VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Electronics Engineering (SENSE)

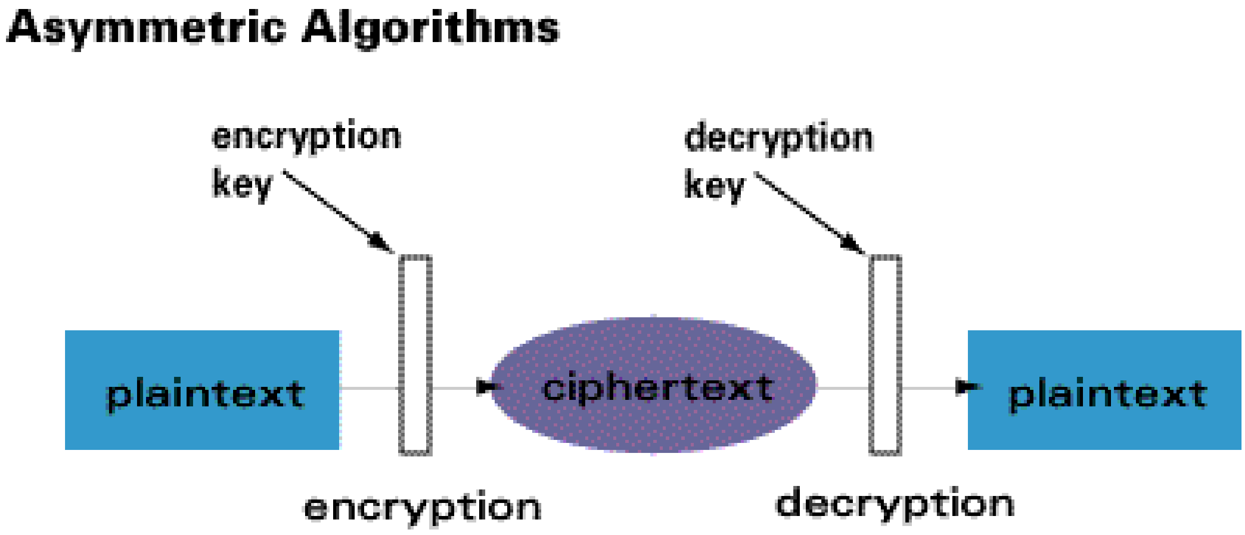
PROJECT BASED LEARNING (CAMP) - REPORT

COURSE CODE / NAME	BECE306L – DIGITAL COMMUNICATION SYSTEMS		
PROGRAM / YEAR	B.Tech (Electronics and Communication Engineering)		
LAST DATE FOR REPORT SUBMISSION			
DATE OF SUBMISSION			
TEAM MEMBERS DETAILS	**REGISTER NO.**	**NAME**	
	21BEC1009	MONISH BABU A	
J TITLE	**INTEGRATION OF CRYPTOGRAPHY AND STEGANOGRAPHY FOR SECURE COMMUNICATION**		
COURSE HANDLER'S NAME	**Ralph Samuel Thangaraj**	**REMARKS**	
COURSE HANDLER'S SIGN			

ABSTRACT:

In today's technological landscape, ensuring information security remains a challenging task. Over the past decade, various strategies have been proposed to protect the transmission of confidential data over the public network, such as the Internet. One effective solution to address this concern involves combining steganography and cryptography.

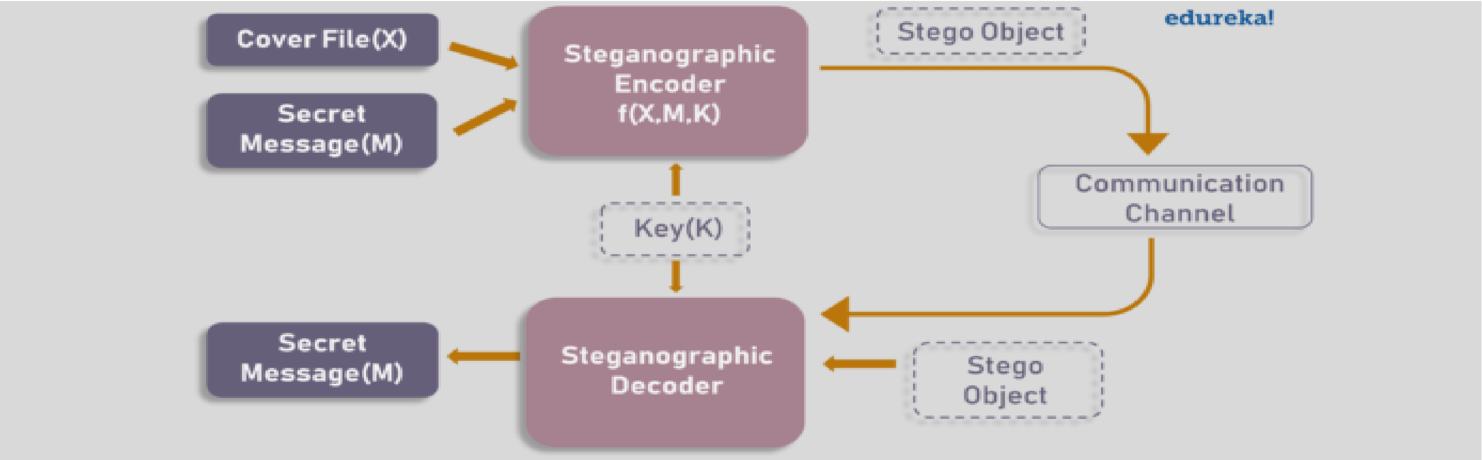
In this approach, multiple encryption algorithms like bitxor operation, bits shuffling, and stego key-based encryption are utilized to encrypt both the secret key and confidential information. These encrypted elements are then concealed within the pixels of a host image. Additionally, before data concealing, the input audio/image undergoes transposition, adding an extra layer of security. By employing techniques like image transposition, bits shuffling, bitxoring, stego key-based encryption, and gray-level change, the proposed method achieves five distinct security levels, significantly enhancing data retrieval difficulty. To generate and combine different shares, the KN share AES-based technique is employed. Objective analysis using various picture quality assessment metrics is conducted to evaluate the suggested approach, which demonstrates promising results concerning imperceptibility and security. The efficacy of the proposed method is further confirmed by high-resolution stego images.



INTRODUCTION:

In today's modern world, safeguarding confidential data has become of utmost importance, and steganography has emerged as an advanced technique for securing such information. Steganography involves concealing secret data, which could be in the form of messages, audio, pictures, or videos, within other images, audios, messages, or videos. Its purpose is to protect sensitive information from potential threats. By embedding encrypted data, the steganography process ensures that the presence of hidden information remains undetected. A steganography system consists of three main components: the cover-object (e.g., an image file), the secret data, and the resulting stego-object (e.g., stego-image, stego-video, stego-audio, or stego-text).

Various coverings can be employed to hide the concealed information, and LSB-based algorithms are commonly used for optimization in many applications. To encrypt the secret data, cryptography is often utilized, with integrity, authenticity, and secrecy being its primary objectives. The stego-image generated by steganography is further processed using KN Share Cryptography technology, which employs AES encryption and decryption. In this method, photos are split into N shares and combined using a K number (K=N), making the KN Share technique highly efficient.



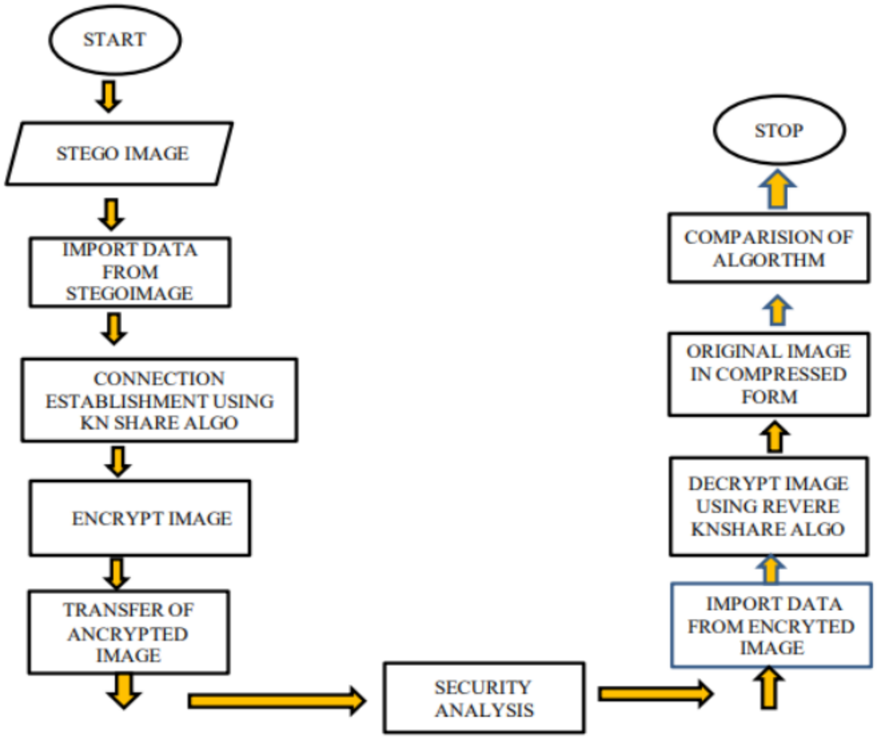
ALGORITHM:

- Start
- Clear command window, workspace, and figures
- Read the carrier image and display it
- Read the secret image and display it
- Get the size of the carrier image
- Resize the secret image to match the size of the carrier image
- Extract color planes from the carrier and secret images
- Generate a secret key
- Encrypt the secret message by performing bitwise XOR with the secret key
- Display the encrypted secret message
- For each pixel in the carrier image:
  a. Extract the least significant bit (LSB) of the red plane of the carrier image
  b. Extract the most significant bit (MSB) of the red plane of the secret image
  c. Divide the MSB by 128 to convert it to a decimal value between 0 and 1
  d. Add the decimal value to the extracted LSB
- Store the resulting values as the red plane of the steganographic image
- Repeat steps for the green and blue planes of the carrier image
- Combine the modified color planes into the final steganographic image
- Display the steganographic image
- For each pixel in the steganographic image:
- a. Extract the LSB of the red plane and multiply it by 128 to convert it back to an 8-bit value
- b. Store the result as the recovered red plane
- Repeat step 17 for the green and blue planes of the steganographic image

- Combine the recovered color planes into the output image
- Display the recovered encrypted image from steganography
- Decrypt the recovered image by performing bitwise XOR with the secret key
- Display the decrypted secret message signal
- End

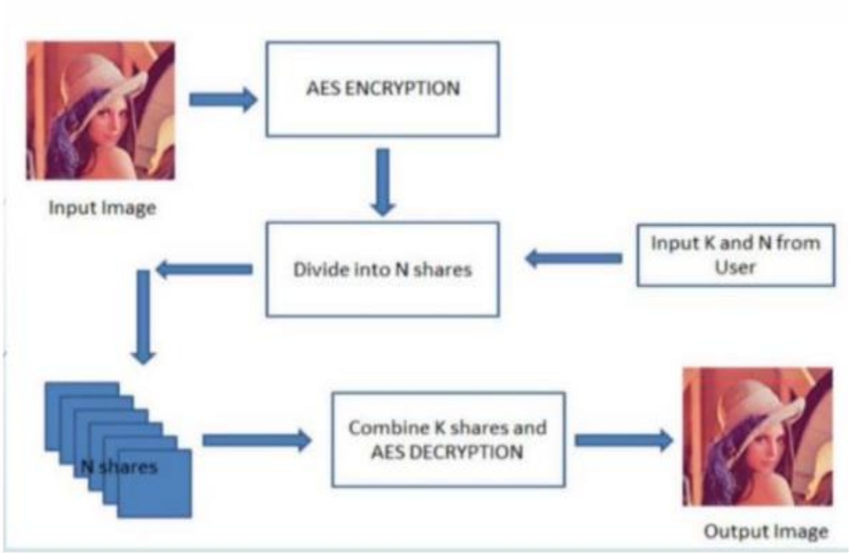
IMPLEMENTATION (REAL TIME):

We need a secret image and a cover image (a carrier image made up of RGB colors) for the steganography process. To do this, we are using the Least Significant Bit (LSB) based method to embed both images, which are actually stego images. These operations are carried out using the steganography system and the key algorithm. By using the reverse steganography technique, we may extract the hidden image from the stego image.

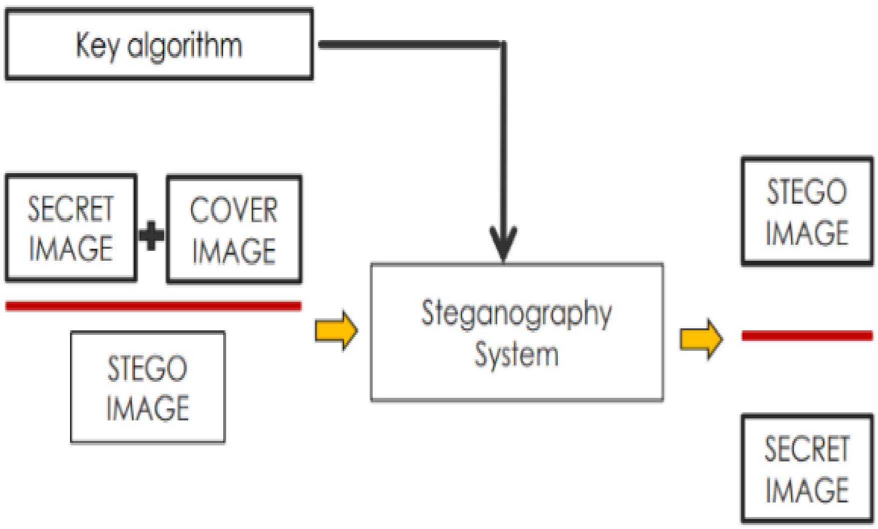

START
STEGO IMAGE
IMPORT DATA FROM STEGOIMAGE
CONNECTION ESTABLISHMENT USING KN SHARE ALGO
ENCRYPT IMAGE
TRANSFER OF ANCRYPTED IMAGE
SECURITY ANALYSIS
IMPORT DATA FROM ENCRYTED IMAGE
DECRYPT IMAGE USING REVERE KNSHARE ALGO
ORIGINAL IMAGE IN COMPRESSED FORM
COMPARISION OF ALGORTHM
STOP

The stego picture obtained by the LSB-based steganography procedure is extracted in the figure. The KN Share key algorithm is then used to establish the connection. The encrypted image is transmitted and put through two passes of security inspection. The original

data form of the image, which was encrypted, was then sent on to the following step. The Reverse KN Share algorithm is used to further decrypt the original encrypted image. The original, compressed version of the encrypted image is provided.
We use any photograph that needs to be shared in secrecy for this project. A safe encryption key provided by the user is used to encrypt this image. Additionally, the K N Secret Sharing Algorithm divides the encrypted image into N distinct shares.



These N shares can be spread, but only K of these shares are necessary for the end user to produce the original image. The original image is still in encrypted form once it is produced. The image may now only be decrypted using the same key that was originally used to encrypt it, adding another layer of security.



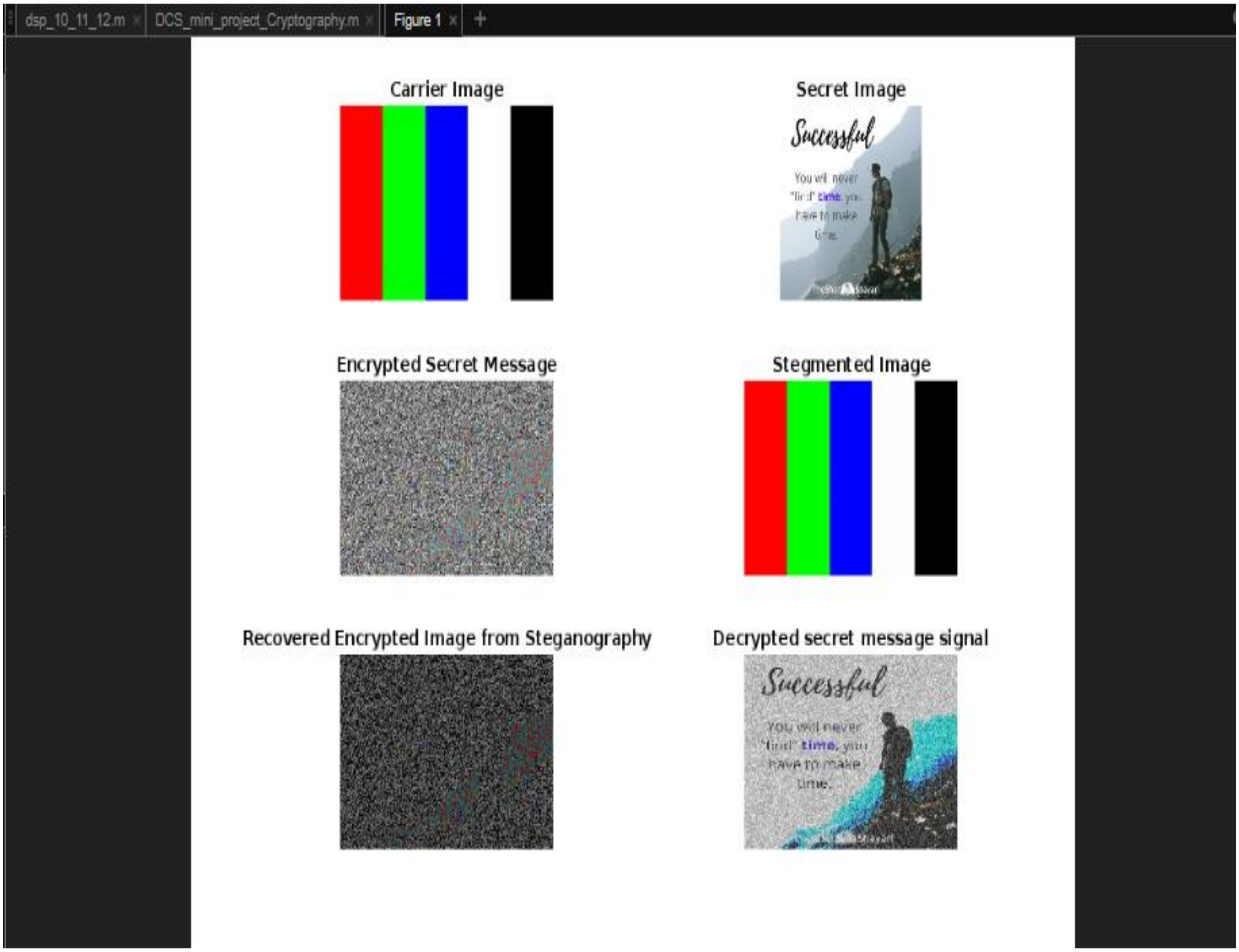
CODING:

```
clc
clear all
close all
warning off
a=imread('testimage.png'); %File name of the carrier image
nexttile;
imshow(a); %Printing of the carrier image
title('Carrier Image');
x=imread('SECRET.png'); %File name of the secret image
nexttile;
imshow(x); %Printing of the image
title('Secret Image');
[r c g]=size(a); %Getting the size of the test image
x=imresize(x,[r c]); %Resizing the message and carrier signal
ra=a(:,:,1); %Extracting the red plane of the carrier signal
ga=a(:,:,2); %Extracting the green plane of the carrier signal
ba=a(:,:,3); %Extracting the blue plane of the carrier signal
rx=x(:,:,1); %Extracting the red plane of the message signal
gx=x(:,:,2); %Extracting the green plane of the message signal
bx=x(:,:,3); %Extracting the blue plane of the message signal
sk=uint8(rand(r,c)*255);%Secret key
rx=bitxor(rx,sk); %Performing BITXOR operation with red plane message
signal and the secret key
gx=bitxor(gx,sk); %Performing BITXOR operation with green plane
message signal and the secret key
bx=bitxor(bx,sk); %Performing BITXOR operation with blue plane message
signal and the secret key
nexttile;
imshow(cat(3,rx,gx,bx));
title('Encrypted Secret Message');

%Extracting and adding the image with carrier for red plane
(Encryption)
for i=1:r
    for j=1:c
       nc(i,j)= bitand(ra(i,j),254); %Performing BITAND operation with
carrier signal pixels and 254
       ns(i,j)= bitand(rx(i,j),128); %Performing BITAND operation with
message signal pixels and 128
       ds(i,j)=ns(i,j)/128; %Extracting only the image pixels
       fr(i,j)=nc(i,j)+ds(i,j); %Adding the extracted image with the
carrier
```

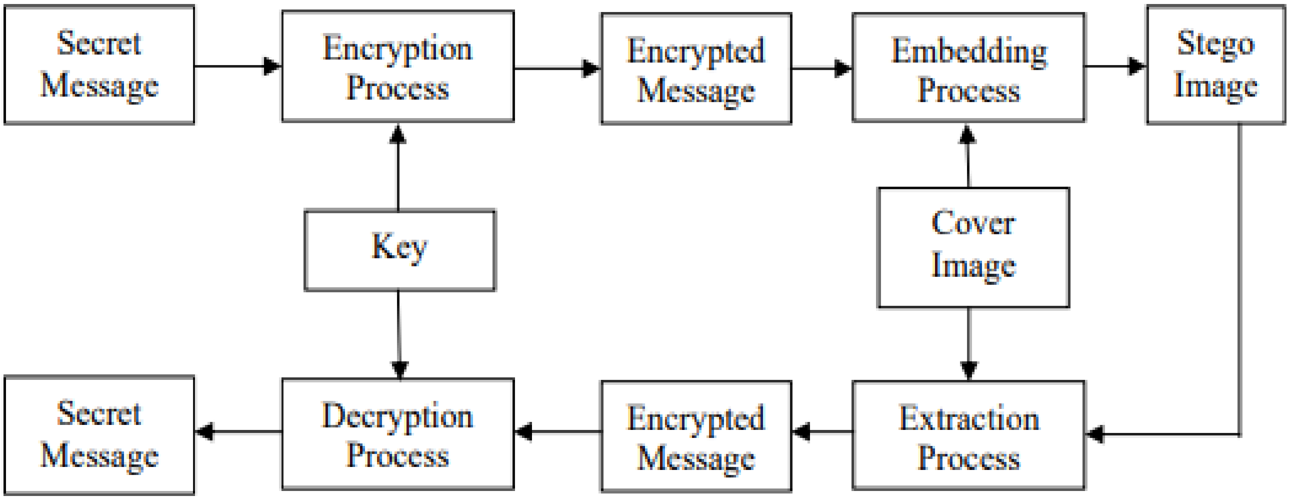
```
    end
end %End of loops

%Extracting and adding the image with carrier for green plane
(Encryption)
redsteg=fr;
for i=1:r
    for j=1:c
       nc(i,j)= bitand(ga(i,j),254);
       ns(i,j)= bitand(gx(i,j),128);
       ds(i,j)=ns(i,j)/128;
       fr(i,j)=nc(i,j)+ds(i,j);
    end
end

%Extracting and adding the image with carrier for blue plane
(Encryption)
greensteg=fr;
for i=1:r
    for j=1:c
       nc(i,j)= bitand(ba(i,j),254);
       ns(i,j)= bitand(bx(i,j),128);
       ds(i,j)=ns(i,j)/128;
       fr(i,j)=nc(i,j)+ds(i,j);
    end
end
bluesteg=fr;
finalsteg=cat(3,redsteg,greensteg,bluesteg); %Combining all the
extracted images

%Decryption operation for the receiver’s end
redstegr=finalsteg(:,:,1);
greenstegr=finalsteg(:,:,2);
bluestegr=finalsteg(:,:,3);
nexttile;
imshow(finalsteg);
title('Stegmented Image');

%Extracting the red plane and performing decryption
for i=1:r
    for j=1:c
        nc(i,j)=bitand(redstegr(i,j),1); %BITAND operation of red
pixel and 1
        ms(i,j)=nc(i,j)*128; %Multiplying them with 128
    end
```

```
end
recoveredr=ms; %Recovered part

%Extracting the green plane and performing decryption
for i=1:r
    for j=1:c
        nc(i,j)=bitand(greenstegr(i,j),1); %BITAND operation of green
pixel and 1
        ms(i,j)=nc(i,j)*128;
    end
end
recoveredg=ms; %Recovered part

%Extracting the blue plane and performing decryption
for i=1:r
    for j=1:c
        nc(i,j)=bitand(bluestegr(i,j),1); %BITAND operation of blue
pixel and 1
        ms(i,j)=nc(i,j)*128;
    end
end
recoveredb=ms; %Recovered part
output=cat(3,recoveredr,recoveredg,recoveredb); %Concatenating all the
recovered output
nexttile;
imshow(output); %Displaying the recovered output
title('Recovered Encrypted Image from Steganography');
red_band=bitxor(output(:,:,1),sk); %Decrypting the message of red
plane with secret key
green_band=bitxor(output(:,:,2),sk); %Decrypting the message of green
plane with secret key
blue_band=bitxor(output(:,:,3),sk); %Decrypting the message of blue
plane with secret key
combined=cat(3,red_band,green_band,blue_band); %Concatenating the
decrypted images
nexttile;
imshow(combined); %Displaying the decrypted image
title('Decrypted secret message signal');
```

RESULTS & INFERENCES:


dsp_10_11_12.m
DCS_mini_project_Cryptography.m
Figure 1
Carrier Image
Secret Image
Encrypted Secret Message
Stegmented Image
Recovered Encrypted Image from Steganography
Decrypted secret message signal

```
Figure denotes Stegmented image which can be obtained by LSB
based steganography method. After Applying Reverse Steganography
the Recovered Image will be obtained.
```



The above figure represents the working of the implementation that has been done using MATLAB. The message is hidden through DWT-based steganography utilizing the Haar wavelet. A spatial domain is transformed into a frequency domain using a wavelet algorithm. Approximation coefficients, horizontal detail coefficients, vertical detail coefficients, and diagonal detail coefficients are the four sub-images that make up the cover image in this scenario. To conceal the message, the embedding process is completed at the transmitter.

APPLICATION ORIENTED LEARNING:

1. Confidential communication and secret data storing

The "secrecy" of the embedded data is essential in this area.

Historically, steganography have been addressed in this area. Steganography provides us with:

(A) Potential capability to hide the existence of confidential data
(B) Hardness of detecting the hidden (i.e., embedded) data
(C) Enhancing the secrecy of the encrypted data

In practical application, steganography involves a series of steps. Initially, you choose a harmless vessel data that suits the size of the data you want to embed. Subsequently, you use an embedding program (a component of the steganography software) along with a specific key to embed the confidential data. When it comes to extraction, you or the intended recipient utilize an extracting program (another component) and the same key ("common key" akin to cryptography) to recover the embedded data. To facilitate confidential communication, a "key negotiation" is necessary with the recipient before initiating the process.

2. Protection of data alteration

We acknowledge the vulnerability of the embedded data in this particular application field. On the Home Page, we emphasized that the embedded data tends to be delicate rather than highly robust, a characteristic shared by most steganography programs. Notably, the Qtech Hide & View program is particularly prone to embedding data in an exceedingly fragile manner, a demonstration of which can be found on another page.

Nevertheless, this fragility presents a unique opportunity for the development of an information-alteration protective system, like the "Digital Certificate Document System." One of its most ground-breaking aspects is that it eliminates the need for an authentication bureau. Once implemented, individuals can securely transmit their "digital certificate data" worldwide through the Internet. Any attempts at forging, altering, or tampering with such certificate data would be readily detected by the extraction program.

CONCLUSION:

The main objective of this project centers around ensuring the security and confidentiality of data. To achieve this, a novel approach to image Steganography and Cryptography is implemented to enhance security. The primary technique employed is LSB based Steganography, while KN share based cryptography is used as a secondary method. Both techniques are combined to bolster overall security.

Over the last decade, data hiding technologies have seen significant progress and widespread adoption. The rise of smart mobile devices has underscored the importance of safeguarding valuable proprietary information, leading to the development of numerous methods and technologies for both ethical and unethical purposes. Particularly concerning are approaches that combine hiding methods with cryptography, allowing information to remain concealed even if the channel is compromised.

Many vendors offer robust privacy protection technologies for desktops, and modern smart mobile platforms such as Android and iPhone come equipped with built-in cryptographic capabilities. However, the real danger lies in data hiding techniques that exploit vulnerabilities in multimedia and protocols, enabling covert communication and information concealment. These emerging methods represent hybrid solutions that effectively blend the strengths of cryptography and steganography. Unfortunately, the threat posed by these techniques often goes unnoticed or unaddressed, allowing their interest, innovation, and advancement to persist unchecked.

REFERENCES:

[1] Flevina Jonese d'souza Advanced Encryption Standard (AES) Security Enhancement using Hybrid Approach.

[2] K Thangadural 2014-An analysis of LSB Based Image Steganography Techniques.

[3] Nitendra Kumar Vishwakarma et.all 2013 -Comparative Analysis of Different Techniques. [4] Rupali Bhardwaj et.all 2015- Enhanced the Security of Image Steganography Through Image Encryption.

[5] Radha S. phadte et.all 2017-Enhanced Blend of Image Steganography and Cryptograph.

[6] INTEGRATION OF STEGANOGRAPHY AND CRYPTOGRAPHY USING MATLAB FOR SECURE DATA COMMUNICATION by Mr. Sharath Tumminakatti, Mr. Suraj Patil, Vinayak Athreya, Mr. Vishnu