Image Steganography with Cryptography using Multiple Key Patterns

Aruna Varanasi
Professor
Sreenidhi Institute of Science and Technology, Hyderabad

M. Lakshmi Anjana
Student
Sreenidhi Institute of Science and Technology, Hyderabad

Pravallika Pasupulate
Student
Sreenidhi Institute of Science and Technology, Hyderabad

ABSTRACT

Image Steganography is the art of hiding information into an innocent image. In this paper a new Steganography method is presented. First, the message is converted into ciphertext using a strong encryption algorithm with the help of secret key. In this proposed Steganography system, sender and receiver share multiple stego keys. Among the multiple stego keys, any one of the stego key is used for steganography. The information about the stego key (key number) is embedded in the first pixels of the original image. Based on the pattern given in the stego key, ciphertext is embedded in the pixels of original image. The receiver first extracts the key number and then obtains the stego key from the database. Using the pattern given in the stego key, ciphertext is retrieved from the pixels of Stego image. The ciphertext is decrypted using the secret key. Thus, it is difficult to recognize that some information is embedded in the cover image, called Stego image, which is same as original image. The proposed system is simple and robust against the attacks.

General Terms

Security, Cryptography, Steganography.

Keywords

Steganography, Encryption, Decryption, Plaintext, Ciphertext, Stego image, Cover image, Secret key, Stego key.

1. INTRODUCTION

Steganography [1] is the art and science of hiding information. It is a Greek origin word which means “hidden writing”. In the word Steganography “Steganos” means “secret or covered” and the word “graphic” means “writing”. In general, Steganography hides the information such as text or secret messages into another media file such as image, text, sound or video.

The main objective of Steganography [2] is to communicate securely in such a way that the true message is not visible to the observer. That is unwanted parties should not be able to distinguish any sense between cover-image (image not containing any secret message) and stego-image (modified cover-image that containing secret message). Thus the stego-image should not deviate much from original cover-image. Today, Steganography is mostly used on computers with digital data[3] being the carriers and networks.

The main terminologies used in Steganography systems are: the cover image, secret message, secret key, stego key, stego image and embedded algorithm. The cover image is the carrier of message such as image, video, audio or some other digital media. The secret message is the sensitive information which is to be hidden in the suitable cover image. The secret key is usually used to encrypt the message depending on the algorithms used. Stego key[4] is the key which is used to embed the ciphertext into the cover image .Stego image is the image which is obtained by embedding the ciphertext into the cover image. Embedded algorithm is the algorithm used for embedding ciphertext into the image.

Information can be hidden by many different ways in images. To hide information, straight message insertion may encode every bit of information in the image or selectively embed the message in “noisy” areas that draw less attention—those areas where there is a great deal of natural color variation. The message may also be scattered randomly throughout the image .A number of ways exist to hide information in digital images. Common approaches include

- Least significant bit insertion,
- Pixel embedding using key

1.1 Least significant bit insertion:

LSB[5] is the most popular Steganography technique used. It hides the secret message in the pixels of RGB image based on it its binary coding. LSB changes the image resolution quite clear as well as it is easy to attack. It is clear that LSB changes the image resolution when the least significant bits add in the binary image format, so that image quality become burst and there become so much difference in the original image and encoded image in the respect of image quality.

The following figure presents an example about pixel values and shows the secret message.LSB[5] is the most popular Steganography technique used. It hides the secret message in the pixels of RGB image based on it its binary coding.The following figure presents an example about pixel values and shows the secret message.

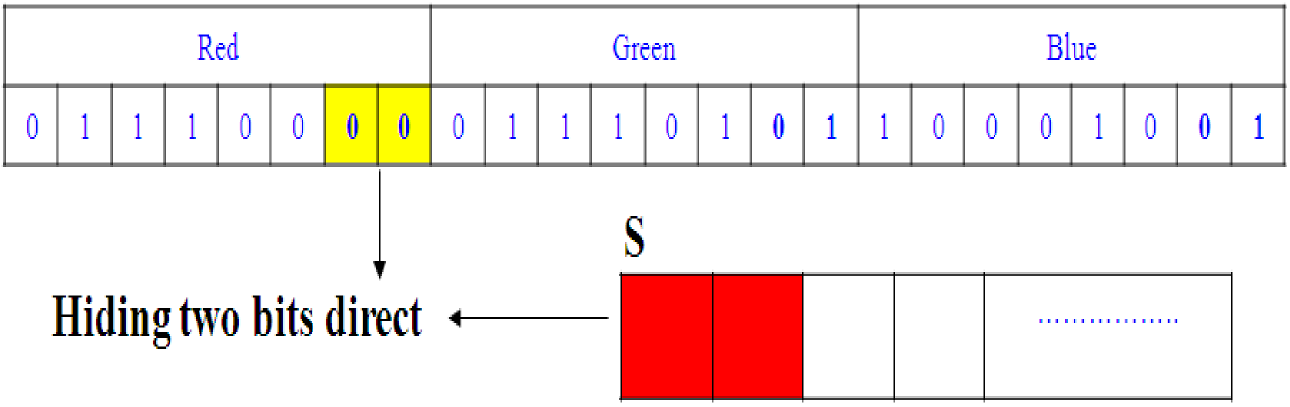


Figure 1: Least significant bit insertion

1.2 Pixel embedding using key:

In this method the secret message is embedded in the pixels of RGB image based on the key shared between sender and receiver. The key is embedded in the first pixels of the original image. . The receiver first extracts the key and using the pattern in the stego key, data is retrieved from the pixels of Stego image. But this method has some drawbacks like if the key shared between two parties are compromised then it attacks the confidentiality. As the key is sent in the first pixels of the stego image it can be intercepted and modified by the intruder.

In this paper a new Steganography system is presented, implemented and analyzed. The proposed method initially encrypts the given original message using strong encryption algorithm and the ciphertext thus obtained is embedded into the image by selecting any one of the multiple stego keys (key number). By using this technique the cover image and the stego image looks similar and the resolution of the image is also not changed as least significant bit insertion .Thus the message is provided high security by applying both the techniques.

2. DEVELOPMENT OF A PROCEDURE FOR STEGANOGRAPHY

In this paper a new Steganography method is presented. First, the message is converted into ciphertext using a strong encryption algorithm using secret key. In this proposed Steganography system, sender and receiver share multiple stego keys. Among the multiple stego keys, any one of the key is used for steganography. The information about the stego key (key number) is embedded in the first pixels of the original image. Based on the pattern given in the stego key, ciphertext is embedded in the pixels of original image. The receiver first extracts the key number and then obtains the stego key from the database. Using the pattern given in the stego key, ciphertext is retrieved from the pixels of Stego image. The ciphertext is decrypted using the secret key. Thus, it is difficult to recognize that some information is embedded in the cover image, called Stego image, which is same as original image.

The proposed Steganography system is explained by using the following example

Example:
Let us consider the following four stego keys which are shared between sender and receiver

K1[]={‘r’,’b’,’g’,’g’,r’}

K2[]={’g’,’g’,r’,’b’,’r’}

K3[]={‘r’,’g’,’g’,’b’,’r’}

K4[]={’g’,r’,’b’,’b’,’g’}

Now, assume that the sender selects the key 1. The cipher text is first converted into byte array. And then the values are embedded in the pattern which is mentioned in the key.
Let us say,

Plaintext: security

Secret key: network

Ciphertext: Yãd6K<Ë¸

Byte array: [89 243 100 54 75 60 203 184]

Pixels of cover image:

(11001000 00100111 11101001)
(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
(11001110 00100111 10101001)
(11001000 10100111 10001001)

Pixels of stego image:

(00000001 00100111 11101001)…… key number in r
(01011001 11101001 11001000) …… r
(00100111 11001000 11110011) …… b
(11001000 01100100 11101001) ……. g
(11001110 00110110 10101001) ……. g
(01001011 10100111 10001001) …….. r

The underlined pixel attribute in first pixel indicate the embedded key number. And the underlined pixel attributes in rest pixels indicate ciphertext in the pixels of cover image. The subsequent ciphertext will follow in similar way as discussed above.

The receiver retrieves the key number from the pixel and the ciphertext from the stego image as given by the pattern in the key used. The ciphertext obtained is decrypted by using the respective decryption algorithm for the encryption algorithm used at sender.

As the multiple keys are used in the proposed system it is robust against the attacks.

Figure 2: Cover image

Figure 3: Stego image

2.1 Algorithms for the proposed system:

(a)Algorithm for embedding text into the cover image:

Step1: Start

Step2: Accept the plaintext as input.

Step3: Encrypt the plaintext using strong encryption algorithm to get the ciphertext using
secret key.

Step4: Accept the image in which the ciphertext or the sensitive message is to be embedded
i.e the cover image.

Step5: Calculate the length of the ciphertext and store in a variable cipherlen.

Step6: The value of cipherlen is embedded in the green attribute of first pixel of the cover image.

Step7: Choose any random key from the multiple stego keys.

Step8: The selected key number is embedded in the red attribute of first pixel of the cover image.

Step9: Check whether the cipherlen is larger than key length or not. If the keylength is larger than cipherlen then based on the pattern given in the stego key the ciphertext is embedded into the image

Step10: If the cipherlen is larger than keylength then the ciphertext is divided into blocks of keylength size and the ciphertext is embedded into the image based on the pattern in the selected random stego key.

Step11: With the new pixel values generate stego image.

Step12: The image generated using proposed steganographic system is sent to the receiver.

Step13: Stop.

(b)Algorithm for retrieving plaintext from Stego image:

Step1: Start

Step2: Accept the stego image generated by the sender
.

Step3: Read the cipherlen from the first pixel of the stego image.

Step4: Read the key number embedded into the stego image from first pixel of stego image.

Step5: Check whether the cipherlen is larger than key length or not. If the keylength is larger than cipherlen then based on the pattern given in the stego key the ciphertext is retrieved from the image

Step6: If the cipherlen is larger than keylength then the pixels is divided into blocks of keylength size and the ciphertext is retrieved from these blocks of pixels based on the pattern in the stego key.

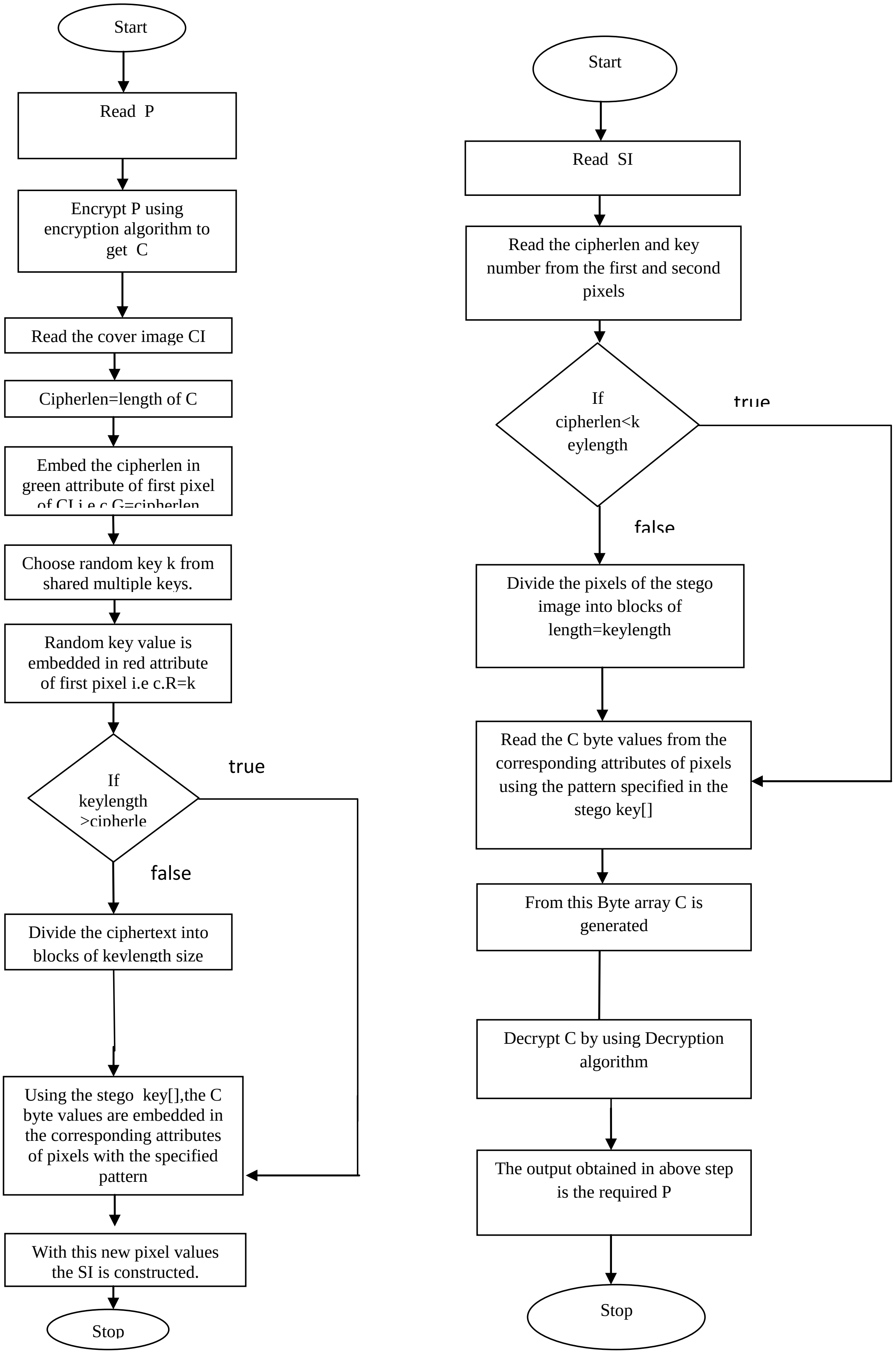
Step7: The ciphertext is thus obtained from the Stego image.

Step8: Decrypt the ciphertext by using the corresponding decryption algorithm using secret key.

Step9: The output obtained from the decryption is the required plaintext or the sensitive information.

Step10: Stop.

2.2 Flow charts:P=Plaintext, C=Ciphertext, c=Pixel color, CI=Cover Image, SI=Stego Image

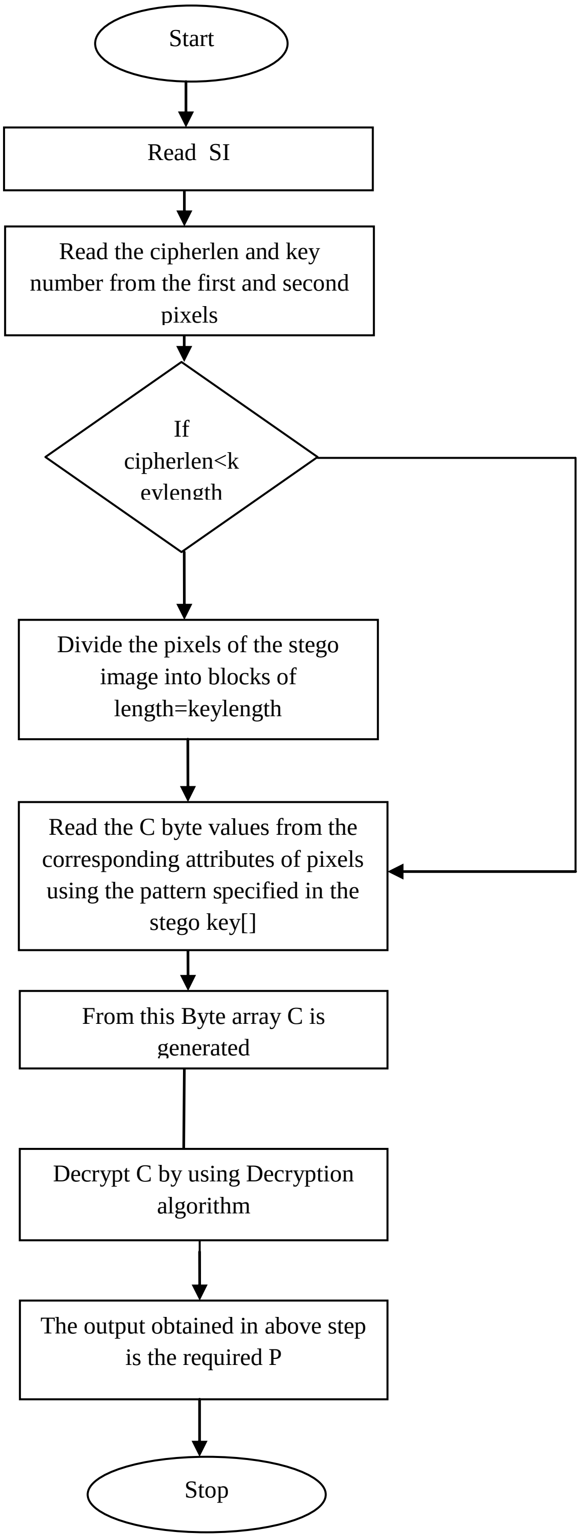


(a) Process of embedding text in Cover image

Start
Read P
Encrypt P using encryption algorithm to get C
Read the cover image CI
Cipherlen=length of C
Embed the cipherlen in green attribute of first pixel of CI i.e c.G=cipherlen
Choose random key k from shared multiple keys.
Random key value is embedded in red attribute of first pixel i.e c.R=k
If keylength >cipherle
true
false
Divide the ciphertext into blocks of keylength size
Using the stego key[],the C byte values are embedded in the corresponding attributes of pixels with the specified pattern
With this new pixel values the SI is constructed.
Stop

(b) Process of Retrieving Plaintext from Stego image

Start
Read SI
Read the cipherlen and key number from the first and second pixels
If cipherlen<keylength
true
false
Divide the pixels of the stego image into blocks of length=keylength
Read the C byte values from the corresponding attributes of pixels using the pattern specified in the stego key[]
From this Byte array C is generated
Decrypt C by using Decryption algorithm
The output obtained in above step is the required P
Stop


Start
Read SI
Read the cipherlen and key number from the first and second pixels
If cipherlen<k eylength
Divide the pixels of the stego image into blocks of length=keylength
Read the C byte values from the corresponding attributes of pixels using the pattern specified in the stego key[]
From this Byte array C is generated
Decrypt C by using Decryption algorithm
The output obtained in above step is the required P
Stop

3. ILLUSTRATION OF PIXEL EMBEDDING IN AN IMAGE

Let us consider a cover image C which is given below.

Figure 4: Cover image

The sensitive message that has to be embedded in the cover image be “security”.

Plaintext: security

Secret key: data embedding.123456789 (key size:192 bits)

Ciphertext: 7AsrYfRN/opJjg3FxHMHvA==

This ciphertext is converted into byte array and the data is embedded in the pixels using the proposed system. The output is the Stego image which contains the sensitive information in the pixels. The Stego key used in embedding process is given below.

Stego key: {‘r’,’g’,’b’,’r’,’g’,’b’}

The Stego image obtained after embedding is as shown below.

Figure 5: Stego image

4. RESULTS

4.1 Sender side:

Input: The plaintext that has to be embedded in the cover image is given as input for the proposed system.

Plaintext: security

Secret key: data embedding.123456789 (key size:192 bits)

Ciphertext: 7AsrYfRN/opJjg3FxHMHvA==

This ciphertext is converted into byte array and this data is embedded in the pixels using the algorithm mentioned in section 2(b). The following image is the cover image which is given as input. In this cover image the secret information which is to be sent is embedded using stego key.

Stego key: {‘b’,’g’,’r’,’b’,’r’,’g’}

Figure 6: Sender’s input image

Output: The following image is the output image or the Stego image obtained after the secret information is embedded. We can observe that the cover image and the Stego image are similar.

Figure 7: Sender’s output image

4.2 Receiver side:

Input: The following Stego image is given as input to the proposed system at receiver side. By using the algorithm mentioned in section 2(b) the ciphertext is retrieved from the stego image using stego key.

Stego key: {‘b’,’g’,’r’,’b’,’r’,’g’}

Figure 8: Receiver’s output image

Output: The ciphertext is obtained as output is decrypted using the same algorithm used at sending side and plaintext is obtained using the same secret key used in encryption.

Ciphertext: 7AsrYfRN/opJjg3FxHMHvA==

Secret key: data embedding.123456789 (key size:192 bits)

Plaintext: security

5. CONCLUSION

In this paper a new steganography method is presented. First, the message encrypted using a strong encryption algorithm with the help of secret key. In this proposed Steganography system, sender and receiver share multiple stego keys. Among the multiple stego keys, any one of the stego key is used for steganography. The information about the stego key (key number) is embedded in the first pixels of the original image. Based on the pattern given in the stego key, ciphertext is embedded in the pixels of original image. The receiver first extracts the key number and then obtains the stego key from the database. Using the pattern given in the stego key, ciphertext is retrieved from the pixels of Stego image. The ciphertext is decrypted using the secret key. Thus, it is difficult to recognize that some information is embedded in the cover image, called Stego image, which is same as original image. The proposed system is simple and robust against the attacks.

REFERENCES

[1] Abdelmgeid Amin Ali and Al – Hussien Seddik Saad “*Image Steganography Technique By Using Braille method of Blind People (LSBraille)*”.

[2] Akanksha Kaushal, Prof Vineeta Chaudhary “*A Secure Data Hiding Technique: Steganography*”, International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 2, Issue 10, October 2013.

[3] A. Nag , S. Biswas , D. Sarkar , P. P. Sarkar "*A novel technique for image steganography based on Block-DCT and Huffman Encoding*", International Journal of Computer Science and Information Technology, Volume 2, Number 3, June 2010.

[4] Chamkor Singh and Gaurav Deep "*Cluster Based Image Steganography Using Pattern Matching*", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS).

[5] Neil F.Johnson,Sushil Jojodia – "*Exploring Steganography : Seeing the Unseen*" IEEE computer, February 1998 pp26-34.

[6] R. Chandramouli, Nasir Memon, "*Analysis of LSB Based Image Steganography Techniques*" Proc. IEEE ICIP pp1019-1022, 2001.

Aruna Varanasi is presently working as Professor and head in the Department of Computer Science and Engineering (CSE), Sreenidhi Institute of Science and Technology (SNIST), Hyderabad, India. She was awarded "Suman Sharma" by Institute of Engineers (India), Calcutta for securing highest marks among women in India in AMIE course.

M.Lakshmi Anjana is presently pursuing Bachelors of technology in Computer Science and technology from Sreenidhi Institute of Science and technology(SNIST), Hyderabad ,India.

Pravallika Pasupulate is presently pursuing Bachelors of technology in Computer Science and technology from Sreenidhi Institute of Science and technology(SNIST), Hyderabad ,India.