security	Accuracy
Quick Stego	Advanced Encryption Standard (AES)	Easy	Very fast	Secure	Less
Hide in Picture	Blowfish	Very Easy	Fast	Secure	High
Chamel-eon	Least Significant Bit (LSB)	Hard	Slow	Less Secure	Least

4. Problem Identification

Steganography is an ancient theme. Many authors have proposed several solutions to the problem of encrypting the secret messages in the image so that the intermediate person can not read it. Consider two people Alice and Bob. They want to communicate with each other so that the third person can not understand what they are talking about. So they used the steganography method, hiding secret messages in Pictures or any other way.

There were many problems encountered in several research papers. Hiding information using the LSB technique is not efficient since the quality of the output image decreases significantly. The PSNR also plays a very important role in the encryption of text messages. The higher the PSNR, the higher the image quality. So we will also focus on the quality of the image after the inclusion process. Then, to hide text messages, many papers use random pixel steganography. Random steganography of pixels has many drawbacks as it is based on the pixel value as its name implies. The area is not sufficient to hide the text data in the images due to the pixel-based transformation. This problem is also addressed in our project. Most existing image security systems are out of date to protect against the latest rape attacks. Therefore, we have proposed an effective and robust image security framework designed especially for images. Therefore, to the sum of all there are many problems before in the method of steganography. They have to deal seriously. In this project, we demonstrate image steganography with the Threshold selection algorithm based on Canny Edge Detection.

5.METODOLOGY

This section introduces a new edge-based Steganography architecture. This section describes several steps related to the processing and hiding of text in images. Section 5.1 describes the complete system architecture of the text concealment process.

5.1 METHODOLOGY

This section describes the different phases of implementation of Image Steganography. Several steps involved in image steganography, such as pre-processing of images, selection of threshold, hiding of secret messages or embedding and decoding of secret messages. These steps are illustrated in Figure 5.1.

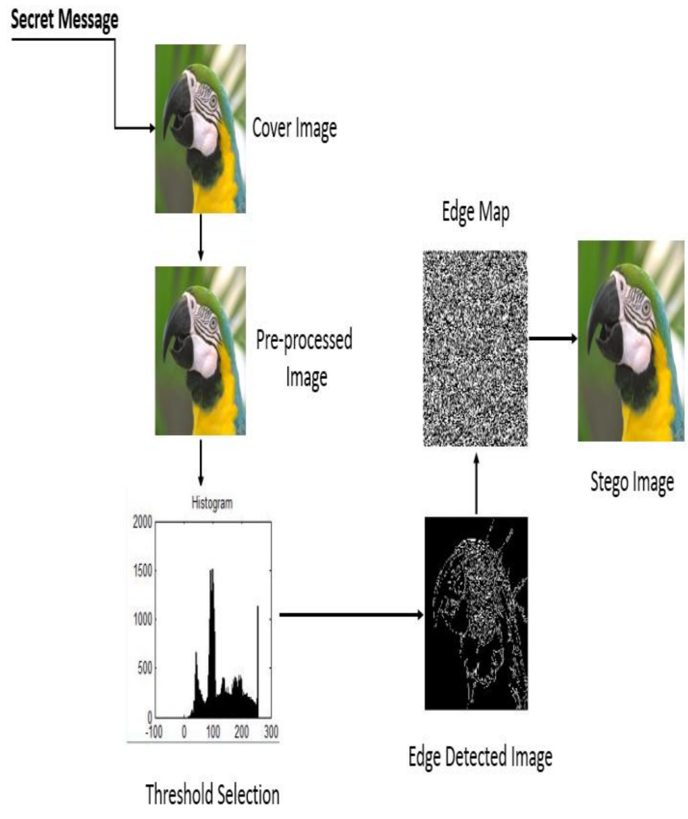

Fig. 5.1. Shows the sender side image steganography system architecture

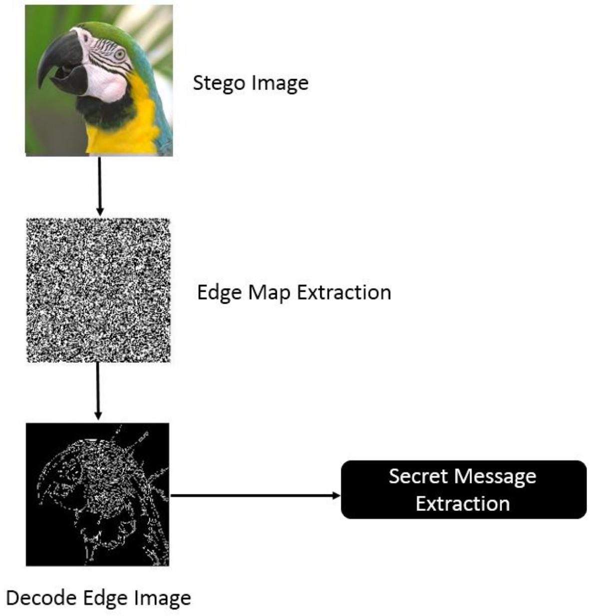

Fig. 5.2. Shows the receiver side image steganography system architecture

Several steps involved in the realization of steganography image. The steps are described in later sections. For threshold selection, getThreshold () is used and edge detection is based on the Canny Edge Detection algorithm.

5.2 PICTURE OF THE COVER

The cover image is the image in which the information is hidden. The information can be any size. The higher the image quality, the higher the information can be stored. The capacity of the images also depends on the edge. The number of edges is more the ability to hide the images increases. The cover image is the entrance to our process. The image of the peacock is taken as input. It is of TIF format.

Fig. 5.3. Peacock RGB Image, TIF format, 256 X 256 size

5.3Secret Message

The secret message is the information that must be stored secretly to be transferred to another person so that the third person can not read it. The length of the secret message depends on the edge and the quality of the image. The more the image quality the more will be the storage capacity.

Hide information also depends on the edge detection algorithm. If there are a lot of borders present, then obviously the ability to maintain information also increases.

Fig. 5.4. Shows the message to embed into image

5.4Pre-Processing

There were several methods used for converting RGB to HSI image. It is the simplest. The conversion is carried out by applying the following algorithm steps:

1.Read the RGB color images and represent the RGB image in the range [0,1].

2. Look for HSI components:

$$\theta = \cos^{-1}\left(\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)^2]}\right)$$

3. H (Hue): if (B <= G) then ϴ

if (B> G) then 360-ϴ

4. Calculate the saturation: $S\ (\text{Saturation}) = 1 - \frac{3}{R+G+B}[\min(R,G,B)]$

5. Calculate the intensity:

$$I(\text{Intensity}) = \frac{1}{3}(R+G+B)$$

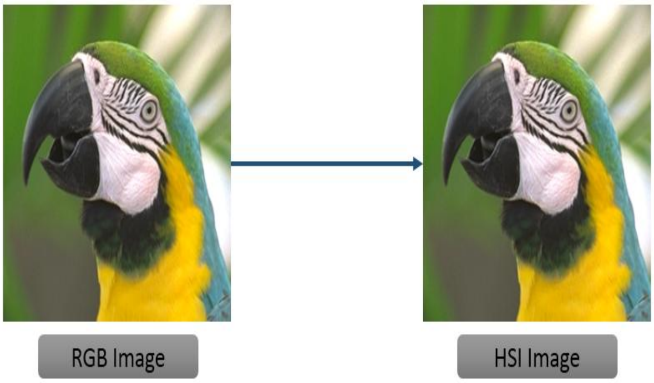

Fig. 5.5. Shows the conversion of RGB to HSI Image using 6 Steps Algorithm

5.5 Threshold Selection

The threshold value is returned by the Canny Edge Detection algorithm, ie, high threshold, low threshold. All of these parameters are used to identify the edges in the cover images. The high threshold parameter describes that there are strong edges present and a low threshold describes that there are relatively low edges present in an image. Based on the size of the secret message, the threshold is adjusted. If the message is too log, then the high threshold value is selected so that the message can be accommodated at the edges effectively. Several effects of change in the threshold value are shown in Fig. 5.6.

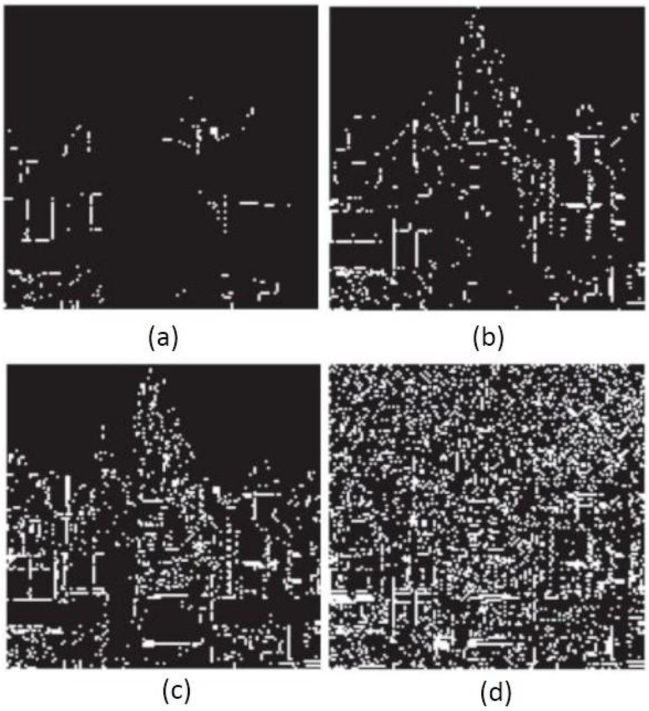

Fig. 5.6. Effects of threshold. (a) Threshold 0 (b) Threshold .25 (c) Threshold .50(d) Threshold .75

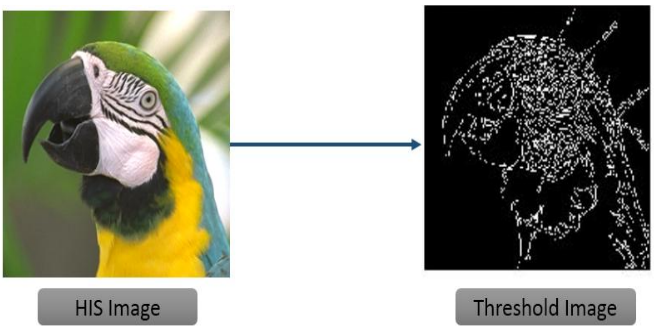

Fig.5.7. Shows the extraction of threshold form HSI image

Algorithm for Getting Threshold: getThresholdValue(I, N, w)
Data: I: Image, N: Length of augmented message to beembedded, w: width of the Gaussian Kernel
Result: threshold: *threshold th* for Canny to get N pixels

/* limit is set to 1% of the messagelength*/

/*no. of edge pixels,$ne \leq N + 0.01 \times N$*/

and $ne \geq N$

/*ne = number of edge pixels in I, whenCanny edge detector is used on I withhigh threshold th and low threshold*/

$tl = 0.4 * th$ and w

limit $\leftarrow$- $0.01 \times N$;

*threshold*_max ←- 1;

*threshold*_min ←- 0;

set ←- *false*;

repeat

th ←- [(*t*max + *t*min)]/2;

ne ←- getEdgePixelCount(Canny(*I*, *t*h, *t*l, *w*));

/*it returns the number of pixels inthe edges obtained through Cannyedge detector*/

diff ←- *ne*-*N*;

if *diff >limit* **then**

threshold_min←- *t*h;

end

else if *diff <0* **then**

threshold_max←- *t*h;

end

else

set ←- *false*;

end

until *set = true*;

return *threshold (th)*

5.6 Edge Map

The border map is used to represent the vector fields of an image. Vector fields can be represented in the border map using the getEdgeMap method. It provides the continuous stream of vector mapping input in sequence order. It provides input as well as output points that can be useful for determining the content of the image. Edge Mapping is a very robust and error-free technique for storing image points in it. In our work, we have used the border map to store the edges of the images to hide the secret message in it. This will help to use the full capacity of the image to store the information that is the transfer to another person in secret. Figure 5.8 shows the construction of the edge map.

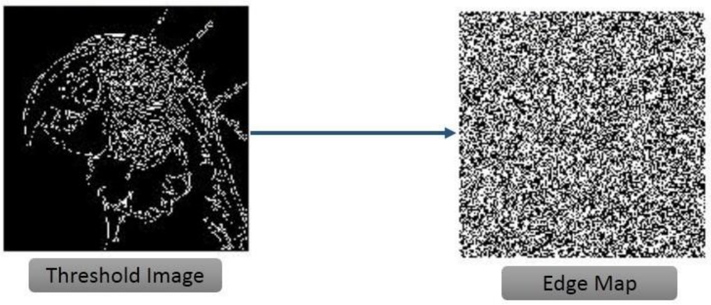

Fig. 5.8. Shows the construction of Edge Map

5.7Embedding of Message

Most of the steganographic techniques described use the LSB technique. However, the LSB technique is not as efficient in performing hidden data in images. Therefore, we have used the 2-bit LSB technique to ensure that the hidden data in the images efficiently. Structural detectors can easily detect LSB text, but with the help of the 2LSB technique we can minimize the realization of performing by structural detectors.

Embedding of Secret Message: embedding(I, M, P, w)

Data: I: Image, M: Augmented message in binary, P: Stegokey, w: width of the Gaussian kernel
Result: S: Stego image

```
S ←- I;
I ← bitand(I,252);
L ←  |M|;
threshold ←- getThreshold(I, L, w);
e ←- CannyEdgeDetection(I, th, tl, w);
/*Shuffle eand S using Stego key P*/
e ←- randomPermutation(e,P);
S ←- randomPermutation(S,P);
index ←- 0;
for each edge pixel i in e do
Sx,y = bitand(Sx,y,252); // x,y are
co-ordinates of pixel i
Sx,y = Sx,y + 2*Mindex+1 + Mindex;
index ←- index + 2;
end
/* Embed threshold and width in non-edgepixels of S*/
e ←- CannyEdgeDetection(I, 0, 0, 0.1); // Pixels in e aremaximum number of edge pixels for a
given image
e' ←- complementOf(e); // Pixels in e' arenon-edge pixels
for i = 1: 16 in e' do
S (x,y)= bitANDoperation(Sx,y,254);
S(x,y)= Sx,y + threshold(i);
end
for i = 17: 32 in e' do
S(x,y)= bitand(Sx,y,254);
Sx,y = Sx,y +w(i-16);
end
```

Stego_Image←- randomPermute(S,P); // Reshuffle S toget Stego Image: S
return Stego_Image;

5.8 Stego Image

After you embed secret information in the image, it is ready to be transmitted to the recipient. It can be transferred via the Internet, email and by other means as well. The stego image is similar to the original image. There is no loss in image quality. Fig. 4.9. Shows the original image and stego image.

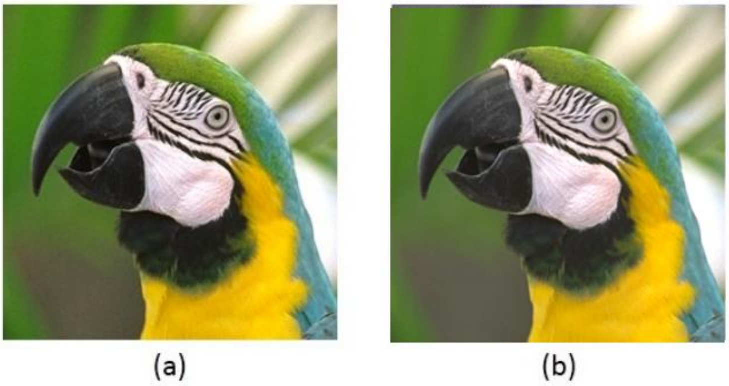

Fig. 5.9. Shows (a) Cover Image and (b) Stego Image

5.9 Receiver Side Decoding

After coding the secret message in the picture, the receiver-side decoder requires. The construction of the decoder is the same as the algorithm on the encoder side. If there is mismatch with any of the algorithms used on the encoder side, that decoding process can not be successfully performed. Fig. 4.10. Shows the reverse engineering approach in which the stego image is converted and processed using Edge Map Extraction.

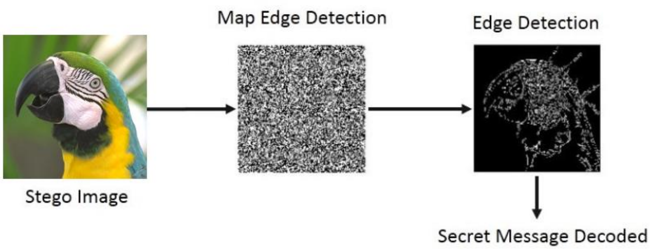

Fig. 5.10. Shows the decoding process of stego image

The algorithm involved in decoding of secret message is described in algorithm below.

Decoding of Secret Messages: decode: retrieve secret message
Data: I: stego image, T: Threshold, P: stego key, w: Kernelwidth
Result: Message: Secret message

S ←- I;

S ←- bitand(S,252);

threshold ←- T;

tl ←- 0.4 ∗ th;

e ←- Canny(S',th, tl, w);

e ←- randomPermutation(e,P);

// Shuffle S to get order of embedding

S ←- randomPermutation(S, P);

index ←- 0;

for each edge pixel i in e do

val ←- bitand(Sx,y,3); // x, y are co-ordinates

of pixel i

Mindex+1 ←- val mod 2;

val ←- val/2;

Mindex = val;

index ←- index + 2;

end

// extract first C bits to get message

size

msg_size ←- Message[1:C];

Message←- Message[C + 1 : msg_size];

return (Message);

6.Result and Discussions

6.1 Environment Setup

We have used MATLAB for evaluation and validation of our project. Image processing tool is used to pre-process and implement Canny Edge Detection algorithm.

TABLE II. Presents Various Tools and Method Used

Simulation Software Used	MATLAB
Simulation Software Version	Version 2013b
Tools Used	Image Processing Tool
Algorithm Used	Canny Edge Detection

Stego Key Algorithm	Symmetric Key Algorithm
Cover Image Format	JPEG, TIF, BMP, GIF,PNG
Image Type	RGB to HSI

6.2 Experiment Results

The Proposed methodology is tested and run successfully on MATLAB software. The input dataset is peacock.jpg file which is taken from the web. Various other images with different format are analyzed. The description are provided in subsequent sections. We have performed experiment and analyzed performance using PSNR metrics and time required for algorithm to run. The attributed of images are shown in fig. 5.1 and fig. 5.2.

name: Peacock
size: 256 X 256
format: TIFF

Fig. 6.1. Shows various attributes of peacock image.

name: Baby-Girl
size: 256 X 256
format: JPEG

Fig. 6.2. Shows various attributes of baby girl image.

We have analyzed image baby girl and peacock image using Canny Edge detection method by varying the threshold level. We have adjusted threshold in three levels. The output of threshold generated are given in TABLE II presents various metrics for validating performance of proposed steganography method.

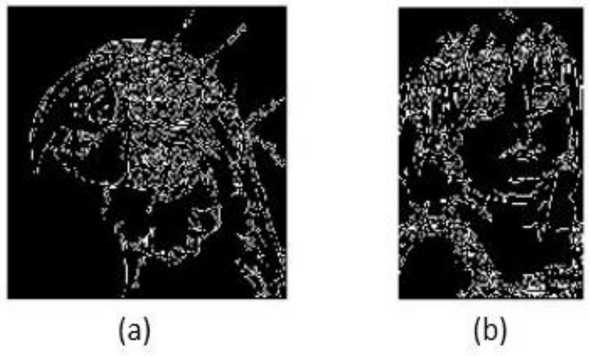

Fig. 6.3. Shows the threshold selection of two images peacock and baby girl respectively

6.3 metrics for measuring quality of image

Comparing restoration results requires a measure of image quality. Commonly used measures isPeak Signal-to-Noise Ratio. The formula is presented below for calculating PSNR value.

$$e_{MSE} = \frac{1}{MN} \sum_{n=1}^{M} \sum_{m=1}^{N} [\hat{g}(n,m) - g(n,m)]^2$$

$$\text{PSNR} = -10 \log_{10} \frac{e_{MSE}}{S^2}$$

Where S is the (max) pixel value. PSNR is measured in decibels (dB). PSNR is a good measure for comparing restoration results for the same image. The more is the dB the more will the quality of image. Hence 30 dB image look much better than 10 dB image. Table shows the comparative analysis of two images on the basis of time of execution and PSNR value.

TABLE I. Shows PSNR value and time required for execution of algorithm

Image	Size	Execution time	PSNR
Peacock.tif	256 X 256	21.321 sec	41.88908 dB
Baby-girl.jpg	256 X 256	22.543 sec	61.564 dB

7. Conclusion

In our work, we have developed a new technique for steganography in RGB images. The information is hidden in the images in the vector space. The cover image is selected and the secret message is embedded. The data is secretly hidden in the edges that are selected dynamically depending on the size of the text. The proposed method is free of structural attraction since it uses the 2LSB technique. The 2LSB technique is structurally free. Canny Edge detection is used to detect edges as a function of the threshold. We have defined several threshold levels for an image. For example, low and high threshold value. The higher the threshold value, the larger the space allocated by an image. Media with a higher threshold value we can hide more umber of textual information in the image.

7.1 Scope of Future Work

Due to the time limit, several interesting ideas have not yet been implemented. However, it will be worth a try in the future. One of the most important things I would like to improve is to apply steganography to 3D digital images using the Canny Edge Detection algorithm.

References

[1] Song, S., Zhang, J., Liao, X., Du, J. and Wen, Q., "A new secure communication protocol that combines steganography and cryptography", Procedia Engineering, vol. 15, pp. 2767-2772, 2011.

[2] Benlcouiri, Y., Ismaili, M., Azizi, A., and Benabdellah, M., "Securing images by secret key steganography", Applied Mathematical Sciences, vol. 6, no 111, p. 5513-5523, 2012

[3] Anand, D. and Niranjan, U., "Watermarked medical images with patient information", in Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th annual International Conference of the IEEE, vol. 2, pages 703-706, IEEE, 1998.

[4] Judge, J. C., "Steganography: past, present, future," tech. rep., Lawrence Livermore National Lab., CA (USA), 2001.

[5] Bin Sahib, S. and Zamani, M., "An Introduction to Visual Ganography Techniques,"

[6] Roy, R., Changder, S., Sarkar, A., and Debnath, NC, "Evaluating image teganography techniques: Future research challenges," in Computing, Management and Telecommunications (ComManTel), 2013 International Conference on, pp 309-314, IEEE, 2013.

[7] Chanu, Y., Tuithung, T., and Manglem Singh, K., "A Brief Survey on Image Stannanography and Stabilization Techniques," in Emerging Trends and Applications in Computer Science (NCETACS), 3rd National Conference 2012 on, pp. 52-55, IEEE, 2012.

[8] An overview of image gestanography by T. Morkel, J.H.P. Eloff, M.S. Olivier. Information and Computer Security Architecture (ICSA) Department of Computer Science University of Pretoria, 0002, Pretoria, South Africa

[9] Johnson, N.F. Jajodia, S., "Exploring Steganography: Seeing the Unseen", Computer Journal, February 1998

[10] Li, Y., Li, C.-T. and Wei, C.-H., "Protection of mammograms using blind steganography and watermarks," in Information Assurance and Security, 2007. IAS 2007. Third International Symposium on, pp. 496-500, IEEE, 2007

[11] "Detection of LSB steganography in color and grayscale images" Jessica Fridrich, Miroslav Goljan and Rui Du State University of New York, Binghamton

[12] KB Raja, Venugopal KR and LM Patnaik, "A Secure Stegonographic Algorithm using LSB, DCT and image compression on raw images", Technical report, Computer Science and Engineering Department, Visvesvarya University College of Engineering, Bangalore University, December 2004

[13] P. Kruus, C. Scace, M. Heyman and M. Mundy. (2003), "An overview of steganography techniques for image files." Advanced Security Research Journal.

[14] M.M. Amin, M. Salleh, S. Ibrahim, et al., "Information Hiding Using Steganography", 4th National Conference on Telecommunication Technology Proceedings (NCTT2003), Shah Alam, Malaysia, 2003

[15] An overview of data concealment in digital images by E Lin, E Delp Center for education and research Information security and security Purdue University, West Lafayette, IN 47907-2086

[16] Gonzalez, R.C., Woods, R.E., and Eddins, S.L., Digital image processing using MATLAB. Pearson Education India, 2004

[17] Akash Modi, Manu Bansal, "An Improved LSB Steganographic Data Hiding Algorithm", International Journal of Advanced Research in Informatics and Software Engineering, Volume 5, Number 5, May 2015.

[18] Yojna Chandel and Sunil Chhillar, "An overview of hiding reversible data", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, issue 9, September 2015.

[19] Arun Kumar Singh, Juhi Singh and Dr. Harsh Vikram Singh., "Steganography in Images Using LSB Technique", International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 5 Number 1 January 2015.

[20] S. Raveendra Reddy and Sakthivel S M, "A Fpga Implementation of Data Hiding Using Lsb Matching Method", IJRET: International Journal of Research in Engineering and Technology, Volume: 04 Issue: 03.

[21] Sandipan Dey, Ajith Abraham and Sugata Sanyal, "An LSB Data Hiding Technique Using Prime Numbers".

[22] Kede Ma, "Assessment of objective quality for image conversion from color to gray", International Electrical and Electronics Engineering (IEEE), Volume: 24 Number: 12 2015.

[23Arun Navjot Kaur and Manpreet Singh, "Changed approach with Lsb In Image Steganography", International Journal of Research -Granthaalayah, Vol.3 (Iss.5): May 2015.

[24] Deepak Tomarand S. Vijaylakshmi, "Steganography based on Singal Bit Data Hiding", International Journal of

Innovative Research and Studio (IJIRS),Vol. 3 Issue 4May 2014.

[25] Mukesh Garg and A.P. Gurudev Jangra, “An Overview of Different Type of Data Hiding Scheme in Image using Steganographic Techniques”,International Journal of Advanced Research in Computer Science and Software Engineering,Volume 4, Issue 1, January 2014.

[26] Saiful Islam,MangatRModi and PhalguniGupta, “Edge-basedimagesteganography”, EURASIPJournalonInformationSecurity a Springer open Journal,2014.

[27] Krati vyas and B.L.Pal,, “A Proposed Method In Image Steganography To Improve Image Quality With Lsb Technique”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 1, January 2014.

[28] Vandana Yadav, Sanjay Kumar Sharma,“ A New Approach for Image Steganography Using Edge Detection Method for Hiding Text in Color Images Using HSI Color Model”, International Journal of Scientitic Research in Science, Engimeering and Technology
Vol. 3, Issue 2, April 2017

[29] Deepesh Rawat and Vijaya Bhandari, “Steganography Technique for Hiding Text Information in Color Image using Improved LSB Method”, International Journal of Computer Applications (0975 – 8887) Volume 67– No.1, April-2013