

**UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY,  
C.S.J.M. UNIVERSITY, KANPUR**



**B. TECH PROJECT FILE  
ON  
IMAGE STEGANOGRAPHY  
USING MATLAB**

Supervised by-  
Mr. Sunil Kumar

Submitted by-  
Ashish Kumar Srivastava  
-CSJMA14001390136  
Tanisha Srivastava  
-CSJMA14001390175

## **CONTENTS**

<b>S.no.</b>	<b>Topics</b>
1.	Introduction
2.	Literature Survey
3.	Proposed Methodology
4.	System Analysis and Design
5.	Code Analysis
6.	Screenshots of the workflow
7.	Summary
8.	References

# **INTRODUCTION**

## **Steganography**

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information. Many different carrier file formats can be used, but digital images are the most popular because of their frequency on the internet. For hiding secret information in images, there exists a large variety of steganography techniques some are more complex than others and all of them have respective strong and weak points. Different applications may require absolute invisibility of the secret information, while others require a large secret message to be hidden. This project report intends to give an overview of image steganography, its uses and techniques. It also attempts to identify the requirements of a good steganography algorithm and briefly reflects on which steganography techniques are more suitable for which applications.

One of the reasons that intruders can be successful is the most of the information they acquire from a system is in a form that they can read and comprehend. Intruders may reveal the information to others, modify it to misrepresent an individual or organization, or use it to launch an attack. One solution to this problem is, through the use of steganography. Steganography is a technique of hiding information in digital media. In contrast to cryptography, it is not to keep others from knowing the hidden information but it is to keep others from thinking that the information even exists.

Steganography become more important as more people join the cyberspace revolution. Steganography is the art of concealing information in ways that prevents the detection of hidden messages. Steganography include an array of secret communication methods that hide the message from being seen or discovered.

# **LITERATURE SURVEY**

## **Overview**

The word steganography comes from the Greek “Seganos”, which mean covered or secret and – “graphy” mean writing or drawing. Therefore, steganography mean, literally, covered writing. It is the art and science of hiding information such its presence cannot be detected and a communication is happening. A secrete information is encoding in a manner such that the very existence of the information is concealed. Paired with existing communication methods, steganography can be used to carry out hidden exchanges.

The main goal of this projects it to communicate securely in a completely undetectable manner and to avoid drawing suspicion to the transmission of a hider data. There has been a rapid growth of interest in steganography for two reasons:

- The publishing and broadcasting industries have become interested in techniques for hiding encrypted copyright marks and serial numbers in digital films, audio recordings, books and multimedia products.
- Government restrictions on the availability of encryption services have motivated people to study methods by which private messages can be embedded in seemingly innocuous cover messages.

The basic model of steganography consists of Carrier, Message and password. Carrier is also known as cover-object, which the message is embedded and serves to hide the presence of the message.

Message is the data that the sender wishes to remain it confidential. It can be plain text, cipher text, other image, or anything that can be embedded in a bit stream such as a copyright mark, a covert communication, or a serial number. Password ensures that only recipient who knows the corresponding decoding will be able

to extract the message from a *cover-object*. The *cover-object* with the secretly embedded message is then called the *Stego-object*.

Recovering message from a *stego-object* requires the *cover-object* itself and a corresponding password if it was used during the encoding process. The original image may or may not be required in most applications to extract the message.

In general, the information hiding process extracts redundant bits from *cover-object*. The process consists of two steps:

- Identification of redundant bits in a *cover-object*. Redundant bits are those bits that can be modified without corrupting the quality or destroying the integrity of the *cover-object*.
- Embedding process then selects the subset of the redundant bits to be replaced with data from a secret message. The *stego-object* is created by replacing the selected redundant bits with message bits.

## **Detecting Steganography:**

The art of detecting Steganography is referred to as **Steg-analysis**.

There are many methods that can be used to detect Steganography such as:

Viewing the file and comparing it to another copy of the file found on the Internet (Picture file). There are usually multiple copies of images on the internet, so you may want to look for several of them and try and compare the suspect file to them. For example if you download a JPED and your suspect file is also a JPED and the two files look almost identical apart from the fact that one is larger than the other, it is most probable you suspect file has hidden information inside of it.

## **PROPOSED METHODOLOGY**

Steganography is a really interesting subject and outside of the mainstream cryptography and system administration that most of us deal with day after day. This can be used for hidden communication. We have explored the limits of steganography theory and practice. We opted out the enhancement of the image steganography system using LSB approach to provide a means of secure communication.

**Least significant bit (LSB)** insertion is a simple approach to embedding information in image file. The simplest steganography techniques embed the bits of the message directly into least significant bit plane of the cover-image in a deterministic sequence.

Digital images used as cover file are mainly of two types- 24-bit images and 8-bit images. In 24-bit images we can embed three bits of information in each pixel. In 8-bit images, one bit of information can be hidden into images. After applying the LSB algorithm the image obtained having secret message is called stego-image. LSB technique as the name implies replaces the least significant bit of the pixel with the information to be hidden. Since LSB is replaced there is no effect on cover image and hence unintended user will not get the idea that some message is hidden behind the image. However , a little change in level of intensity of original and modified pixel, but it cannot be detected visually.

The following example explain how the letter A can be hidden into the three pixels i.e. eight bytes of an 24-bit image.

**Pixels:** (00100111 11101011 11001010)

(00100111 11011000 10101001)

(11001000 00110111 11011001 )

**A:** 010100111

**Result:** (00100110 11101011 11001010)

(00100111 11011000 10101000)

(11001001 00110111 11011001)

The main advantage of LSB method is easy to implement and high message payload and there is less chance of degradation of quality of original image.

Modulating the least significant bit does not result in human perceptible difference because the amplitude of the change is small. In this technique, the embedding capacity can be increased by using two or more least significant bits. At the same time, not only the risk of making the embedded message statistically detectable increase but also the image fidelity degrades. Hence a variable size LSB embedding schema is presented, in which the number of LSBs used for message embedding/extracting depends on the local characteristics of the pixel. The advantage of LSB-based method is easy to implement and high message payload. LSB hides the message in such way that the humans do not perceive it.

## **SYSTEM ANALYSIS AND DESIGN**

Steganography system requires any type of image file and the information or message that is to be hidden. It has two modules encrypt and decrypt.

MATLAB supports all the types of images so there is no specific restriction on the type of image to be used. We have used a png format image which is truecolor i.e. 24 bit image. The idea is to store 3 bits of message in 1 pixel of the image so that it does not reflect any differentiable changes in the stego image.

The algorithm used for Encryption and Decryption in this application provides using several layers lieu of using only LSB layer of image. Writing data starts from last layer (8th or LSB layer); because significance of this layer is least and every upper layer has increased significance from its lower layers. So every step we go to upper layer image quality decreases and image retouching transpires.

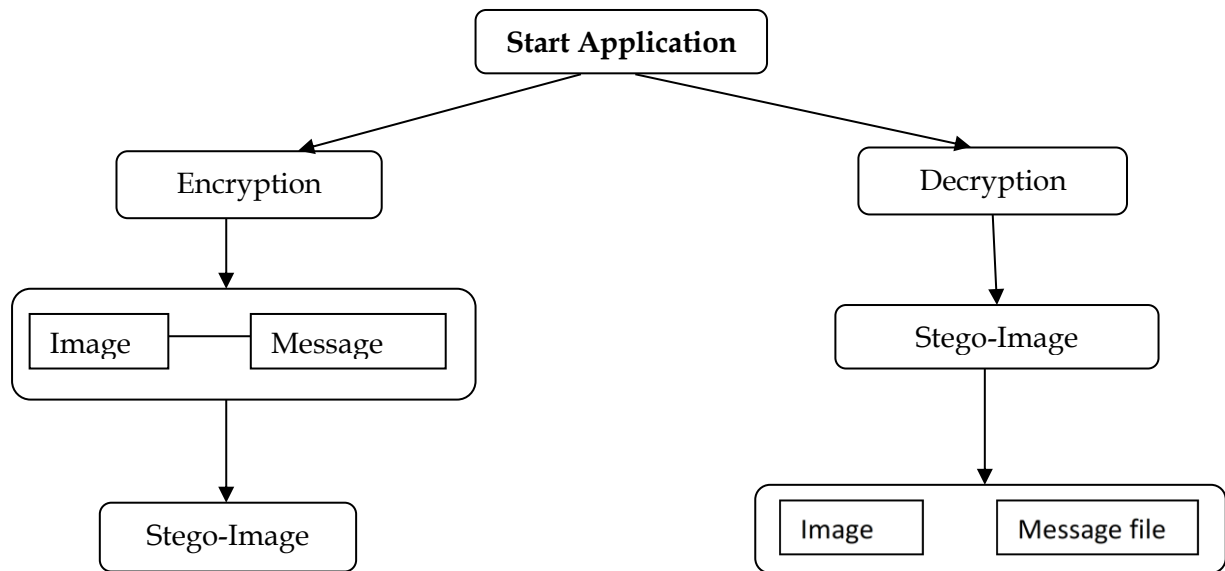
The encrypt module is used to hide information into the image such that no one can see that information or message. This module requires any type of image and a text file that has the message which is to be hidden.

The decrypt module is used to get the hidden information from an image file. It takes the image file as an input, and give the message file that was hidden in that image, as the output in the destination folder.

Before encrypting the message into the image, we must check that the size of the message should not be greater than the size of the image.

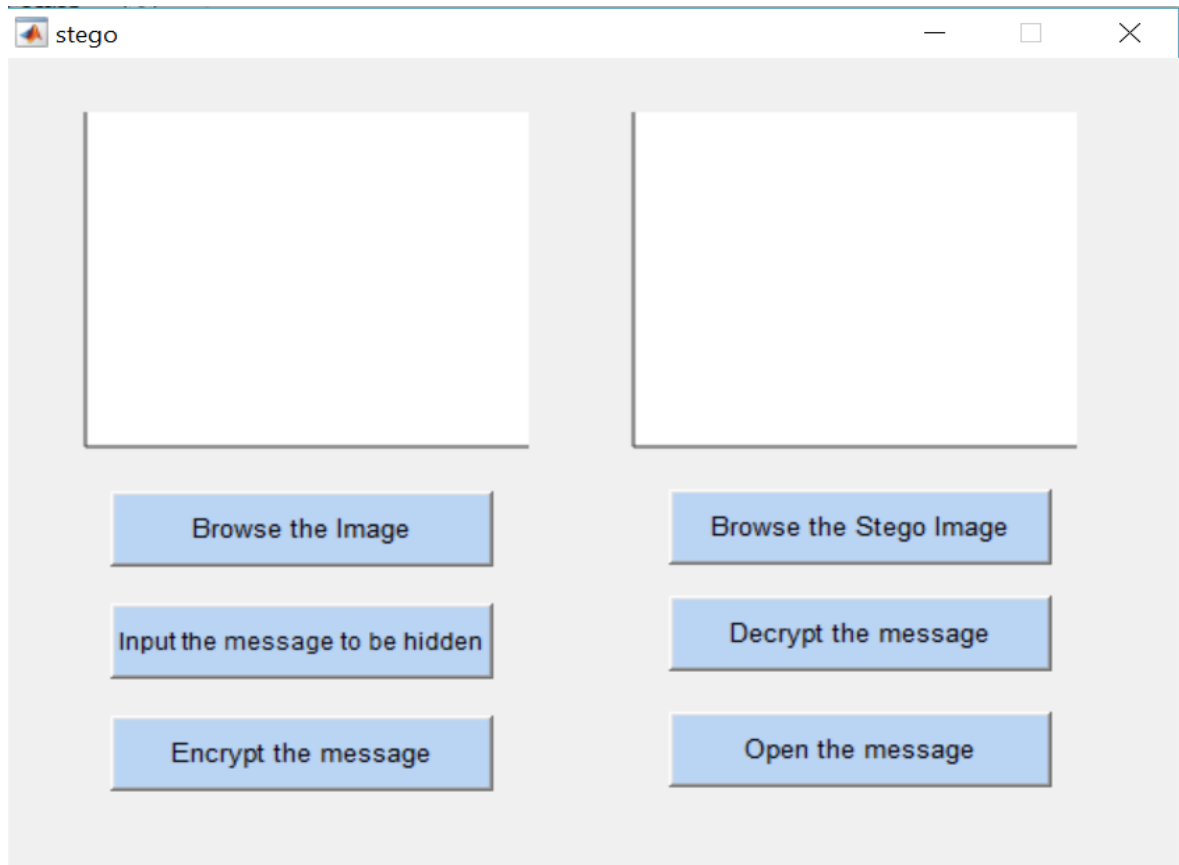


The graphical representation of this system is as follows:-



## CODE ANALYSIS

The **GUI** for the system is as follows-



Here we have used two axes- axes1 and the axes2 which displays the cover image and the stego-image calling of their respective functions.

The **Encryption** side has three pushbuttons-

1. Browse the image- Used to browse and select the cover image from the folder specified.
2. Input the message to be hidden- Opens a text file, in which the message that is to be hidden, is given by the user.
3. Encrypt the message- Finally the algorithm for the encryption of the message processes on click of this button. Then a pop-window shows saying- 'The Message is Encrypted'.

The **Decryption** side has three pushbuttons-

1. Browse the Stego image- Used to browse the stego image which has the message hidden in it.
2. Decrypt the message- Runs the entire decryption algorithm on its click and creates a new text file that stores the message which is extracted from the stego image.
3. Open the message- Opens the text file which shows the message which was hidden inside the stego image.

The **Encryption** process works on the following algorithm-

```
fid=fopen('msg.txt','r+');
f=fread(fid);
bin=transpose(dec2bin(f,8));
bs=bin(:);
len=length(bs);

b=zeros(len,1);
for k=1:len
    if(bs(k)=='1')
        b(k)=1;
    else
        b(k)=0;
    end
end

new=handles.inputimg;
t=new;
h=size(new,1);
w=size(new,2);
s1=h*w*3;

if len> s1
    fprintf('\nImage File Size %d\n',s1);
    fprintf('Text File Size %d\n',len);
    disp('Text File is too Large');
else
    fprintf('\nImage File Size %d\n',s1);
```

```
fprintf('Text File Size %d\n',len);
disp('Text File is Small');

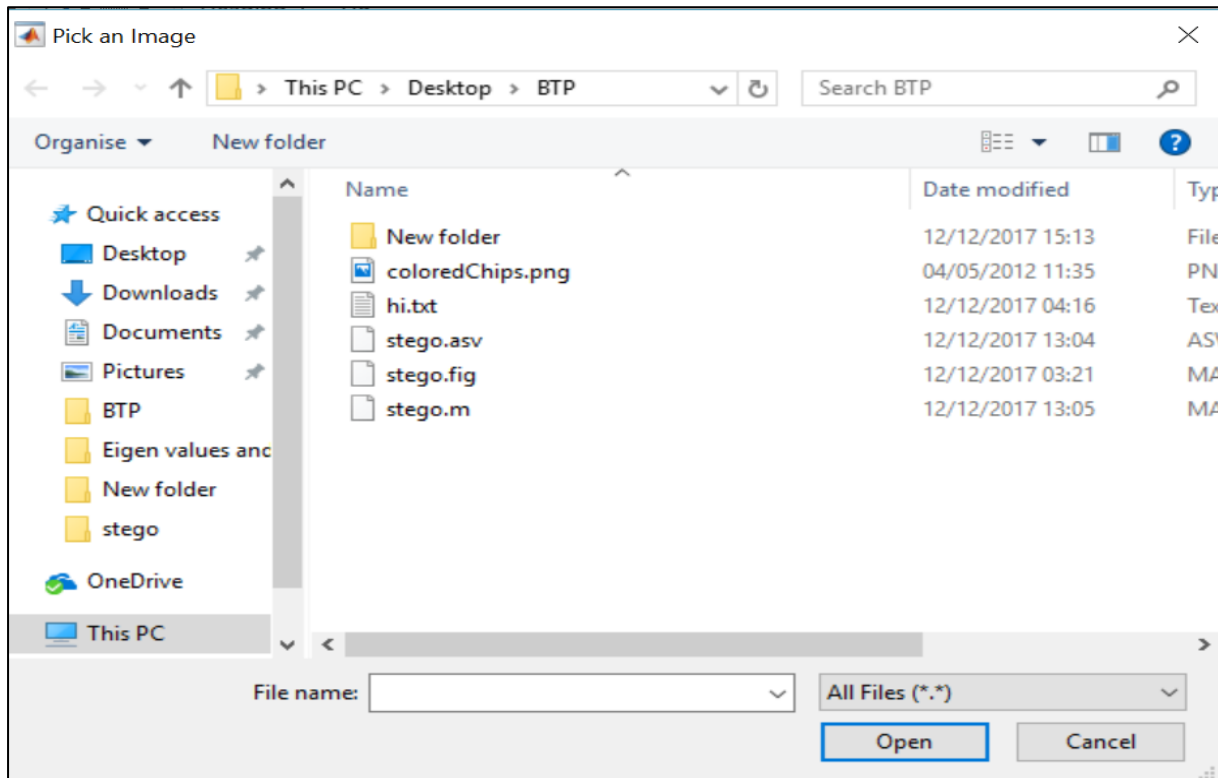
ms=1;
for i=1:h
    for j=1:w
        for k=1:3
            if(ms<=len)
                lsb=mod(double(new(i,j,k)),2);
                if(lsb==1 && b(ms)==0)
                    t(i,j,k)=new(i,j,k)-1;
                elseif(lsb==0 && b(ms)==1)
                    t(i,j,k)=new(i,j,k)+1;
                else
                    t(i,j,k)=new(i,j,k);
                end
            end
            ms=ms+1;
        end
    end
end
imwrite(t,'hide.png');
end
msgbox('Message is Encrypted');
```

The **Decryption** function works on the following algorithm-

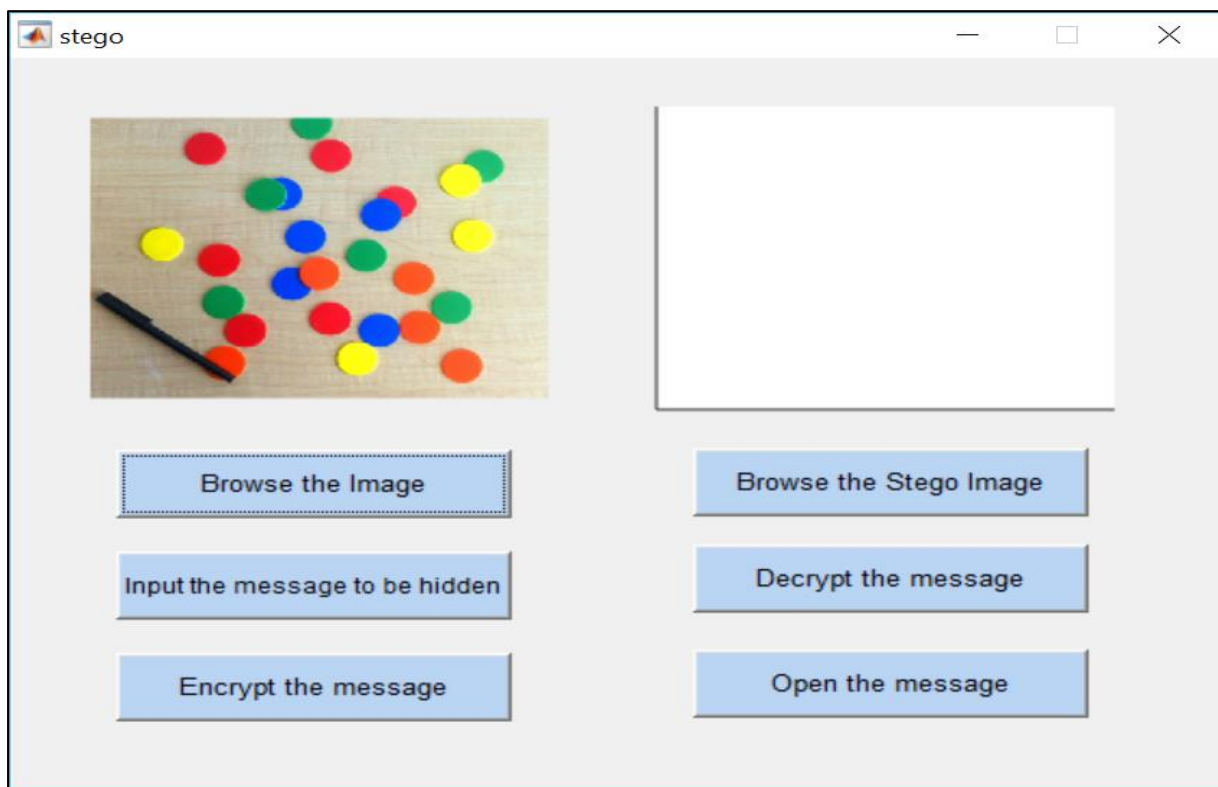
```
imag=handles.inputsteg;
ln=handles.sz;
h1 = size(imag,1);
w1 = size(imag,2);
r=1;
for i = 1 : h1
    for j = 1 : w1
        for k=1:3
            if (r <= ln)
                b(r) = mod(double(imag(i,j,k)),2);
            end
            r=r+1;
        end
    end
end
a=ln/8;
bv=num2str(b);
bv(isspace(bv))=" ";
bf=bv(:);
t=[8 a];
bs=reshape(bf,t);
bt=transpose(bs);
bd=bin2dec(bt);
td=transpose(bd);
sh=char(td);
fi=fopen('hidden.txt','wt');
fprintf(fi,'%s',sh);
fclose(fi);
msgbox('Message is successfully Decrypted');
```

## SCREENSHOTS OF THE WORKFLOW

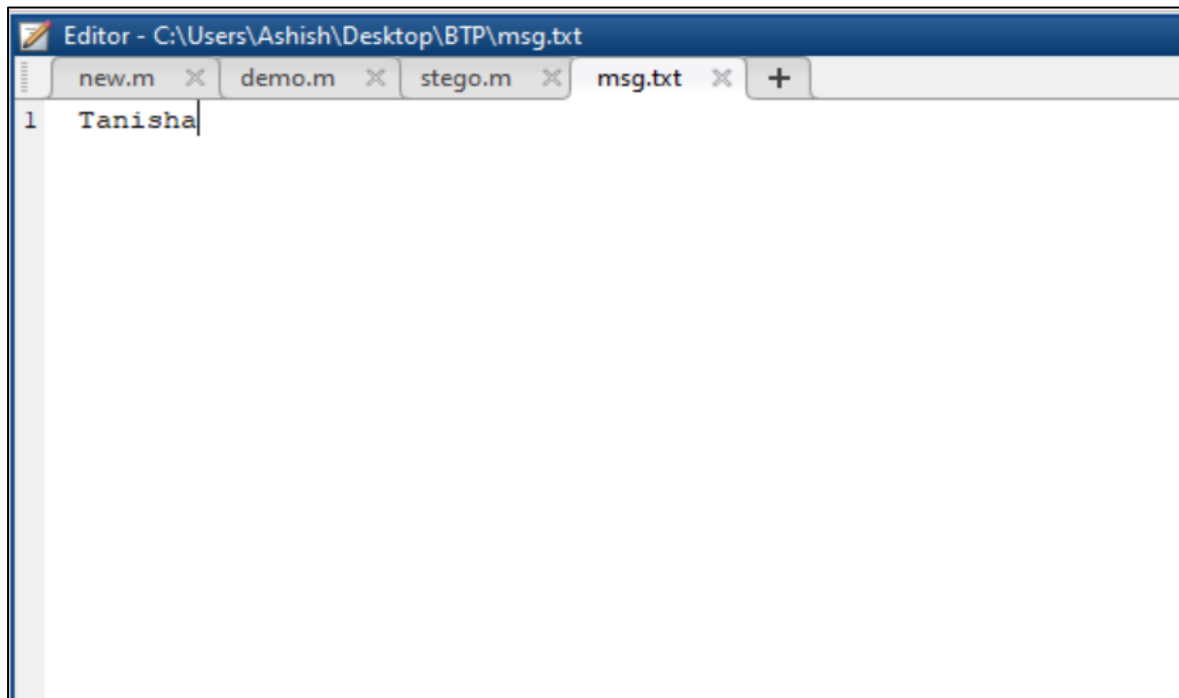
1. On clicking 'Browse the image' button, the following window appears to select the desired image from the working directory.



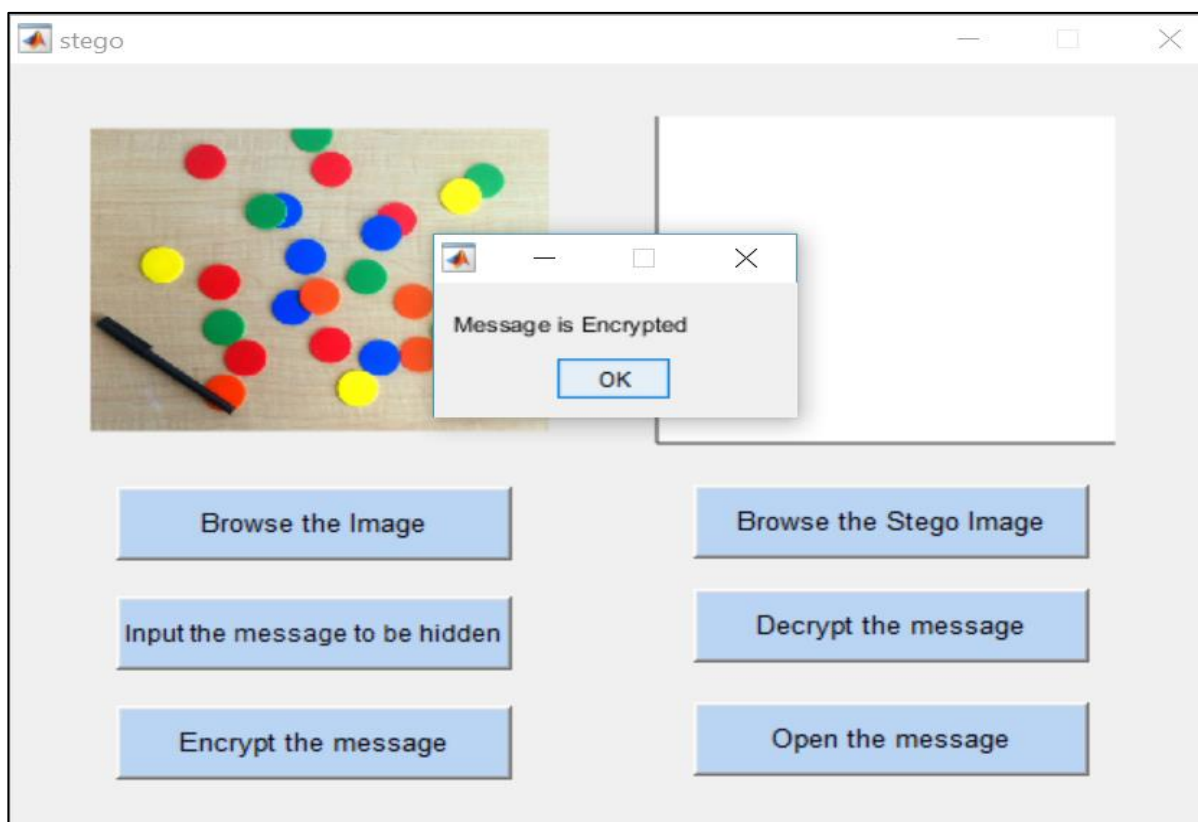
On selection of the image it is shown in the axes1 as-



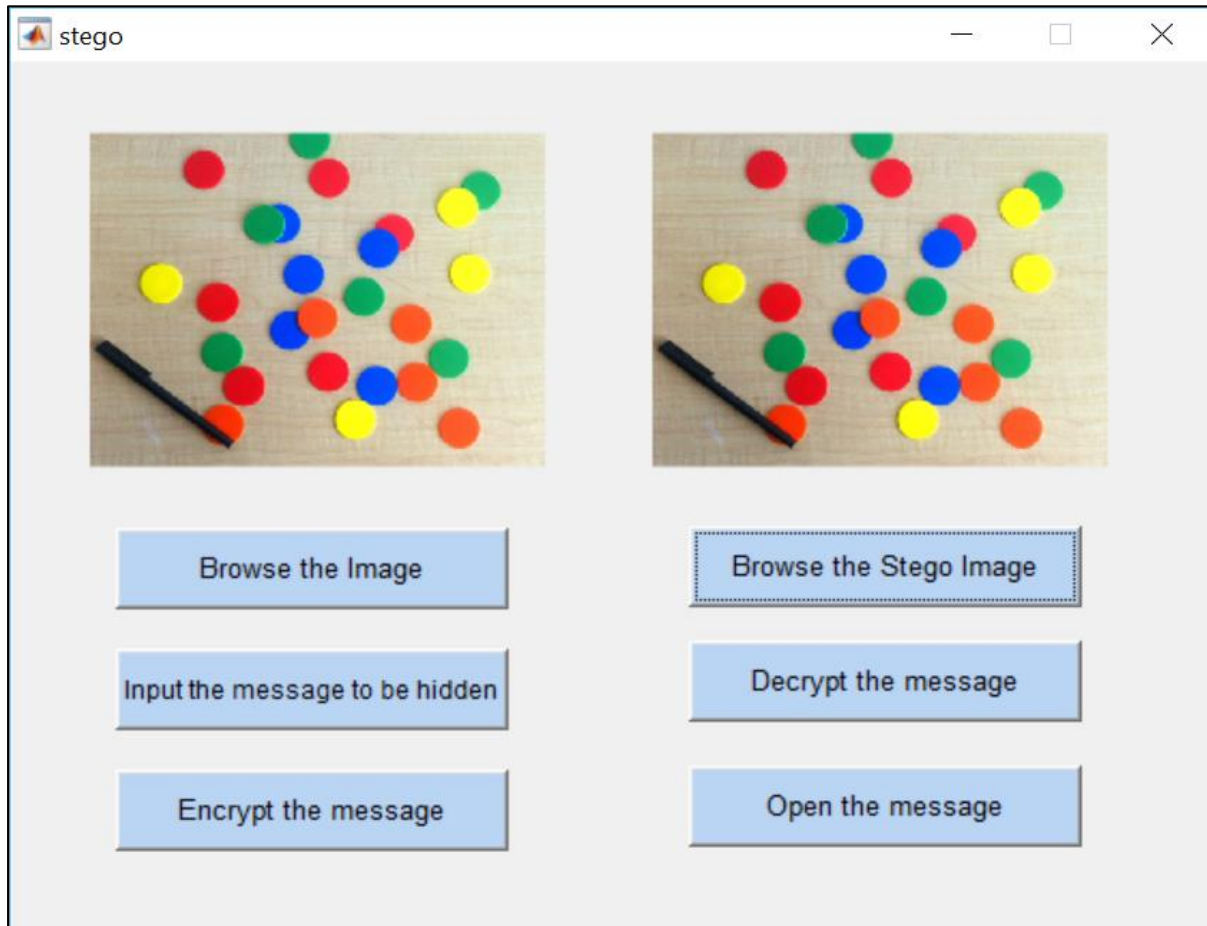
2. Input the message button opens the following text file, for the user to enter the message into it.



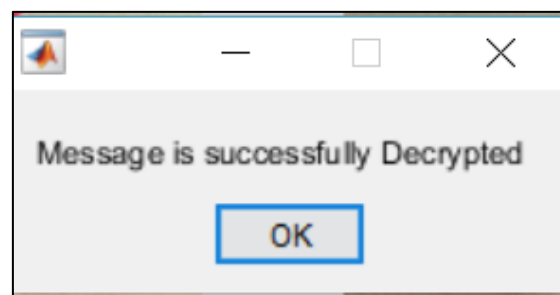
3. The Encrypt the message button runs the algorithm and shows a pop-up window as-



4. On the decryption part, when the stego image which has the hidden message, appears in the axes2. And it can be seen that these two images look the exactly the same.

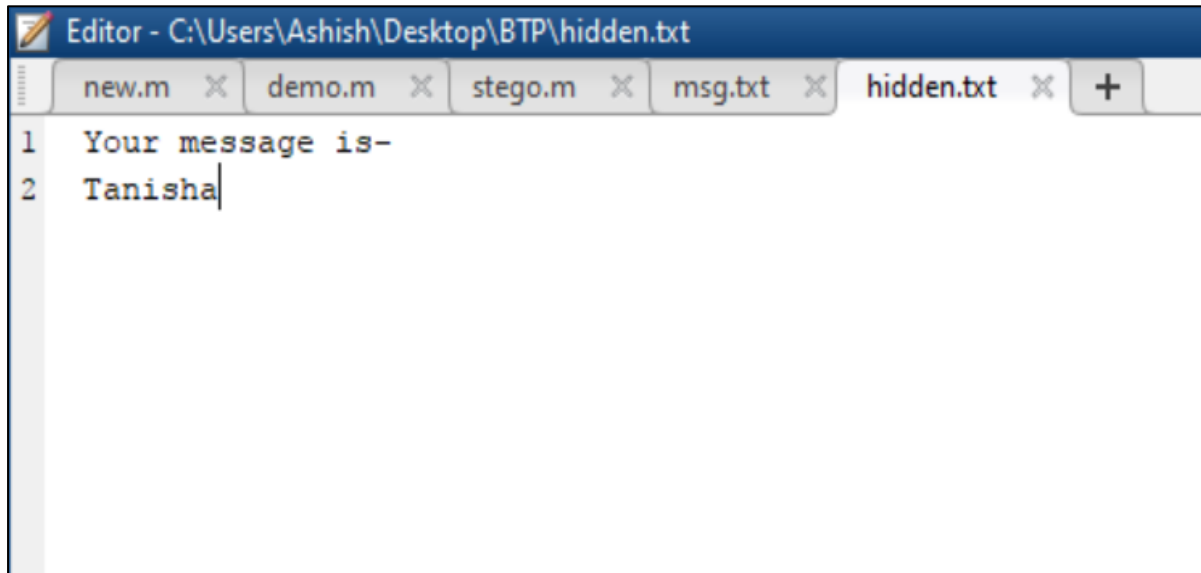


5. Decrypt the message button runs the decryption algorithm and shows a pop-up as-





6. Then finally a text file generated by the decryption algorithm, which has the extracted message is opened as-



The screenshot shows a text editor window titled "Editor - C:\Users\Ashish\Desktop\BTP\hidden.txt". The window has several tabs open: "new.m", "demo.m", "stego.m", "msg.txt", and "hidden.txt". The "hidden.txt" tab is active, and the text inside reads:

```
1 Your message is-  
2 Tanisha|
```

## **SUMMARY**

Steganography is a really interesting subject and outside of the mainstream cryptography and system administration that most of us deal with day after day.

Steganography can be used for hidden communication. We have explored the limits of steganography theory and practice. We printed out the enhancement of the image steganography system using LSB approach to provide a means of secure communication.

This steganography application software provided for the purpose to how to use any type of image formats to hiding any type of files inside them. The master work of this application is in supporting any type of pictures without need to convert to bitmap, and lower limitation on file size to hide, because of using maximum memory space in pictures to hide the file.

Since ancient times, man has found a desire in the ability to communicate covertly. The recent explosion of research in watermarking to protect intellectual property is evidence that steganography is not just limited to military or espionage applications. Steganography, like cryptography, will play an increasing role in the future of secure communication in the “digital world”.

## **REFERENCES**

### **Websites**

Following websites are referring to create this project reports.

- <http://www.google.com>
- <http://www.microsoft.com>
- <http://www.wikipedia.org>