Object Oriented Steganography using Skin Tone Detection and RSA Encryption Scheme

Aruna Mittal[1]

[1]Department of Computer Science & Engineering, Disha Institute of Management & Technology,
Chhattisgarh Swami Vivekanand technical University Bhilai, India
mittalaruna28@gmail.com

Abstract: ***Steganography is the science that involves communicating secret data in an appropriate multimedia carrier, e.g., image, audio and video files. While cryptography scrambles the secret message to render it unintelligible, on the other hand Steganography hides the very existence of secret data. Thus Steganography does not replace cryptography but rather boosts the security using its obscurity features. Its main objectives are undetectability, robustness, resistance to various image processing methods, compression, and payload capacity. These unique features separate it from other related techniques such as watermarking and cryptography. The proposed work investigates current state-of-the-art methods and provides a new and efficient approach to digital image steganography. It also establishes a robust steganographic system called Steganoflage. Steganoflage advocates an object-oriented approach in which skin-tone detected areas in the image are selected for embedding where possible. The key objectives of the work are: 1) RSA and OAEP Encryption of the secret text message, 2) HSV model based skin-tone detection algorithm using wavelet domain. Each of these components is tested against relevant performance measurements. The results are promising and provide satisfactory PSNR against various image manipulations and noises such as Poisson, Gaussian, Speckle, Salt and Pepper, and transformations such as rotation, scaling etc.***

Keywords: Cropping, Gaussian, HSV, OAEP, Poisson, PSNR, RSA, Skin tone detection, Stego Image

1. Introduction

With advancements in digital communication technology and the growth of computer power and storage, the difficulties in ensuring individuals' privacy becomes increasingly challenging. The degrees to which individuals appreciate privacy differ from one person to another. Various methods have been investigated and developed to protect personal privacy. Encryption is probably the most obvious one, and then comes steganography. Encryption lends itself to noise and is generally observed while steganography is not observable.

All of the existing methods of steganography focus on the embedding strategy and give no consideration to the pre-processing stages, such as encryption, as they depend heavily on the conventional encryption algorithms which obviously are not tailored to steganography applications where flexibility, robustness and security are required. The proposed work tries to integrate both of them (Object Oriented Steganography and Cryptography) to enhance security of messages being sent via various digital media.

Rest of the paper is organized as follows. Section II presents literature survey and theoretical background. In section III proposed method is described in detail with skin tone detection, DWT, embedding and extraction procedure step by step. Section IV demonstrates the experimental results. Finally conclusions are provided in section V.

2. Literature Review

2.1 Steganography exploiting the image format

Steganography can be accomplished by simply appending the secret message found in the text file 'Message.txt' into the JPEG image file 'Cover.jpg' and produces the stego-image 'Stego.jpg'. The idea behind this is to abuse the recognition of EOF (End of file). In other words, the message is packed and inserted after the EOF tag. When Stego.jpg is viewed using any photo editing application, the latter will just display the picture ignoring anything coming after the EOF tag. However, when opened in Notepad for example, our message reveals itself after displaying some data. The embedded message does not impair the image quality. Neither image histograms nor visual perception can detect any difference between the two images due to the secret message being hidden after the EOF tag. Whilst this method is simple, but this technique cannot resist any kind of editing to neither the Stego-image nor any attacks by Steganalysis experts.

Another naïve implementation of steganography is to append hidden data into the image's Extended File Information (EXIF), which is a standard used by digital camera manufacturers to store information in the image file, such as, the make and model of a camera, the time the picture was taken and digitized, the resolution of the image, exposure time, and the focal length. This is metadata information about the image and its source located at the header of the file. T here is possibility of using such headers in digital evidence analysis to combat child pornography [1]

2.2 Steganography in the image spatial domain

Image domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation and are sometimes characterized as "simple systems". The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format. Potdar et al. [2] used a spatial domain technique in producing a fingerprinted secret sharing

steganography for robustness against image cropping attacks. Their paper addressed the issue of image cropping effects rather than proposing an embedding technique. The logic behind their proposed work is to divide the cover image into sub-images and compress and encrypt the secret data. The resulting data is then sub-divided in turn and embedded into those image portions. To recover the data, a Lagrange Interpolating Polynomial was applied along with an encryption algorithm. The computational load was high, but their algorithm parameters, namely the number of sub-images (n) and the threshold value (k) were not set to optimal values leaving the reader to guess the values. Bear in mind also that if n is set to 32, for example, that means 32 public keys are needed along with 32 persons and 32 sub-images, which turns out to be unpractical. Moreover, data redundancy that they intended to eliminate does occur in their stego-image.

Color palette based steganography exploits the smooth ramp transition in colors as indicated in the color palette. The LSBs here are modified based on their positions in the palette index. Johnson and Jajodia [3] were in favor of using BMP (24-bit) instead of JPEG images. Their next-best choice was GIF files (256-color). BMP as well as GIF based steganography apply LSB techniques, while their resistance to statistical counter attacks and compression are reported to be weak [3, 4, 5, 6, 7]. BMP files are bigger compared to other formats which render them improper for network transmissions. JPEG images however, were at the beginning avoided because of their compression algorithm which does not support a direct LSB embedding into the spatial domain. In [8], the authors claimed that changes as small as flipping the LSB of one pixel in a JPEG image can be reliably detected. The experiments on the Discrete Cosine Transform (DCT) coefficients showed promising results and redirected researchers' attention towards this type of image. In fact acting at the level of DCT makes steganography more robust and less prone to statistical attacks.

2.3 Steganography in the image frequency domain

Steganography in the transform domain involves the manipulation of algorithms and image transforms. These methods hide messages in more significant areas of the cover image, making it more robust. Many transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression. Most of the techniques here use JPEG images as vehicles to embed their data. JPEG compression uses the DCT to transform successive sub-image blocks (8x8 pixels) into 64 DCT coefficients. Data is inserted into these coefficients' insignificant bits; however, altering any single coefficient would affect the entire 64 block pixels [9]. As the change is operating on the frequency domain instead of the spatial domain there will be no visible change in the cover image given those coefficients are handled with care [10].

Abdelwahab and Hassan [11] propose a data hiding technique in the DWT domain. Both secret and cover images are decomposed using DWT (1st level). Each of which is divided into disjoint 4x4 blocks. Blocks of the secret image fit into the cover blocks to determine the best match. Afterwards, error blocks are generated and embedded into coefficients of the best matched blocks in the HL of the cover image. Two keys must be communicated; one holds the indices to the matched blocks in the CLL (cover approximation) and another for the matched blocks in the CHL of the cover. Note that the extracted payload is not totally identical to the embedded version as the only embedded and extracted bits belong to the secret image approximation while setting all the data in other sub images to zeros during the reconstruction process.

3. Proposed Method

Proposed method introduces a new method of embedding secret data within skin and as well as in the edge area, as it is not that much sensitive to HVS (Human Visual System).This method takes advantage of Biometrics features such as skin tone edge detection, instead of embedding data anywhere in Image, data will be embedded in selected regions like skin region. Overview of method is briefly introduced as follows. At first skin tone detection is performed on input image using HSV (Hue, Saturation, Value) color model. Secondly cover image is transformed in Frequency domain. This is performed by applying DWT. Then payload (number of bits in which we can hide data) is calculated. Finally secret data embedding is performed in one of the high frequency sub-band by tracing skin pixels in that band. Before performing all steps cropping on input image is performed and then in only cropped region embedding is done, not in whole image. Cropping results into enhanced security, since cropped region works as a key at the decoding side. Here embedding process affects only certain Regions of Interest (ROI) rather than the entire image. So utilizing objects within images can be more advantageous. This is also called as Object Oriented steganography. Then a stego DWT image is produced, so the I-DWT is performed on that. Thereafter I-DWT image is merged with original image, and we get the final stego image.

3.1 Skin color tone detection

Hue-saturation based color spaces were introduced when there was a need for the user to specify color properties numerically. They describe color with intuitive values, based on the artist's idea of tint, saturation and tone. Hue defines the dominant color (such as red, green, purple and yellow) of an area; saturation measures the colorfulness of an area in proportion to its brightness. The "intensity", "lightness" or "value" is related to the color luminance. The intuitiveness of the color space components and explicit discrimination between luminance and chrominance properties made these color spaces popular in the works on skin color segmentation. Image processing applications such as histogram operations, intensity transformations and convolutions operate only on an intensity image. These operations are performed with much ease on an image in the HSV color space. For the HSV being modeled with cylindrical coordinates, the hue (H) is represented as the angle 0, varying from 0o to 360o. Saturation (S) corresponds to the radius, varying from 0 to 1. Value (V) varies along the z axis with 0 being black and 1 being white. When S = 0, color is a gray value of value 1. When S = 1, color is on the boundary of top cone base. The greater the saturation, the farther the color is from

white/gray/black (depending on the intensity). Adjusting the hue will vary the color from red at 0o , through green at 120o, blue at 240o, and back to red at 360o. When V = 0, the color is black and therefore H is undefined. When S = 0, the color is grayscale. H is also undefined in this case. By adjusting V, a color can be made darker or lighter. By maintaining S = 1 and adjusting V, shades of that color are created

These RGB values can be converted into HSV by using eq 1

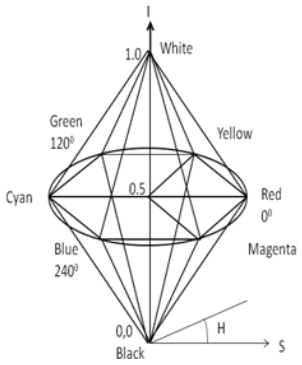

Figure 1 Double Cone Model of HSV Color Space

$$H = \begin{cases} h, B \le G \\ 2\pi - h, B > G \end{cases}$$

$$where, h = \cos^{-1}\frac{\frac{1}{2}(R-G)+(R-B)}{\sqrt{(R-G)^2+(R-G)(G-B)}}$$

$$S = \frac{max(R,G,B) - min(R,G,B)}{max(R,G,B)}$$

$$V = \max(R,G,B) \qquad \ldots(1)$$

3.2 Skin Modeling

The final goal of skin color detection is to build a decision rule that will discriminate between skin and non-skin pixels. This is usually accomplished by introducing a metric, which measures distance (in general sense) of the pixel color to skin tone. The type of this metric is defined by the skin color modeling method. One method to build a skin classifier is to define explicitly (through a number of rules) the boundaries skin cluster in some color space. For example:-

(R, G, B) is classified as skin if:

$$R > 95 \text{ and } G > 40 \text{ and } B > 20$$
$$\max(R,G,B) - \min(R,G,B) > 15$$
$$|R| - |G| > 15 \text{ and } R > G \text{ and } R > B \qquad \ldots\ldots (2)$$

The simplicity of this method has attracted (and still does) many researchers. The obvious advantage of this method is simplicity of skin detection rules that leads to construction of a very rapid classifier.

3.3 Masking and filtering

Masking and filtering techniques are usually restricted to 24 bits or grayscale images for hiding a message. This is achieved for example by modifying the luminance of parts of the image. While masking changes the visible properties of an image, it can be done in such a way that the human eye will not notice the anomalies. Generally masking uses visible aspects of the image; also it is more robust than LSB modification with respect to compression, cropping and different kinds of image processing. Although the information is not hidden at the ”noise” level , rather than it is inside the visible part of the image, which makes it more suitable than LSB modifications in case a lossy compression algorithm like JPEG is being used. In skin detection algorithms masking basically means covering the non skin region with a black mask. Filtering is replacing the white region (representing the skin portion in the binary image) with the original skin portion in the cover image. The masking and filtering operation is shown in Figure 2:

Figure 2 Masking and Filtering Operation

3.4 Discrete Wavelet Transform (DWT)

This is another frequency domain in which steganography can be implemented. DCT is calculated on blocks of independent pixels, a coding error causes discontinuity between blocks resulting in annoying blocking artifacts. This drawback of DCT is eliminated using DWT.DWT applies on entire image. DWT offers better energy 40 compaction than DCT without any blocking artifact. DWT splits component into numerous frequency bands called sub bands known as

LL – Horizontally and vertically low pass
LH – Horizontally low pass and vertically high pass
HL - Horizontally high pass and vertically low pass
HH - Horizontally and vertically high pass

Since Human eyes are much more sensitive to the low frequency part (LL subband) we can hide secret message in other three parts without making any alteration in LL subband [12]. As other three sub-bands are high frequency sub-band they contain insignificant data. Hiding secret data in these sub-bands doesn't degrade image quality that much.

3.5 Implementation of DWT in 1D

In separable DWT the analysis filter bank decomposes the input signal x(n) into two sub band signals, c(n) and d(n). The signal c(n) represents the low frequency part of x(n), while the signal d(n) represents the high frequency part of x(n). We denote the low pass filter by af1 (analysis filter 1) and the high pass filter by af2 (synthesis filter 2). As depicted in figure(3), the output of each filter is then down sampled by 2 to obtain the two sub band signals c(n) & d(n)[13].

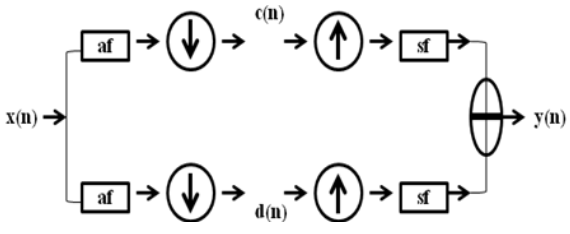

Figure 3 Analysis and Synthesis filter banks applied to 1D Signal

The Synthesis filter bank combines the two sub band signals c(n) & d(n) to obtain a single signal y(n). The synthesis filter bank up-samples each of the two sub band signals. The signals are then filtered using a low pass and high pass filter. We denote the low pass filter by

sf1(synthesis filter 1) and the high pass filter by sf2(synthesis filter 2). The signals are then added together to obtain the signal y(n). If the four filters are designed so as to guarantee that the output signal y(n) equals the input signal x(n), then the filters are said to satisfy the perfect reconstruction condition.

3.6 2-D discrete wavelet transform

Image-processing applications require two-dimensional implementation of wavelet transform. Implementation of 2D DWT [14],[15],[16] is also referred to as multidimensional wavelet transform in literature. In the 2D case, the 1D analysis filter bank is first applied to the columns of the image and then applied to the rows. If the image has N1 rows and N2 columns, then after applying the 1D analysis filter bank to each column we have two sub band images, each having N1/2 rows and N2 columns; after applying the 1D analysis filter bank to each row of both of the two sub band images, four sub band images are obtained, each having N1/2 rows & N2/2 columns. This is depicted in figure (4) given below. The 2D synthesis filter bank combines the four sub band images to obtain the original image of size N1 by N2 [15][16].

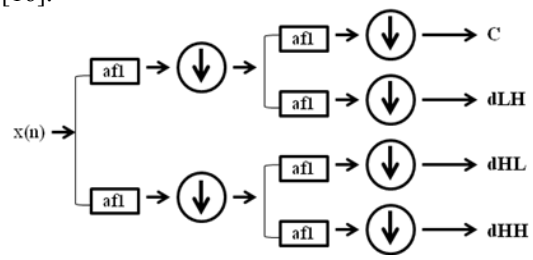

Figure 4 Analysis and Synthesis filter banks applied to 2D Signal

3.7 RSA and OAEP encryption

3.7.1RSA Encryption

The RSA cryptosystem, named after its inventors R. Rivest, A. Shamir, and L. Adleman, is the most widely used public key Cryptosystem. It may be used to provide both secrecy and digital signatures and its security is based on the intractability of the integer factorization.

The scheme makes use of an expression with exponentials. Plaintext is encrypted in blocks, with each block having a binary value less than some number n. That is, the block size must be less than or equal to log2(n); in practice, the block size is i bits, where 2i < n <2i+1. Encryption and decryption are of the following form, for some plaintext block M and cipher text block C:

$$C = M^e \bmod n \quad \text{.. (3)}$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n \quad \text{...(4)}$$

Both sender and receiver must know the value of n. The sender knows the value of e, and only the receiver knows the value of d. Thus, this is a public-key encryption algorithm with a public key of PU = {e, n} and a private key of PU = {d, n}. For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

It is possible to find values of e, d, n such that Med mod n = M for all M < n.

It is relatively easy to calculate mod M^e mod n and C^d for all values of M < n.

It is infeasible to determine d given e and n.

The algorithm is described as

Key Generation:

Select two random numbers p and q such that both are prime and p≠q.
Calculate n=p×q
Calculate φ (n) = (p-1)(q-1)
Select public key e such that gcd (φ(n),e)=1 and 1<e< φ(n).
Calculate private key d such that d=e-1(mod (φ(n)))
Public key is given by PU= {e, n}
Private Key is given by PR= {d, n}

Encryption:

Plaintext M should be such that M<n.
Cipher text $C=M^e$ mod n.

Decryption:

$M = C^d$ mod n.

3.7.2 Security of RSA

Four possible approaches to attacking the RSA algorithm are as follows:
Brute force: This involves trying all possible private keys.
Mathematical attacks: There are several approaches, all equivalent in effort to factoring the product of two primes.
Timing attacks: These depend on the running time of the decryption algorithm.
Chosen cipher text attacks: This type of attack exploits properties of the RSA algorithm.

The defense against the brute-force approach is the same for RSA as for other cryptosystems, namely, use a large key space. Thus, the larger the number of bits in d, the more robust the system is against attacks. However, because the calculations involved, both in key generation and in encryption/decryption, are complex, the larger the size of the key, the slower the system will run. We can identify three approaches to attacking RSA mathematically:

- Factor n into its two prime factors. This enables calculation of f (n) = (p - 1) x (q - 1), which, in turn, enables determination of d e-1 (mod f(n)).
- Determine f (n) directly, without first determining p and q. Again, this enables determination of d e-1 (mod f (n)).
- Determine d directly, without first determining f (n).

3.7.3 OAEP

To overcome the drawbacks of RSA, a randomization approach is combined to it namely OAEP. Optimal Asymmetric Encryption Padding (OAEP) is a padding scheme in the form of a Feistel network which uses a pair of random oracles G and H to process the plaintext prior to asymmetric encryption. When combined with any secure trapdoor one-way permutation f, this processing is proved in the random oracle model to result in a combined scheme which is semantically secure under chosen plaintext attack (IND-CPA). When implemented with certain trapdoor permutations (e.g., RSA), OAEP is also proved secure against chosen cipher text attack. OAEP can be used to build an all-or-nothing transform. OAEP satisfies the following

two goals:
Add an element of randomness which can be used to convert a deterministic encryption scheme (e.g., traditional RSA) into a probabilistic scheme.
Prevent partial decryption of cipher texts (or other information leakage) by ensuring that an adversary cannot recover any portion of the plaintext without being able to invert the trapdoor one-way permutation f.

3.7.4 Implementation of OAEP

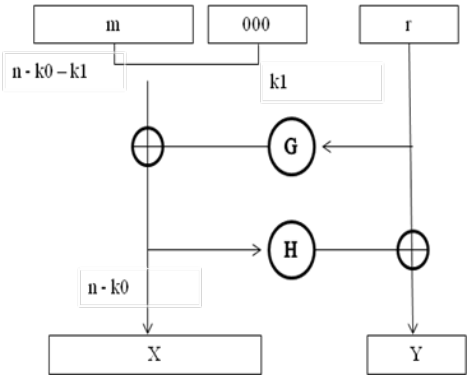

Figure 5 OAEP Diagram

n is the number of bits in the RSA modulus.
k_0 and k_1 are integers fixed by the protocol.
m is the plaintext message, an $(n - k_0 - k_1)$-bit string
G and H are typically some cryptographic hash functions fixed by the protocol.

To encode,
messages are padded with k_1 zeros to be $n - k_0$ bits in length.
r is a random k_0-bit string
G expands the k_0 bits of r to $n - k_0$ bits.
$X = m00..0 \oplus G(r)$
H reduces the $n - k_0$ bits of X to k_0 bits.
$Y = r \oplus H(X)$
The output is X || Y where X is shown in the diagram as the leftmost block and Y as the rightmost block.

To decode,
Recover the random string as $r = Y \oplus H(X)$
Recover the message as $m00..0 = X \oplus G(r)$

The "all-or-nothing" security is from the fact that to recover m, you must recover the entire X and the entire Y; X is required to recover r from Y, and r is required to recover m from X. Since any changed bit of a cryptographic hash completely changes the result, the entire X, and the entire Y must both be completely recovered.

3.8 Encoding and data hiding process

Suppose C is original 24-bit color cover image of PXQ Size.

$$C = \{x_{ij}, y_{ij}, z_{ij} | 1 \le i \le P, 1 \le j \le Q, x_{ij}, y_{ij}, z_{ij} \in \{0,1, \ldots .255\}\} \quad ..(5)$$

Let size of cropped image is Pc×Qc where Pc≤P and Qc≤Q and Pc=Qc. i.e. Cropped region must be exact square as we have to apply DWT later on this region. Let S is secret data. Here secret data considered is binary image of size a×b. Figure 6 represents flowchart of embedding process. Different steps of flowchart are given in detail below.

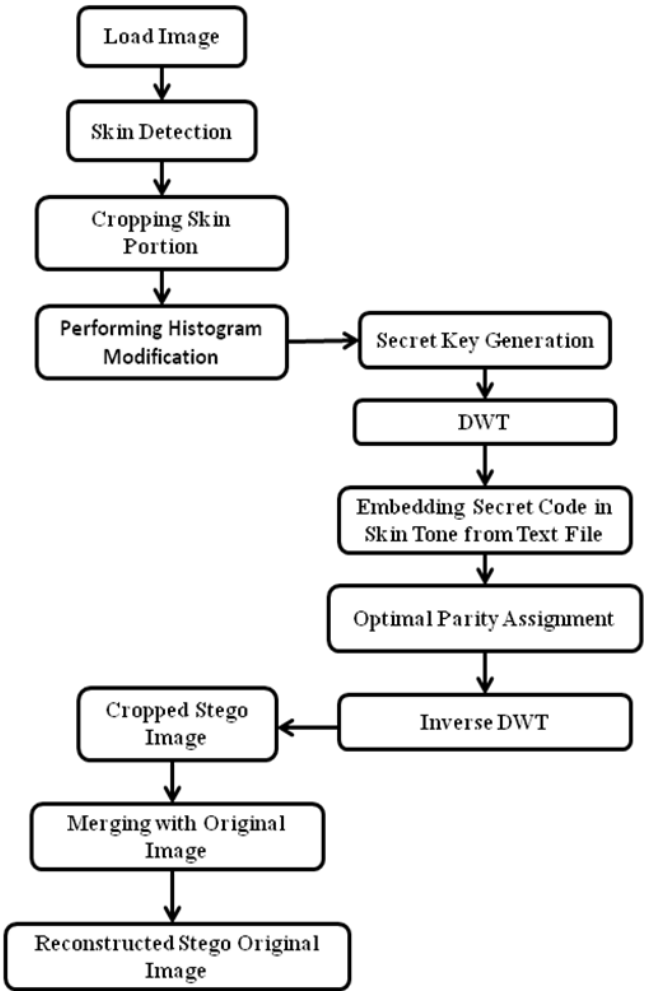

Figure 6 Flowchart of Encoding Process

Steps:

1. Initially load the cover object in which we will hide the secret message (text).
2. After loading the cover object, skin tone detection is performed. This enables us to know where and how much data can be hidden.
3. Cropping: From the detected skin portion, cropping is performed. This is done so that within skin pixels data is hidden at only limited pixel positions. This feature of cropping enhances security, as any eavesdropper cannot detect secret message just by detecting the skin pixels.
4. Histogram Modification: This is performed to adjust the contrast of the colours.
5. Key Generation: This is the step where the secret message to be selected and is encrypted using RSA and OAEP.
6. DWT: Discrete Wavelet Transform is applied to the cropped skin portion.
7. Secret encrypted message is now merged into the transformed skin pixels.
8. Optimal Parity Assignment is used to assign secret code values to limited areas of cropped skin portion, so as to have least effect over the HVS (human visual system).
9. Inverse DWT: Now the transformed image has secret code as well, so it is ready to be merged with the original cover object. The first step to merge this transformed secret message embedded image, with cover object is to inverse transform it.

10. After applying inverse DWT, we get the original cropped image along with secret code. This image is now called stego image. This stego image is now merged with original cover image to get the final reconstructed cover image along with secret data embedded in it. This Stego image is now sent to the receiver by some transmission medium.

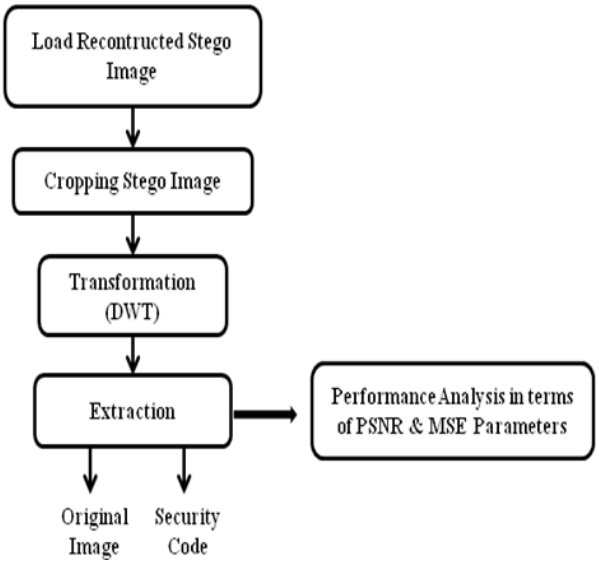

Figure 7 Flowchart of Decoding Process

At the Decoding End following steps are performed:

1. From the Stego Image skin pixels are detected and cropping of Stego image is performed.
2. Now the DWT is performed to get the transformed cropped image.
3. Secret encrypted message is extracted from the transformed cropped stego image. This encrypted message is decrypted (using RSA+ OAEP decryption) to get the secret message.
4. Results of Extraction process are measured in terms of PSNR and MSE. This are discussed below in detail.

4. Results

If In this section we demonstrate simulation results for the proposed scheme. These have been implemented using MATLAB 7.6.0. A 24 bit color image is employed as cover-image of size 256×256, shown in Fig. 8, Fig.9 shows sample secret message image to hide inside cover image.

Figure 8 Cover Image

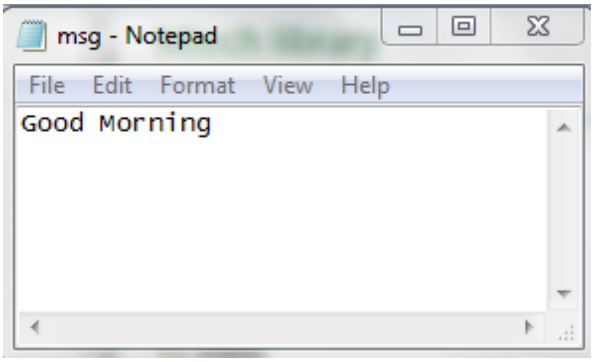

Figure 9 Secret Message Image

Performance measurement for image distortion is well known as peak signal to noise ratio (PSNR) which is classified under the difference distortion metrics and can be applied on stego images.

PSNR is used to evaluate quality of stego image after embedding the secret message. Secret message can be any word. The performance in terms of capacity and PSNR (in dB) is demonstrated for the method in the following subsections. PSNR is defined as per Eq.6

$$PSNR = 10 \log_{10}\left(\frac{255^2}{MSE}\right) \quad \text{-------(6)}$$

$$MSE = \left(\frac{1}{M \times N}\right) \sum_{i=1}^{M} \sum_{j=1}^{N} (a_{ij} - b_{ij})^2 \quad \text{-------(7)}$$

a_{ij} and b_{ij} represents pixel values of original cover image and stego image respectively as in Eq.7 The calculated PSNR as in Eq.6 usually adopts dB value for quality judgement, the larger PSNR is higher the image quality (which means there is a little difference between cover image and stego image).On the contrary smaller dB value means there is a more distortion. PSNR values falling below 30dB indicate fairly a low quality.

However, high quality strives for 40dB or more.

Result Discussion of proposed work

After embedding secret data in cropped image, resulted cropped stego image is shown in Fig. 10. Cover image is now merged with cropped embedded Stego image as is shown in Fig.11. For merging, co-ordinates of first and last pixels of cropped image are calculated and then replaced with the one in original cover image. After performing decoding process on stego image, retrieved output text file consisting of the secret message is shown in Fig 12.

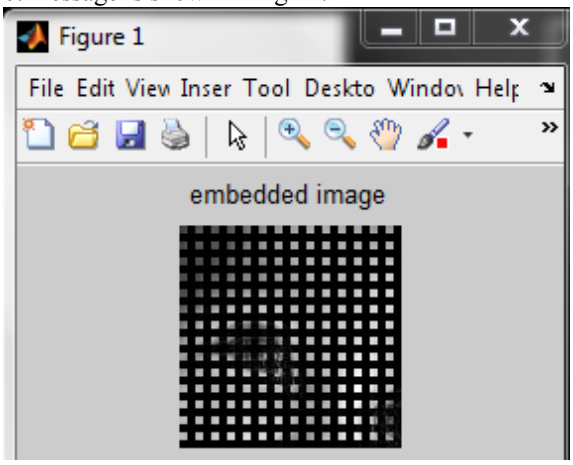

Figure 10 Cropped Stego Image

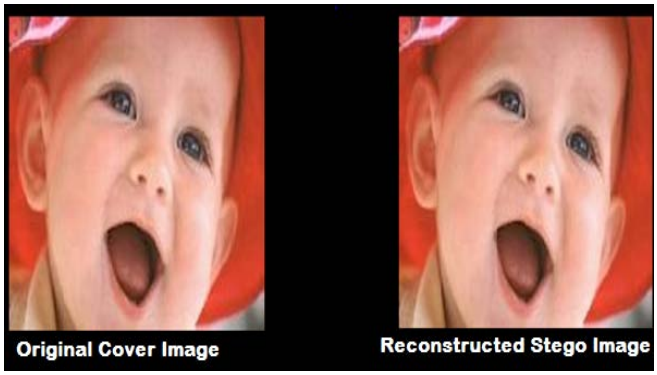

Figure 11 Original and Reconstructed Stego Image

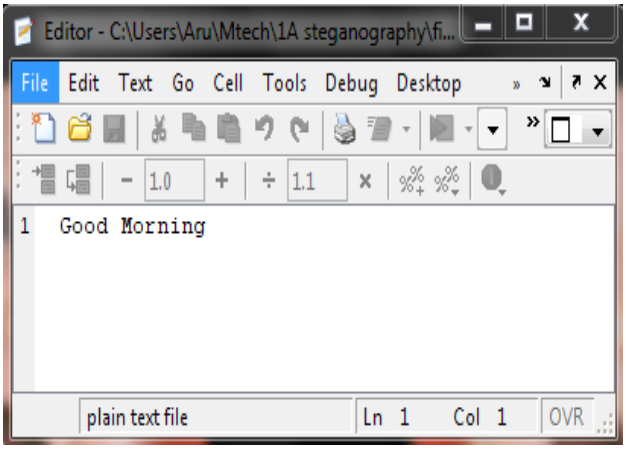

Figure 12 Output Text File (having the secret message)

PSNR is calculated for two different final stego images resulted from a considered image and one more sample image. This PSNR for different cases is shown in table 1. Average PSNR of proposed method is calculated based on the obtained PSNR. Average PSNR obtained by the proposed method is better than the ones proposed by Rekha Nagar and Anjali Shejul (as can be seen in table 2). Table 1 also includes PSNR of considered image after addition of noises (like Gaussian, Salt and Pepper, Speckle, Poisson, and Image Rotation), which are fairly acceptable (having PSNR greater than 40). Thus the proposed method is better than previous ones as well as robust against various noises.

Table 1. Proposed Methods Results of PSNR for Different Images

S. No.	*Cover Image*	*With or Without Addition of Noise*	*PSNR*
1	Image 1	Without Noise	71.4286
2	Image 2	Without Noise	51.9769
3	Image 1	Gaussian Noise	48.6234
4	Image 1	Salt and Pepper Noise	56.3243
5	Image 1	Speckle Noise	49.3423
6	Image 1	Poisson Noise	58.6758
7	Image 1	Image Rotation	42.6754

Table 2. Comparison with The Previous Systems

Title	*Author*	*PSNR*
A DWT based approach for Steganography using Biometrics	Anjali Shejul	48.70
Object oriented steganography based on Biometric and spread spectrum	Meena, Danvir	24.92
An image hiding algorithm using discrete wavelet transform and skin tone detection	Rekha Nagar	51.00
Performance comparison of robust Steganography based on multiple transformation techniques	Shiva Kumar	41.75
Efficient and secure Biometric Image steganography using discrete wavelet transform	Sunita Barve	27.33
Object oriented steganography using RSA and OAEP Encryption	Aruna Mittal	51.97

5. Conclusion

Digital Steganography is a fascinating scientific area which falls under the umbrella of security systems. Proposed framework is based on steganography that uses Biometric feature i.e. skin tone region. Skin tone detection plays a very important role in Biometrics and can be considered as a secure location for data hiding. Secret data is encrypted using RSA and OAEP, thus making the message more secure and tolerant to attacks.

References

[1] P. Alvarez, "Using Extended File Information (EXIF) File Headers in Digital Evidence Analysis", International Journal of Digital Evidence, Economic Crime Institute (ECI), 2(3)(2004)1-5

[2] V.M. Potdar, S. Han and E. Chang, "Fingerprinted Secret Sharing Steganography for Robustness Against Image Cropping Attacks", in: Proceedings of IEEE 3rd International Conference on Industrial Informatics (INDIN), Perth, Australia, 10-12 August 2005, pp. 717-724.

[3] N.F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen", IEEE Computer, 31(2)(1998) 26-34.

[4] M. Jiansheng, L. Sukang, and T. Xiaomei, "A Digital Watermarking Algorithm Based On DCT and DWT," International Symposium on Web Information Systems and Applications (WISA'09), 2009.

[5] I. El-Fegh, D. Mustafa, Z. S. Zubi et al., "Color Image Watermarking Based On The DCT-Domain Of Three RGB Color Channels," in Proceedings of the 10th WSEAS international conference on evolutionary computing, Prague, Czech Republic, 2009, pp. 36-39.

[6] M. Ouhsain, and A. B. Hamza, "Image Watermarking Scheme Using Nonnegative Matrix Factorization And Wavelet Transform," Expert Syst. Appl., vol. 36, no. 2, pp. 2123-2129, 2009.
[7] C.-Y. Chang, H.-J. Wang, and S.-W. Pan, "A Robust DWT-Based Copyright Verification Scheme With Fuzzy ART," Journal of Systems and Software, vol. 82, no. 11, pp. 1906-1915, 2009.
[8] J. Fridrich, M. Goljan and D. Hogeg, "Steganalysis of JPEG Images: Breaking the F5 Algorithm", in: Proceedings of Information Hiding: 5th International Workshop, IH 2002 Noordwijkerhout, The Netherlands, LNCS, Springer, October 7-9, 2002, 2578/2003, pp. 310-323.
[9] A.M. Fard, M. Akbarzadeh-T and F. Varasteh-A, "A New Genetic Algorithm Approach for Secure JPEG Steganography", in: Proceedings of IEEE International Conference on Engineering of Intelligent Systems, 22-23 April 2006, pp. 1-6.
[10] A.I. Hashad, A.S. Madani and A.E.M.A. Wahdan, "A Robust Steganography Technique Using Discrete Cosine Transform Insertion", in: Proceedings of IEEE/ITI 3rd International Conference on Information and Communications Technology, Enabling Technologies for the New Knowledge Society, 5-6 Dec. 2005, pp.255-264.
[11] A.A. Abdelwahab and L.A. Hassan, "A Discrete Wavelet Transform Based Technique For Image Data Hiding", in: Proceedings of 25th National Radio Science Conference, NRSC 2008, Egypt, March 18-20 2008, pp.1-9.
[12] A. Cheddad, J. Condell, K. Curran and P. Mc Kevitt, "Biometric Inspired Digital Image Steganography", in: Proceedings of the 15th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'08), Belfast, pp. 159-168, 2008,
[13] Chauhan, R. P. S., "A Novel Approach to Overcome the Intertwined Shortcomings of DWT for Image Processing and De-noising", International Journal of Engineering Research and Applications (IJERA), pp. 464-470, Jan-Feb 2012,.
[14] R. Gomathi & S. Sevakumaran, "A Bivariate Shrinkage Function for Complex Dual-Tree DWT based Image De-noising", in Proc. ICWAMS-2006, Bucharest, Romania, October 16-18, 2006
[15] I. W. Selesnick, "The Double Density DWT in Wavelets in Signal and Image Analysis: From Theory to Practice", A. Petrosian and F.G. Meyer, Eds. Boston, MA: Kluwer, 2001
[16] I. W. Selesnick, "The Double Density Dual-Tree DWT," IEEE Trans. On Signal Processing, 52(5): 1304-1314, May 2004.

Author Profile

Aruna Mittal received her degree of B.E. in Computer Science and Engineering from Rajeev Gandhi Technical University, Bhopal, in the year 2007. She is pursuing her M.Tech Degree in Information Security from Chhattisgarh Swami Vivekanand Technical University, Bhilai. She has more than 3 years of industry experience. Her interest areas are data warehousing and information security.