Improved LSB based Steganography Techniques for Color Images in Spatial Domain

Mamta Juneja[1] and Parvinder Singh Sandhu[2]
(Corresponding author: Mamta Juneja)

University Institute of Engineering and Technology,Panjab University,Chandigarh, India[1]
Rayat and Bahra Institute of Engineering and Biotechnology,Mohali, India[2]
(Email: er_mamta@yahoo.com)

(Received Apr. 9, 2013; revised and accepted June 10, 2013)

Abstract

This research paper aims to propose a new improved approach for Information Security in RGB Color Images using a Hybrid Feature detection technique; Two Component based Least Significant Bit (LSB) Substitution Technique and Adaptive LSB substitution technique for data hiding. Advanced Encryption Standard (AES) is used to provide Two Tier Security; Random Pixel Embedding imparts resistant to attacks and Hybrid Filtering makes it immune to various disturbances like noise. An image is combination of edge and smooth areas which gives an ample opportunity to hide information in it. The proposed work is direct implementation of the principle that edge areas being high in contrast, color, density and frequency can tolerate more changes in their pixel values than smooth areas, so can be embedded with a large number of secret data while retaining the original characteristics of image. The proposed approach achieved Improved Imperceptibility, Capacity than the various existing techniques along with Better Resistance to various Steganalysis attacks like Histogram Analysis, Chi-Square and RS Analysis as proven experimentally.

Keywords: Adaptive LSB Steganography, AES, hybrid feature detection, random pixel embedding, steganography, two component based LSB steganography

1 Introduction

Steganography [25] is defined as an art and science of communicating in a way which hides the existence of the communication. In contrast to cryptography, where the enemy is allowed to detect, intercept and modify messages without being able to violate certain security premises guaranteed by a cryptosystem, the goal of steganography is to hide messages inside other innocent messages in a way that does not allow any enemy to even detect that there is a second message present. Three popular methods for message concealment in digital images are LSB insertion, masking and filtering and algorithmic transformations [22]. Initial work on LSB steganography was on LSB Substitution [5-10, 53, 55-56] which replaces the same length bits of each original pixel with the embedding data so were easily attacked by Chi-square Test [45, 51, 57]. LSB Matching [28, 30, 37, 41, 60] was also attacked based on the center of mass (COM) of the histogram characteristic function (HCF) [29].Adaptive LSB [26, 34, 36] is based on variable number bits substitution but doesn't not fully exploit the HVS masking characteristics; especially the Edge masking effect and they cannot obtain good imperceptibility. Pixel value differencing (PVD) method [24, 35, 38, 43, 54, 58-59, 62-63] follows the principle that the edge areas can tolerate more changes than smooth areas. However, this principle obeyed by some existing data hiding schemes does not discriminate texture features from edge ones; the edge areas used by these schemes contain both edges and textures. Moreover were complex to work and were easily attacked by scheme proposed in [63]. Edge detection Filter based technique [1, 3, 17, 19, 47, 48] was introduced for steganography in Gray images. But the advancement in image technology to RGB leads to steganography application for color images. Pixel indicator techniques introduced in [15-16, 44, 52] for color images had a major drawback of treating all color components (red, green, blue) equally contradicting Hecht principle, which reveals that the visual perception of intensely red objects is highest and then of intensely Green objects and is least for intensely blue objects i.e. red plays the most significant and Blue plays a least significant role in color formulation. So, we can integrate maximum changes in Blue component and average changes in green component and least change in red component without making much difference in color image. Color component based techniques [12, 21, 39, 46] were not fully tested fully for all types of attacks like targeted and universal and were focused on single component.

Hybrid filter based steganography [11] was tested for gray images moreover didn't test it for targeted and universal attacks. Capacity of 2.8 bits per pixel (bpp) is good but highest PSNR value attained is 28.6 which is very low. Similarly, DHPVD in [40] achieved 49% PSNR but didn't even mention capacity factor. Modified LSB and

PVD in [33] achieved 39% PSNR with good capacity but also explicitly mentioned this in their research paper that they targeted quality and capacity from tradeoff between capacity, quality and robustness. They compromised resistance to attacks for quality and capacity. Further, modifications in [20, 27] also worked on quality and capacity but was successfully attacked [49].

The three most required evaluation criteria's for any good steganography techniques are Robustness, Imperceptibility and Capacity. But there is lesser work done on techniques which would target all these criteria equally.

2 Proposed Approach

Objective of the Research: The Proposed research aims to develop an improved steganography approach for color images with higher imperceptibility/quality, large capacity/payload and better in robustness/resistance to attacks. It will incorporate cryptography to achieve high security and random pixel embedding to attain high immunity to attacks. It would be highly immune to any environmental disturbances like noise due to hybrid filtering. It is integrating the following for improved steganography:1) Hybrid feature (line /edge /boundary /shape) detection technique combining Canny and Hough transform for bifurcating an image into edge and smooth areas 2)Two Component based Least significant bit (LSB) Substitution Technique for hiding encrypted messages in edges of images 3)Adaptive LSB substitution technique for hiding messages to smooth areas.

In addition to above, Advanced Encryption Standard (AES) is used to encrypt hidden message text/file to provide better security by combining Steganography with Cryptography. With this combination even if message would be detected would not be understood by intruder due to encryption. Random pixel embedding technique is also been incorporated to hide data at random pixels in cover image so that couldn't be attacked by sequential attacks. This provides the two tier security to hidden data.

The proposed research is the direct implementation of the principle that edge areas being high in contrast, color, density and frequency can tolerate more changes in their pixel values than smooth areas, so can be embedded with a large number of secret data while achieving high quality of the stego-image. The various algorithms utilized in proposed system are as follows:

2.1 Hybrid Feature Detection technique for extracting Edge and Smooth Areas

The Proposed Hybrid Feature Detection technique for extracting edge and smooth areas from an image is based on combination of canny edge detection introduced in [4] and Enhanced Hough transform edge linking technique introduced in [18]

2.1.1 Edge detection using Canny

The Canny edge detector is widely considered to be the standard edge detection algorithm in the industry. It is known as optimal edge detector due its good detection, good Localization and minimal false edges. It uses a multi-stage algorithm to detect a wide range of edges in images.

Algorithm:

1. Smoothing: Blurring of the image to remove noise.
2. Finding gradients: The edges should be marked where the gradients of the image has large magnitudes.
3. Non-maximum suppression: Only local maxima should be marked as edges.
4. Double thresholding: Potential edges are determined by thresholding.
5. Edge tracking by hysteresis: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.

2.1.2 Line/Edge Linking/ Boundary/ Circle Detection using Enhanced Hough Transform

Edge Linking is implemented using global method named Hough Transform. Classical Hough Transform can locate regular curves like straight lines, circles, parabolas, ellipses, etc. Generalized Hough Transform can be used where a simple analytic description of feature is not possible. This research proposes an enhanced Hough transform by combining both classical and generalized Hough transform to extract lines, edge boundaries, circles and shapes.

Algorithm for Enhanced Hough Transform

a) Algorithm for Line detection: Edge detection is often used as preprocessing to Hough transform. The input image must be a thresholded edge image. The magnitude results computed by the canny operator can be thresholded and used as input. For an input image I of M × N:

1. Locate the HT coordinate system
2. Identify the ranges for θ and ρ. Let the range for for θ be between θl and θh, and the range for ρ between ρl and ρh.
3. Choose quantization intervals $\delta\theta$ and $\delta\rho$ for θ and ρ respectively.
4. Discretize the parameter space of ρ and θ using sampling steps $\delta\rho$ and $\delta\theta$. Let θd and ρd be 1D arrays containing the discretized θ and ρ.
5. Let A(T, R) be an array of integer counter; initialize all elements of A to zero, where $R = \rho h\text{-}\rho l/\delta\rho$ and and $T = \theta h\text{-}\theta l/\delta\theta$.
6. Let L(T,R) be an array of a list of 2D coordinates.
7. For each image pixel (c, r), if its gradient magnitude $g(c, r) > \tau$, where τ is a gradient magnitude threshold.

```
for i=1 to T
```

```
            ρ = c * cos(θd(i)) + r * sin(θd(i))
      find the index k, for the element of ρd closest to ρ
            increment A(i,k) by one.
            add point (c,r) to L(i,k)
```

8. Find all local A(ip, kp) such that A(ip, kp) > t, where t is a threshold.

9. The output is a set of pairs (θd(ip), ρd(kp)) and lists of points in L(ip, kp), describing the lines detected in the image, along with points located on these lines

b) Algorithm for Edge linking

1. Obtain a thresholded edge image

2. Specify subdivisions in the ρθ-plane.

3. Examine the counts of the accumulator cells for high pixel concentrations.

4. Examine the relationship (principally for continuity) between pixels in a chosen cell.

Continuity here normally means distance between disconnected pixels and a gap in the line can be bridged if the length of the gap is less than a certain threshold.

c) Algorithm for Circle detection

A circle in the xy-plane is given by (x − a)2 + (y − b)2 = c2

- So we have a 3D parameter space.
- Simple procedure:

```
set all A[a,b,c]=0
for every (x,y) where g(x,y)>T
    for all a and b
        c=sqrt((x-a)^2+(y-b)^2);
        A[a,b,c] = A[a,b,c]+1;
```

The proposed feature detector would firstly apply canny on input bitmap image whose output would be further refined by applying enhanced Hough Transform to overcome the flaws of canny.

Advantages of new hybrid feature detector:

- It is using probability for finding error rate.
- It is easier to locate various features like lines, edges, triangle, circles and boundaries with same accuracy and efficiency with this method.
- It helped to improve signal to noise ratio which was one of the main objective of this research.
- It is resistant to all kinds of noise, disturbances and provides good detection.
- It is tolerant towards gaps in the boundary line and occlusion in the image.
- It is robust to partial deformation in shape and can detect multiple occurrences of a shape in the same pass

2.2 Adaptive LSB substitution for Smooth Areas

We have used [26] and modified it for smoother areas. In this approach variable number of LSBs would be utilized for embedding secret message bits according to the mentioned algorithm:

For all components (RED, GREEN, BLUE) of each and every pixel of color image across smooth areas, the following embedding process is employed:

1. If the value of the current pixel say cpi, is in the range 240 <= cpi >=255, then we check for the message bit to be embedded:

   If it is 1 then we utilize the fifth bit of the pixel value.

   If the message bit is not 1 then we embed 4 bits of secret data into the 4 LSB’s of the pixel.

   This can be done by observing the first 4 Most Significant Bits (MSB’s). If they are all 1’s then the remaining 4 LSB’s can be used for embedding data.

2. If the value of cpi (First 3 MSB’s are all 1’s), is in the range 224 <= cpi >=239 then we check whether the message bit to be embedded is 0 then we utilize 5 bits of the pixel value. If the message bit is not 0 then embed 3 bits of secret data into the 3 LSB’s of the pixel.

3. If the value of cpi (First 2 MSB’s are all 1’s), is in the range 192 <= cpi >=223 then we embed 2 bits of secret data into the 2 LSB’s of the pixel.

4. And in all other cases for the values in the range 0 <= cpi >=192 we embed 1 bit of secret data in to 1 LSB of the pixel.

5. The embedding process maintains a matrix to keep a track of the pixels where 5 bits are utilized for embedding process. This helps in the retrieving the secret message. Further some more bits are added in lower bits depending on Hue, Saturation and Intensity values so that there is no visual difference in color after embedding.

Similar procedure is adapted for extracting the hidden text from the image.

2.3 Two Components based LSB Substitution Technique for Edge Areas

An Image is represented as arrays of values which represent the intensities of the three colors R (Red), G (Green) and B (Blue), where a value for each of three colors describes a pixel. Thus, each and every pixel is represented by three components(R-Most Significant Byte, G, and B-Least significant Byte). Here, a new LSB based image steganography method is introduced which focuses on Two components(Complete Blue and Partial Green) out of total three components of a pixel of RGB image during embedding of hidden in cover image. The selection of these components is based on significance of each component in visual perception of color image. Blue and green components are selected because a research was conducted by Hecht, which reveals that the visual perception of intensely red objects is highest and then of intensely Green objects and is least for intensely blue objects i.e. red plays the most significant and Blue plays a least significant role in color formulation. So, we can integrate maximum changes in Blue component and average changes in green component and least change in red component without making much difference in color image. Accordingly, all bits of Blue component and partial bits of green components and none of bits of red component are targeted in this proposed technique. In this method, the 8-bits of first component (blue component) of pixels of image would be replaced with secret text message bits followed by embedding of secret message to 4 least significant bit of green component.

2.4 Encryption Using Advanced Encryption Standard (AES)

It was invented by Joan Daemen and Vincent Rijmen, and accepted by the US federal government in 2001 for top secret approved encryption algorithms. It is also referred to as Rijndael, as it is based off the Rijndael algorithm. Reportedly, this standard has never been cracked. As explained in [50], AES is a block cipher. This means that it operates on fixed-length chunks of data (for example, blocks), applying the same transformation to each block. The transformation is controlled by use of the encryption key. Block ciphers (and thus AES) use symmetric keys, which mean that the same key used to encrypt data is also used to decrypt it (or in some cases, a key only trivially different). In operation, a user inputs 128 bytes of plaintext, along with a key, and receives as output 128 bytes of ciphertext. To decrypt the ciphertext, the user inputs it and the key to the algorithm to retrieve the original 128 bytes of plaintext. Encryption proceeds via a number of rounds. For 128-bit keys, AES prescribes ten rounds; for 192-bit keys, it uses 12 rounds; and for 256-bit keys, it uses 14 rounds.AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits. In contrast, the parent Rijndael algorithm can have both key and block sizes of 128, 160, 192, 224, or 256 bits. The 128 bits in a block are arranged in a grid of 4 x 4 bytes (also known as the state).

Algorithm: The algorithm begins with an Add round key stage followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption with the exception that each stage of a round the decryption algorithm is the inverse of its counterpart in the encryption algorithm. The four stages are as follows:

1) Substitute bytes: It carries byte-by-byte substitution during the forward process

2) Shift rows: It shifts the rows of the state array during the forward process

3) Mix columns: It mixes up the bytes in each column separately during the forward process.

4) Add round key: It adds the round key to the output of the previous step during the forward process.

The tenth round simply leaves out the Mix Columns stage. The first nine rounds of the decryption algorithm consist of the following:

- Inverse Shift rows
- Inverse Substitute bytes
- Inverse Add Round Key
- Inverse Mix Columns

Again, the tenth round simply leaves out the Inverse Mix Columns stage**.**

2.5 Random Pixel Embedding

To select the edge pixel randomly, a pseudorandom number generator (PRNG) will be used. Pseudorandom number generator as explained in [13] is an algorithm that generates a sequence of numbers, the elements of which are approximately independent of each other. The outputs of pseudorandom number generators are not truly random - they only approximate some of the properties of random numbers. To use a PRNG, it first requires a seed. Seeding is the technical term for giving it an initial value, from which it can shoot out a sequence. If a PRNG is given the same seed, then it will give the same set of numbers every time and the elements of which are approximately independent of each other.

3 Implementation of Proposed Approach

The proposed system comprises of two components:

1. Embedding Module

2. Extracting Module.

3.1 Embedding Module

Embedding is the process of hiding the embedded message generating the stego image. Hiding information may require a Stego key which is additional secret information, such as password, required for embedding the information. For example, when a secret message is hidden within a cover image, the resulting product is stego image (stego object).

The main algorithm for the Embedded stage can be listed as follow:

1. Input the secret text/image file that to be hidden in the cover image.

2. Select the cover image (BITMAP file) from list of stored Image files and the text files.

3. Extraction of Input cover image to Edge and smooth areas using new hybrid feature detection filter described in section 2.1 and Procedure 3.3.

4. Calculate the size of the secret text.

5. Secret text data is first encrypted using the standard Advanced Encryption standard described in section 2.4 and Procedure 3.4.

6. Substitute the encrypted secret characters from step 5 to cover image obtained from step 3 randomly explained in section 2.5 and Procedure 3.5.

   a) For edge areas, embed secret text data using New Two Component based LSB Substitution Technique described in Section 2.3 and Procedure 3.1.1.

   b) For smooth areas, embed secret text data using Adaptive LSB method section 2.2 described in Procedure 3.1.2.

3.1.1 Procedure for Embedding for Edge Areas

Extract all the edge pixels in the given image and store it in the array called Pixel-Array.

- Obtained Stego image will hide all input Characters.

3.1.2 Procedure for Embedding for Smooth Areas

For All RED-GREEN-BLUE components of all pixels

a) Calculate no. of bits available for embedding as given in Section 2.2

b) Place the remaining characters from character into available bits of pixel array.

Repeat Steps (a) and (b) till we reach end of character array.

3.2 Extracting Module

Extracting is the process of getting the embedded message from the stego image.

The main algorithm for the embedded stage is as follow:

- Extraction of Input cover image to Edge and smooth areas using new feature detection filter as in section 2.1 and Procedure 3.3.
- Extraction of secret text message from stego image is carried from random pixels of cover image using Function defined in section 2.5 and Procedure 3.5.
- For edge areas: Extract data from 8 bits of BLUE component and 4 least significant bits of GREEN component using section 2. 3 and Procedure 3.2.1
- For smooth areas: Extract data from adaptive no. of bits using Section 2.2 and Procedure 3.2.2.
- Apply AES Decryption method described in section 2.4 and Procedure 3.4.

3.2.1 Procedure for Extraction for Edge Areas

- Extract all the characters in the given text file and store it in the array called Character- Array.
- Extract all the characters from the stego key in key array.
- Choose first edge pixel and pick characters from Key-Array and place it in 8 bits of first component of pixel. If there are more characters in Key- Array, then place rest in the 4 bits of its second component and then to next pixel and so on till there are characters in key-array.
- Place some terminating symbol to indicate end of the key. ‘0’ has been used as a terminating symbol in this algorithm.
- Place characters of Character- Array in each 8 bits of first component (blue) and 4 bits of second component (green) of next pixels by replacing it and so on till all the characters has been embedded. Again place some terminating symbol to indicate end of data.

Extract all the pixels in the given image and store it in the array called Pixel-Array.

Now, start scanning pixels from Pixel-Array and keep extracting key characters from first and second (partial) components of all pixels to Key-Array till we get the terminating symbol. If this extracted key matches with the key entered by the receiver, then again start scanning next pixels and extract secret message characters from first (blue) and second (partial green) component of next pixels and place it in Character Array till we get terminating symbol.

3.2.2 Procedure for Extraction for Smooth Areas

For All RED-GREEN-BLUE components of all pixels:

a) Calculate no. of bits available for embedding as given in Section 2.2

b) Now, start scanning pixels from Pixel-Array and keep extracting characters from the no. of bits determined by step a in character array till we get terminating symbol.

3.3 Procedure for Feature Detection

1. The algorithm reads a 24-bit color image denoted by CI= {cp1, cp2, cp3, …, cpn-1} where cpi is the ith pixel in the image and n is the total number of pixels. In the same context, every color pixel cpi can be represented as cpi={rci[rc0,….rc7], gci[gc0,….gc7], bci[bc0,….bc7]}, where i is the index of the ith pixel, rci is the ith bit of color component rc, gci is the ith bit of color component gc, and bci is the ith bit of color component bc. Here CI is a 24-bit image so each of its pixels is made up of three color components each of which is of length 8 bits.

2. The algorithm converts CI into a Gray image denoted by f(CI)=GI where GI is gray image. The purpose of this conversion is to ease the processing of subsequent steps.

3. Three parameters 1) the size of the Gaussian filter, 2) a low threshold, and 3) a high threshold are automatically chosen where results of filter are optimal.
4. The Canny edge detection algorithm is executed on GI as described in section 2.1.1 using the three parameters selected in step 3. The results are a collection of lines, curves, and points denoting the edges or the boundaries of the objects in the image GI. The pixels that constitute the extracted edges are represented as EP= {ep1, ep2, ep3 …epr-1} where ej is the jth pixel that makes up the edges and r is the total number of these pixels.

Thereafter we apply Enhanced Hough transform to extract various other features like shapes lines and circles.

The output image after applying Canny edge detector is having distorted edges which are not properly joined with each other so edge linking is required to fill those edge gaps. Therefore a global edge linking technique i.e. Hough Transform has been applied on the output image obtained by canny so as to get more refined edges. Hough Transform would even detect line, edges, linkings and shapes which were not traceable through Canny. The edge pixels retrieved after applying Hough Transform given in Section 2.1.2 are represented as HEP= {hep1, hep2, hep3 …hepq-1} where hepj is the jth pixel in the image and q is total number of edge pixels.

3.4 Procedure for Encryption Using AES

Pseudo code of AES round transformation:

```
Round (State, ExpandedKey[i])
{
SubBytes (State);
ShiftRows (State);
MixColumns (State);
AddRoundKey (State, ExpandedKey[i]);
}
FinalRound (State, ExpandedKey [Nr])
{
SubBytes (State);
ShiftRows (State);
AddRoundKey (State, ExpandedKey [Nr]);
}
```

3.5 Procedure for Random Pixel Embedding

```
INPUT:  (Key, Seed)
OUTPUT: random_data, (Key', Seed')
  random_data = F(Key, Seed)
  Key' = F(Key, Seed+1)
  Seed' = F(Key', Seed)
  return random_data
```

4 Experimental Results and Comparison Analysis

4.1 Evaluation Criteria: Steganography Techniques are broadly Evaluated in Three Aspects

Criteria I: Imperceptibility/Quality

Criteria II: Capacity or Payload

Criteria III: Robustness or Resistance to Attacks

1. Imperceptibility / Stego-image quality:

It is the scale to measure the quality of stego image after hiding the details inside. It provides the imperceptibility and invisibility measurement and is highest if the differences in cover and stego image are not visible. As we all know, the higher the stego-image quality, the more invisible the hidden message. Therefore, the stego-image quality is a very important criterion to use when we evaluate the performance of a steganographic technique. We can judge whether the stego-image quality is acceptable to the human eye by using Peak Signal-to-Noise Ratio (PSNR).

PSNR: Imperceptibility takes advantage of human psycho visual redundancy, which is very difficult to quantify. PSNR can also be used as metrics to measure the degree of imperceptibility:

$$PSNR = 10LOG_{10}\frac{255^2}{MSE}dB$$

$$MSE = \left(\frac{1}{MXN}\right)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left(P(x,y) - P'(x,y)\right)^2$$

where M and N are the number of rows and number of columns respectively of the cover image, P(x,y) is the pixel value from the cover image, P'(x,y) is the pixel value from the stego-image. Signal to noise ratio quantifies the imperceptibility, by regarding the message as the signal and the message as the noise.

2. Payload/Hiding Capacity:

The payload indicates the maximum number of bits that can be hidden with an acceptable resultant stego-image quality. Because the scheme would be of no value if the stego-image turned out seriously distorted despite the fact that it can hold a large amount of secret data, the hiding capacity does have its limit, especially when it comes to the binary image. We can say that a scheme does have its contribution to this field of research if it proves to either increase the payload while maintaining an acceptable stego-image quality or improve the stego-image quality while keeping the hiding capacity at the same level, or better if it can get both promoted.

3. Robustness/Resistance to attacks:

It is resistance of stego image to various steganalysis attacks. It is immunity of stego image to all types of manipulations, operations carried on it and successfully transferring the hidden information. Steganalysis is the science of detecting hidden information. The main objective of Steganalysis is to break steganography and the detection of stego image is the goal of steganalysis. Steganalysis algorithms introduce statistical differences between cover and stego image. So Robustness of system is defined as defeating such attacks and successfully transferring information without anyone even knowing its existence.

4.2 Results for Criteria I and II

The outcome results for various evaluation factors for cover image are shown in Table 1.

4.3 Comparison Analysis with Existing Techniques for Evaluation Criteria I and II

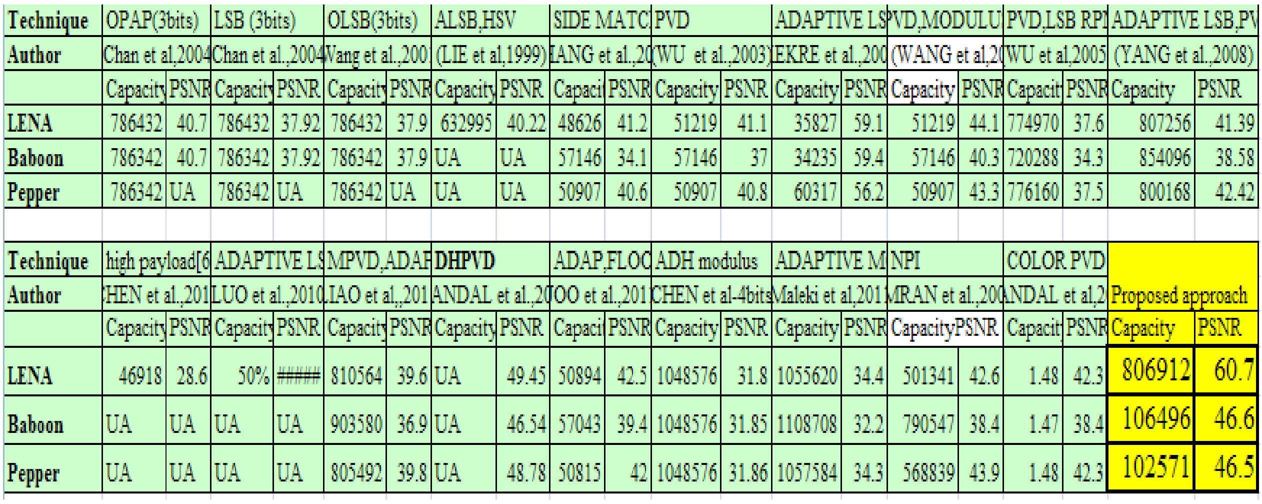
The Comparison of existing techniques and proposed results on the basis of PSNR and Capacity is shown in Table 2.

From Comparsion Analysis it is concluded that the proposed approach is better in Evaluation Criteria 1 and 2 than already exisiting techniques.It provide better imeperceptibilty/quality and hiding capacity than previous known techniques.

Table 1: Embedded data bytes, value of percentage of used pixel of cover image,Percentage of used bytes changed,average no.of bits changed per pixel and to store data,MSE,RMSE,PSNR and Capacity

	Embedded Data Bytes	**% of used Pixel in Image**	**% of Changed Bytes**	**Avg. # Bits/Pixel**	**MSE**	**RMSE**	**Mean**	**Standard Deviation**	**PSNR**	**Capacity**
LEENA	261121	0.4545	42.65	1.47	0.0553	0.2353	256.5	361.3	60.6999	806,912
BABOON	267134	0.4552	39.54	1.13	1.4479	1.2.33	91	129.4	46.5235	106,496
PEPPER	267135	0.4545	41.39	1.13	1.4334	1.1973	92.5	129.4	46.567	102571
FAMILY	270936	0.4545	41.73	1.5	0.015	0.1225	213	299.8	66.37	544,768

Table 2: Capacity and PSNR value Comparison with existing techniques (UA-Unavailable)



Technique	OPAP(3bits)		LSB (3bits)		OLSB(3bits)		ALSB,HSV		SIDE MATC		PVD		ADAPTIVE LS		VD,MODULU		PVD,LSB RPI		ADAPTIVE LSB,PV	
Author	Chan et al,2004		Chan et al.,2004		Wang et al.,200		(LIE et al,1999)		IANG et al.,20		(WU et al.,2003)		EKRE et al.,200		(WANG et al,20		WU et al,2005		(YANG et al.,2008)	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capaci	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacit	PSNR	Capacity	PSNR
LENA	786432	40.7	786432	37.92	786432	37.9	632995	40.22	48626	41.2	51219	41.1	35827	59.1	51219	44.1	774970	37.6	807256	41.39
Baboon	786342	40.7	786342	37.92	786342	37.9	UA	UA	57146	34.1	57146	37	34235	59.4	57146	40.3	720288	34.3	854096	38.58
Pepper	786342	UA	786342	UA	786342	UA	UA	UA	50907	40.6	50907	40.8	60317	56.2	50907	43.3	776160	37.5	800168	42.42

Technique	high payload[6		ADAPTIVE LS		MPVD,ADAF		**DHPVD**		ADAP,FLOC		ADH modulus		ADAPTIVE M		NPI		COLOR PVD			
Author	HEN et al.,201		LUO et al.,2010		IAO et al.,,201		ANDAL et al.,20		OO et al.,201		CHEN et al-4bits		Maleki et al,201		MRAN et al.,200		NDAL et al,2		Proposed approach	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capaci	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacit	PSNR	Capacity	PSNR
LENA	46918	28.6	50%	######	810564	39.6	UA	49.45	50894	42.5	1048576	31.8	1055620	34.4	501341	42.6	1.48	42.3	806912	60.7
Baboon	UA	UA	UA	UA	903580	36.9	UA	46.54	57043	39.4	1048576	31.85	1108708	32.2	790547	38.4	1.47	38.4	106496	46.6
Pepper	UA	UA	UA	UA	805492	39.8	UA	48.78	50815	42	1048576	31.86	1057584	34.3	568839	43.9	1.48	42.3	102571	46.5

4.4 Evaluation Analysis for Criteria III-Robustness/ Resistance to Attacks

4.4.1 Application of Steganlysis Attacks

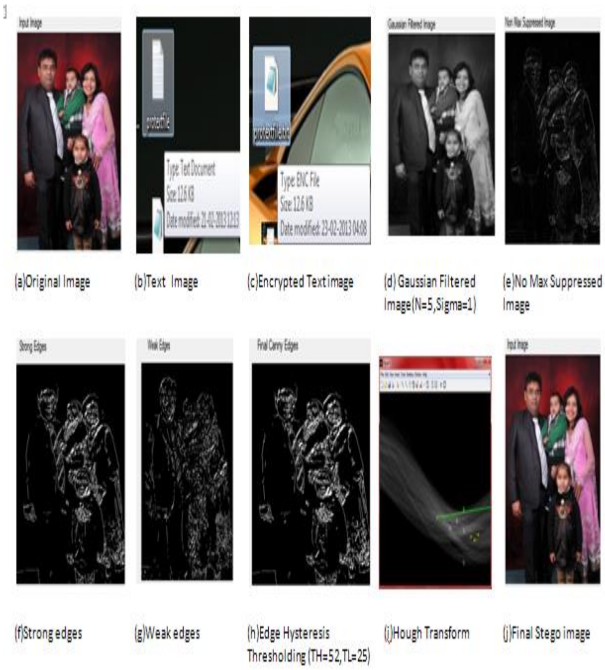
Steganalysis is the science of detecting hidden information. The main objective of Steganalysis is to break steganography and the detection of stego image is the goal of steganalysis. Almost all steganalysis algorithms rely on the Steganographic algorithms introducing statistical differences between cover and stego image. Robustness of proposed technique is thoroughly tested through various steganlysis attacks. These include Visual analysis, Statistical analysis and Universal Analysis as explained in [2, 32, 42].

4.4.1.1 Visual Anlysis

The results of proposed technique on 24 bit Color image (family_pic.bmp) can be seen in Figure 1.This approach successfully resisted visual attacks explained in [57] as could not find any visual difference in input cover image and stego image as demonstrated in Figure 1.

4.4.1.2 Statistical Analysis: Histogram Analysis

The results of Histogram analysis tecnhnique proposed in [45] and are shown in Figure 2 and there are no differenes found in histograms of original and stego image so could not be attacked.



(a)Original Image (b)Text Image (c)Encrypted Text image (d) Gaussian Filtered Image(N=5,Sigma=1) (e)No Max Suppressed Image

(f)Strong edges (g)Weak edges (h)Edge Hysteresis Thresholding(TH=52,TL=25) (i)Hough Transform (j)Final Stego image

Figure 1: (a) to (j): Steps of transformation of Input cover image(family_pic.bmp) to StegoImage

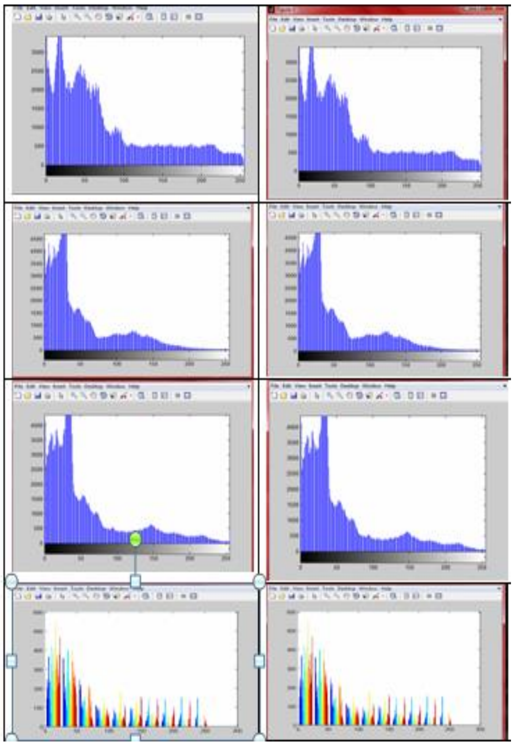


Figure 2: Histogram Analysis of Red, Green, Blue, All components of original image and Stego-image (family_pic.bmp)

4.4.1.3 Statistical Analysis: Chi-Square Attack

The proposed approach was even tested by Chi-Square Attack proposed in [51, 57] and could succesfuly sustain these attacks as shown in Figure 3.

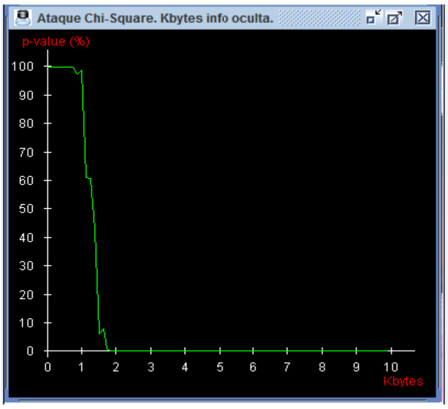
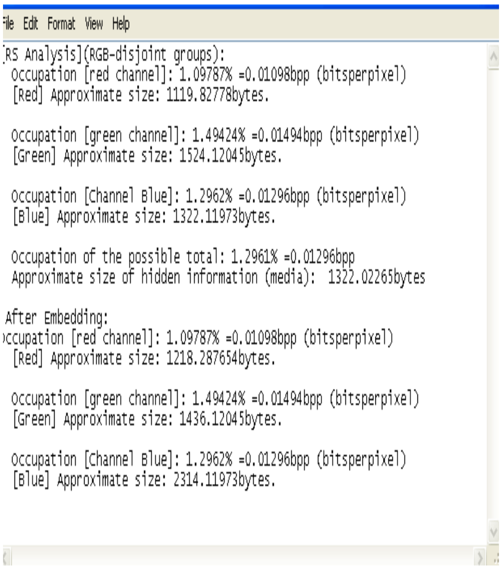


Figure 3 :Chi-Sqauare Attack on Famiy_pic.bmp

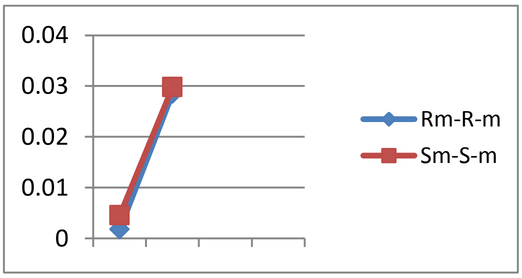
4.4.1.4 Statistical Analysis: RS-Analysis

Result of RS analysis as explained in [14] is shown in Figure 4 and Table 3.The proposed approach could not be attacked by this.



```
[RS Analysis](RGB-disjoint groups):
 Occupation [red channel]: 1.09787% =0.01098bpp (bitsperpixel)
 [Red] Approximate size: 1119.82778bytes.

 Occupation [green channel]: 1.49424% =0.01494bpp (bitsperpixel)
 [Green] Approximate size: 1524.12045bytes.

 Occupation [Channel Blue]: 1.2962% =0.01296bpp (bitsperpixel)
 [Blue] Approximate size: 1322.11973bytes.

 Occupation of the possible total: 1.2961% =0.01296bpp
 Approximate size of hidden information (media):  1322.02265bytes

After Embedding:
Occupation [red channel]: 1.09787% =0.01098bpp (bitsperpixel)
 [Red] Approximate size: 1218.287654bytes.

 Occupation [green channel]: 1.49424% =0.01494bpp (bitsperpixel)
 [Green] Approximate size: 1436.12045bytes.

 Occupation [Channel Blue]: 1.2962% =0.01296bpp (bitsperpixel)
 [Blue] Approximate size: 2314.11973bytes.
```

(a)



(b)

Figure 4: (a, b) RS-Analysis of (family_pic.bmp)

Table 3: RS Analysis for family_pic.bmp

For Family	Initial value	After Embedding
Rm-R-m	0.0017843	0.028498
Sm-S-m	0.0045498	0.029788

5 Conclusions

The above proposed technique achieves the goal of an implementation of new steganography approach for images which integrates three new techniques a) Hybrid feature (line/edge/boundary/circle) detector technique integrating Canny and Enhanced Hough transform for bifurcating an image into edge and smooth areas b) Two Component based LSB Substitution Technique for hiding encrypted messages in edges of images c) Adaptive LSB substitution technique for hiding messages to smooth areas. It achieves the target of 50% hiding capacity and Imperceptibility(PSNR value) with minimum MSE(mean square error)while hiding more data on edges than smooth areas as edges being high in contrast, color, density, frequency and other noise disturbances can tolerate more changes in their pixel values than smooth areas. This technique effectively demonstrates the utilization of maximum number of bits of RGB Color image with total of 12 bits (8bits of Blue component and 4 bits of Green component) out of total of 24 bits. Embedding capacity and PSNR in the range of 50-60% for edge areas, 40-50% for smoother areas and above 50% for whole image is observed for various test images. In addition to above, it provides the better form of feature detection while integrating Canny Edge detector and Enhanced Hough Transform for more refined results. Noise and other disturbances have lesser effects on results in the proposed technique due to hybrid filtering. The two tier security of the hidden data is achieved by utilizing Advanced Encryption standard (AES) and resistance to various attacks by Random Pixel Embedding Technique. The proposed technique has successfully sustained various steganalysis attacks like visual and statistical.

References

[1] R. H. Alwan, F. J. Kadhim, and A. T. Al-Taani, “Data embedding based on better use of bits in image pixels,” *International Journal of Signal Processing*, vol. 2, no.1, pp. 104-107, 2005.

[2] R. Amirtharajan, J. Qin and J. B. B. Rayappan, “Random image steganography and steganalysis: present status and future directions,” *Information Technology Journal*, vol. 11, pp. 566-576, 2012.

[3] Y. Bassil, “Image steganography based on a parameterized canny edge detection algorithm,” *International Journal of Computer Applications*, vol. 60, no.4, Dec.2012.

[4] J. F. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Anal. Machine Intelligence*, vol. 8, no. 6, pp. 679-697, 1986.

[5] C. K. Chan and L. M. Cheng, “Improved hiding data in images by optimal moderately-significant-bit replacement,” *IEE Electronics Letters*, vol. 37, no. 16, pp. 1017-1018, 2001.

[6] C. K. Chan and L. M. Cheng, “Hiding data in images by simple LSB substitution,” *Pattern Recognition*, vol. 37, pp. 469-474, 2004.

[7] C. C. Chang, M. H. Lin, and Y. C. Hu, “A fast and secure image hiding scheme based on lsb substitution,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 4, pp. 399-416, 2002.

[8] C. C. Chang, J. Y. Hsiao, and C. S. Chan, “Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy,” *Pattern Recognition*, vol. 36, pp.1538-1595, 2003.

[9] C. C. Chang and H. W. Tseng, “A steganographic method for digital images using side match,” *Pattern Recognition Letters*, vol. 25, pp.1431-1437, 2004.

[10] C. C. Chang, C. S. Chan, and Y. H. Fan, “Image hiding scheme with modulus function and dynamic programming strategy on partitioned pixels,” *Pattern Recognition*, vol. 39, no. 6, pp. 1155-1167, 2006.

[11] W. Chen, C. Chang, and T. Le, “High Payload steganography mechanism using hybrid edge detector,” *Expert Systems with applications*, vol. 37, pp. 3292-3301, 2010.

[12] Y. C. Chou, C. C. Chang, and K. M. Li, “A large payload data embedding technique for color images,” *Fundamenta Informaticae*, vol. 88, no.1-2, pp. 47-61, 2008.

[13] N. Ferguson and B. Schneier, *Practical Cryptography*, John Wiley, 2003.

[14] J. Fridrich, M. Goljan, and R. Du, “Reliable Detection of LSB steganography in color and grayscale images,” in *Proceedings of ACM workshop on Multimedia and Security: New Challenges*, pp. 27-30, 2001.

[15] A. Gutub, A. Al-Qahtani, and A. Tabakh, “Triple-A: secure RGB image steganography based on randomization,” in *Proceedings of IEEE/ACS International Conference on Computer Systems and Application*, pp.400-403, May 10-13, 2009.

[16] A. Gutub, M. Ankeer, M. Abu-Ghalioun, A. Shaheen, and A. Alvi, “Pixel indicator high capacity technique for RGB image based steganography,” in *Proceedings of 5th IEEE International Workshop on Signal Processing and its Applications*, University of Sharjah, Sharjah, U.A.E, 18-20 Mar.,2008.

[17] K. Hempstalk, “Hiding behind corners: using edges in images for better steganography,” in *Proceedings of the Computing Women's Congress*, Hamilton, New Zealand, 11- 19 Feb. 2006.

[18] P. V. C. Hough, *Method and Means for Recognizing Complex Patterns*, U.S. Patent 3069654, 1962.

[19] M. Hussain and M. Hussain, "Embedding data in edge boundaries with high PSNR," in *Proceedings of 7th International Conference on Emerging Technologies*, pp.1-6, Sep. 2011.

[20] M. Hussain, "Pixel intensity based high capacity data embedding method," in *International Conference on Information and Emerging Technologies*, pp.1 -5, 2010.

[21] A. S. Imran, M. Y. Javed, and N. S. Khattak, *A Robust Method for Encrypted Data Hiding Technique Based on Neighborhood Pixels Information*, World Academy of Science, Engineering and Technology, 2007.

[22] N. F. Johnson, Z. Duric, and S. Jajodia, *Information Hiding, and Watermarking - Attacks & Countermeasures*, Kluwer Academic Publishers, 2000.

[23] J. C. Joo, T. W. Oh, H. Y. Lee, and H. K. Lee, "Adaptive steganographic method using the floor function with practical message formats," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 1, pp. 161-175, Januray 2011.

[24] K. H. Jung, K. J. Ha, and K. Y. Yoo, "Image data hiding method based on multi-pixel differencing and lsb substitution methods," in *Proceedigns of the International Conference on Convergence and Hybrid Information Technology*, pp. 355-358, Daejeon, Korea, Aug. 2008.

[25] M. Kahn, *Steganography Mailing List*, 5 July 1995.

[26] H. B. Kekre, A. Athawale, and P. N. Halarnkar, "Increased capacity of information hiding in lsb's method for text and image" *International Journal of Electrical, Computer, and Systems Engineering*, vol. 2, no. 4, pp. 246-249, 2008

[27] H. B. Kekre, A. Athawale, and P. N. Halarnkar, "Performance evaluation of pixel value differencing and kekre's modified algorithm for information hiding in images," *ACM International Conference on Advances in Computing, Communication and Control*, pp. 342-346, 2009.

[28] A. Ker, "Steganalysis of LSB Matching in Gray scale Images," *IEEE Signal Processing Letter*, vol. 12, no.6, pp. 441-444, June 2005.

[29] A. Ker, "Improved detection of LSB steganography in grayscale images," in *Proceedgins of the 6th International Workshop*, LNCS 3200, pp. 97-115, Springer-Verlag, Toronto, Canada, May 2004.

[30] P. Mohan Kumar and K. L. Shunmuganathan, "Developing a secure image steganographic system using TPVD adaptive LSB matching revisited algorithm for maximizing the embedding rate," *Information Security Journal: A Global Perspective*, vol. 21, no. 2, pp. 65-70, 2012.

[31] Y. K. Lee and L. H. Chen, "High capacity image steganographic model," in *IEEE Proceddings of Visual Image Signal Process*, vol. 147, no. 3, pp. 288-294, 2000.

[32] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, no. 2, pp. 142-172, Apr. 2011.

[33] X. Liao, Q. Y. Wen, and J. Zhang, "A steganographic method for digital images with four-pixel differencing and modified LSB substitution," *Journal of Visual Communication and Image Representation*, vol. 22, no. 1, pp. 1-8, 2011.

[34] W. N. Lie and L. C. Chang, "Data hiding in images with adaptive numbers of least significant bits based on the human visual system," in *Proceedings of IEEE International Conference on Image Processing*, pp. 286-290, Kobe, Japan, Oct. 1999.

[35] J. C. Liu and M. H. Shih, "Generalizations of Pixel value differencing steganography for data hiding in images," *Fundamenta Informaticae*, vol. 83, no. 3, pp. 319-335, 2008.

[36] S. H. Liu, T. H. Chen, H. X. Yao, and W. Gao, "A variable depth LSB data hiding technique in images," in *Proceedings of the 2004 International Conference on Machine Learning and Cybernetics*, vol. 7, pp. 3990-3994, Aug. 2004.

[37] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited," *IEEE Transactions on Infomation Forensics Security*, vol. 5, no. 2, pp.201 -214, 2010.

[38] N. Maleki, M. Jalali, and M. V. Jahan, "An adaptive data hiding method using neighborhood pixels differencing based on modulus function," in *International Conference on Information Processing, Computer Vision, and Pattern Recognition*, Las Vegas, Nevada, USA, July 2011.

[39] J. K. Mandal and D. Das, "Color image steganography based on pixel value differencing in spatial domain," *International Journal of Information Sciences and Techniques*, vol. 2, no.4, pp. 83-93, July 2012.

[40] J. K. Mandal and A. Khamrui, "A data-hiding scheme for digital image using pixel value differencing (DHPVD)," in *International Symposium on Electronic System Design*, pp. 347-351, 2011.

[41] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285-287, 2006.

[42] A. Nissar and A. H. Mir, "Classification of steganalysis techniques: A study," *Digital Signal Processing*, vol. 20, no.6, pp.1758-1770, Dec. 2010.

[43] Y. R. Park, H. H. Kang, S. U. Shin, and K. R. Kwon, "A steganographic scheme in digital images using information of neighboring pixels," in *Proceedings of the International Conference on Natural Computation*, LNCS 3612, pp. 962–968, Springer-Verlag, 2005.

[44] M. T. Parvez and A. Gutub, "RGB intensity based variable-bits image steganography," in *Proceedings of 3rd IEEE Asia-Pacific Services Computing Conference*, Yilan, Taiwan, Dec. 2008.

[45] N. Provos and P. Honeyman, “Detecting steganographic content on the Internet,” in *Proceedings of NDSS'02: Network and Distributed System Security Symposium*, pp. 1-13, 2002.

[46] J. J. Roque and J. M. Minguet, “SLSB: Improving the steganographic algorithm LSB,” in *7th International Workshop on Security in Information Systems*, pp. 57-66, 2009.

[47] N. S. Arjun and A. Negi, “A filtering based approach to adaptive steganography,” in *IEEE Region 10 Conference*, pp.1-4, Nov. 2006.

[48] M. Singh, B. Singh and S. S. Singh, “Hiding encrypted message in the features of images,” *International Journal of Computer Science and Network Security*, vol. 7, no.4, pp. 302-307, Apr. 2007.

[49] N. Singh, B. S. Bhati, and R. S. Raw, “A novel digital image steganalysis approach for investigation,” *International Journal of Computer Applications*, vol. 47, no. 12, pp. 18-21, June 2012.

[50] *Specification for the Advanced Encryption Standard (AES)*, Federal Information Processing Standards Publication 197, 2012.

[51] C. A. Stanley, “Pairs of values and the chi-squared attack,” in *CiteSeer*, pp. 1-45, 2005.

[52] G. Svvalin and S. K. Lenka, “A novel approach to RGB channel based image steganography technique,” *International Arab Journal of e-Technology*, vol. 2, no. 4, pp. 181-186, June 2012.

[53] C. C. Thien and J. C. Lin, “A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function,” *Pattern Recognition*, vol. 36, pp. 2875-2881, 2003.

[54] C. M. Wang, N. I Wu, C. S. Tsai, and M. S. Hwang, “A high quality steganographic method with pixel-value differencing and modulus function.” *Journal of Systems and Software*, vol. 81, no. 1, pp. 150-158, 2008.

[55] R. Z. Wang, C. F. Lin, and J. C. Lin, “Hiding data in images by optimal moderately significant bit replacement,” *IET Electronics Letters*, vol. 36, no. 25, pp. 2069-2070, 2000.

[56] R. Z. Wang, C. F. Lin, and J. C. Lin, “Image hiding by optimal LSB substitution and genetic algorithm,” *Pattern Recognition*, vol. 34, pp. 671-683, 2001.

[57] A. Westfeld and A. Pitzmann, “Attacks on steganographic systems: Breaking the Steganographic utilities ezstego, jsteg, steganos and s-tools-and some lessons learned,” in *Proceedings of the 3rd Information Hiding Workshop*, LNCS 1768, pp. 61-76, Springer-Verlag, 1999.

[58] D. C. Wu and W. H. Tsai, “A steganographic method for images by pixel-value differencing,” *Pattern Recognition Letter*, vol. 24, no. 9-10, pp. 1613-1626, 2003.

[59] H. C. Wu, N. I. Wu, C. S Tsai, and M. S Hwang, “Image steganographic scheme based on pixel-value differencing and LSB replacement methods,” *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, no. 5, pp. 611-615, 2005.

[60] X. Li, B. Yang, D. Cheng, and T. Zeng, “A generalization of LSB matching,” *IEEE Signal Processing Letters*, vol. 16, no. 2, pp. 69-72, 2009.

[61] C. H. Yang and C. Y. Weng, “A Steganographic method for digital images by multi-pixel differencing,” in *Proceedings of the International Computer Symposium*, pp. 831-836, Dec. 2006.

[62] C. H. Yang, C. Y. Weng, S. J. Wang, and H. M. Sun, “Adaptive data hiding in edge areas of images with spatial LSB domain systems,” *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 488-497, 2008.

[63] X. Zhang and S. Wang, “Vulnerability of pixel value differencing steganography to histogram analysis and modification for enhanced security,” *Pattern Recognition Letters*, vol. 25, pp. 331-339, 2004.

Mamta Juneja is an Assistant Professor in University Institute of Engineering and Technology, Panjab University, Chandigarh, India. She did masters in Computer Science from Punjab Technical University, India and currently pursuing Doctorate in the same. Her interest areas include Image Processing, Steganography, Information Hiding and Information Security.

Parvinder S. Sandhu is Doctorate in Computer Science and Engineering and working as Professor in Computer Science & Engineering department at Rayat & Bahra Institute of Engineering and Bio-Technology, Mohali, Punjab, INDIA. He is editorial committee member of various International Journals and conferences. He has published more than 150 research papers in various referred International journals and conferences. He chaired more than 100 renowned International Conferences and also acted as keynote speaker in different countries. His current research interests are Software Reusability, Software Maintenance, Machine Learning and Image Processing.