A MINOR PROJECT REPORT

ON

IMAGE STEGANOGRAPHY WITH ENCRYPTION

SUBMITTED IN PARTIAL FULFILLMENT FOR THE AWARD OF DEGREE OF

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by:

Pranati Tiwari(9919102056)
Mridnal Garg(9919102032)
Raghav Joshi(9919102035)

Under the Guidance of

DR. KAPIL DEV TYAGI

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA (U.P.)

May, 2022

CERTIFICATE

This is to certify that the minor project report entitled, "**Image Steganography with Encryption**" submitted by **Pranati Tiwari, Mridnal Garg and Raghav Joshi** in partial fulfilment of the requirements for the award of Bachelor of Technology Degree in **Electronics and Communication Engineering** of the Jaypee Institute of Information Technology, Noida is an authentic work carried out by them under my supervision and guidance. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Signature of Supervisor:
Name of the Supervisor: Dr. Kapil Dev Tyagi
ECE Department,
JIIT, Sec-128,
Noida-201304

Dated: May 19, 2022

DECLARATION

We hereby declare that this written submission represents our own ideas in our own words and where others' ideas or words have been included, have been adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission.

Place: Noida
Date: May 19, 2022

Name: Pranati Tiwari
Enrollment: 9919102056

Name: Mridnal Garg
Enrollment: 9919102032

Name: Raghav Joshi
Enrollment: 9919102035

Table of Contents

ABSTRACT

The word steganography means" covered in hidden writing". The object of steganography is to send a message through some innocuous carrier (to a receiver while preventing anyone else from knowing that a message is being sent to all. Computer based steganography allows changes to be made to what are known as digital carriers such as images or sounds.

The changes represent the hidden message, but result if successful in non discernible change to the carrier. The information may be nothing to do with the carrier sound or image or it might be information about the carrier such as the author or a digital watermarking or fingerprint.

In steganography information can be hidden in carriers such as images, audio files, text files, and video and data transmissions. When message is hidden in the carrier a stego carrier is formed for example a stego-image. Hopefully it will be perceived to be as close as possible to the original carrier or cover image by the human senses.

Images are the most widespread carrier medium. The are used for steganography in the following way. The message may firstly be encrypted. They are used for steganography in the following way. The message may firstly be encrypted. The sender embeds the secret message to be sent into a graphic file.This results in the production of what is called stego-image. Additional secret data may be needed in the hiding process e.g. a stego key etc. This stego-image is then transmitted to the recipient.

The recipient extractor extracts the message from the carrier image. The message can only be extracted if there is a shared secret between the sender and the recipient. This could be the algorithm for extraction or a special parameter such as stego key. A stego analyst or attacker may try to intercept the stego image

Acknowledgement

We express our sincere regard and indebtedness to our project mentor DR.KAPIL DEV TYAGI for giving us an opportunity to work under his guidance. Like a true mentor, he motivated and inspired us through the entire duration of our work, without which this project could not have seen the light of the day.This project helped us in understanding the various parameters which are involved in the development of a fully functional client side web application.

Our appreciation also goes to the members of Project for their kind co-operation and encouragement which help us in completion of this project.

We convey our regards to all the other faculty members of Department of Electronics and Communication Engineering, JIIT NOIDA for their valuable guidance and advices at appropriate times.

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 What is Steganography?

Steganography derives from the Greek word steganos, meaning covered or secret, and graphy (writing or drawing). On the simplest level, steganography is hidden writing, whether it consists of invisible ink on paper or copyright information hidden in an audio file. Where cryptography scrambles a message into a code to obscure its meaning, steganography hides the message entirely.

These two secret communication technologies can be used separately or together—for example, by first encrypting a message, then hiding it in another file for transmission. As the world becomes more anxious about the use of any secret communication, and as regulations are created by governments to limit uses of encryption, steganography's role is gaining prominence. What Steganography essentially does is exploit human perception, human senses are not trained to look for files that have information hidden inside of them, although there are programs available that can do what is called Steganalysis (Detecting use of Steganography). Steganography hide the secret message within the host data set and presence imperceptible and is to be reliably communicated to a receiver. The host data set is purposely corrupted, but in a covert way, designed to be invisible to an information analysis.

1.2 Where Hidden Data Hides?

It is possible to alter graphic or sound files slightly without losing their overall viability for the viewer and listener. With audio, you can use bits of file that contain sound not audible to the human ear. With graphic images, you can remove redundant bits of color from the image and still produce a picture that looks intact to human eye and is difficult to discern from its original. It is in those bits that stego hides its data. A stego program uses an algorithm, to embed data in an image or sound file, and a password scheme to allow you to retrieve information. allow you to retrieve information.

1.3 Background of the Problem

"Steganography can be traced back to the ancient era, where someone would peel wax off a tablet that was covered in wax, write a message underneath the wax then re-apply the wax. The recipient of the message would simply remove the wax from the tablet to view the message".

"During World War II invisible ink was used to write information on pieces of paper so that the paper appeared to the average person as just being blank pieces of paper. Previously Secret Data consists of linguistic or language forms of hidden writing. The later, such as invisible ink, try of hide messages physically. One disadvantage of linguistic steganography is that users must equip themselves to have a good knowledge of linguistry. In recent years, everything is trending toward digitization. And with the development of the internet technology, digital media can be transmitted conveniently over the network.

Therefore, messages can be secretly carried by digital media by using the steganography techniques, and then be transmitted through the internet rapidly 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.

1.4 Problem Statement

"The internet is one of the most powerful modern tools of information and communication technology and the underlying issue has always been the security of information. Unfortunately, it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique that will be used to implement this is called steganography", (Petitcolas et al., 1999).

"Many governments have created laws to either limit the strength of cryptographic systems or to prohibit it altogether, forcing people to study alternative methods of securing information transfer", (Dunbar, 2002). "Business has also started to realise the potentials of steganography in communicating trade secrets or new product information"

1.5 Study Assumptions

The sender and receiver must have shared some secret information before exchanging any hidden data. Steganography assumes that prior information is shared by two communicating parties. Pure steganography however requires no prior information is shared by two communicating parties.

CHAPTER 2

LITERATURE SURVEY

The aim of this chapter is to perform a comprehensive survey on the Image Steganography and various techniques to perform to find and discover an effective technique in implementing the technique to perform steganography and to summarise the various results from various research and conference papers. Surely this would result in covering a large number of researches that are hard to put in the same canvas.

This chapter is a source of such information for researchers in order for them to be precisely correct on result comparison before publishing new achievements in this field. In this literature survey three digital libraries are chosen due to limited resources and the huge number of articles under this topic. However, this can be clearly seen that these libraries cover a significant amount of the related literature sources for our study.

Three different digital libraries were used to execute a research:

1. IEEE Xplore
2. Research Gate
3. Scihub

2.1 Information Hiding in Images Using Steganography Techniques [1]

Steganography is a technique that prevents unauthorised users to have access to the important data. Many techniques are available to prevent unauthorised users from copying information without owner permission. Two of these techniques are cryptography and steganography. Cryptography is a rule or protocol between transmitter and receiver using some encryption keys to understand each other. Those encryption keys can be private or public. Unauthorised users can see the coded information without understanding or being able to read it. The second method is steganography, which is embedded information which does not appear to users. Steganography is managing a secret path for sending information invisibly. Digital watermarking is one of the popular applications for steganography. Users can hide important information within an image by using an invisible watermark when they transmit data. Moreover, a visible watermark can be used in many applications such as author, creator, and document. Images have some unimportant regions the human visual system cannot recognise by replacing these regions with other information. A user can change the least significant bit in each pixel with his/her own information without the quality of an image being decreased. Also, this alteration does not affect the intensity of the colour.

2.2 An image steganography approach based on k-least significant bits (k-LSB) [2]

Image steganography is the operation of hiding a message into a cover image. the message can be text, codes, or image. Hiding an image into another is the proposed approach in this paper. Based on LSB coding, a k-LSB-based method is proposed using k least bits to hide the image. For decoding the hidden image, a region detection operation is used to know the blocks contains the hidden image. The resolution of stego image can be affected, for that, an image quality enhancement method is used to enhance the image resolution. The effectiveness of the proposed method for hiding an image into another is evaluated by the metric peak signal-to-noise ratio (PSNR). In steganography, when the secret data is embedded, PSNR measures the ratio of noise between the stego image and the original one. Based on the PSNR value, the image quality will be better when its value is higher.

2.3 An improved LSB based image steganography technique for RGB images. [3]

There are number of steganography techniques proposed to hide data like LSB, DCT, pixel-value differencing, DFT etc. into images with precision level. But these techniques suffering from some problems like less hiding capacity, degrade the quality of image and security of hidden data after hiding more data into it. To overcome these problems this paper proposed an improved LSB technique for color images by embedding the information into three planes of RGB image in a way that enhances the quality of image and achieves high embedding capacity. The PSNR value of the proposed technique is better than previous steganography methods. The proposed method is made on the basis of sensitivity of human eyes to various color wavelengths. This selective approach induces lower noise and high security for transferring images.The LSB approach replaces the least significant Bit of the pixel in the cover image. The earlier approaches used to hide the secret message for colored images leads to high noise in the stego-image due to this the secret information is susceptible to be detected. But the proposed method results in better image quality, secure and reliable as it has sliced the image into three planes i.e. Red, green and blue and then insert the message in each plane on the basis of color sensitivity.

2.4 A new approach for LSB based image steganography using secret key. [4]

This paper introduces a best approach for LSB based on image steganography that enhances the existing LSB substitution techniques to improve the security level of hidden information. It is a new approach to substitute LSB of RGB true color image. The new security conception hides secret information within the LSB of image where a secret key encrypts the hidden information to protect it from unauthorised users. In general, in LSB methods, hidden information is stored into a specific position of LSB of image. For this reason, knowing the retrieval methods, anyone can extract the hidden information.Thus the hidden information is stored into different position of LSB of image depending on the secret key. As a result, it is difficult to extract the hidden information knowing the retrieval methods.

2.5 A Robust RGB Channel Based Image Steganography Technique using a Secret Key [5]

The paper proposes a RGB channel based steganographic technique for images imparting better information security. This technique inserts the information into deeper layers of the selected RGB channel and the position is determined depending on the status of channel and value of the secret key. Pixels of the cover image are selected depending on the environment of the channels and hidden information. The ambiguity of pixel, channel and position selection process increases robustness of the steganographic system. The technique is also less vulnerable to unintentional attacks like image manipulation as data hides in the deeper layer of the pixels. The system uses the RGB channels of the stego-image and the secret key to extract the hidden information.In this proposed technique, RGB channel of the cover image (JPEG) is used for carrying data. It utilises the secret key, channel environment and secret message to hide information in one of the three channels. It hides secret data into variable position of deeper layer of the channels and changes the LSBs of Blue and Green components accordingly that causes minimum distortions in the stego-image.

2.6 A detailed look at steganographic techniques and the use in an open-system environment[6]

Coding secret messages in digital images is by far the most widely used of all methods in the digital world of today. This is because it can take advantage of the limited power of the human visual system. Almost any plain text, cipher text, image and any other media that can be encoded into a bit stream can be hidden in a digital image.Least significant bit encoding is by far the most popular of the coding techniques used for digital images. By using the LSB of each byte (8 bits) in an image for a secret message, you can store 3 bits of data in each pixel for 24-bit images and 1 bit in each pixel for 8-bit images. As you can see, much more information can be stored in a 24-bit image file. Depending on the color palette used for the cover image (i.e., all gray), it is possible to take 2 LSB's from one byte without the HVS being able to tell the difference. The only problem with this technique is that it is very vulnerable to attacks such as image changes and formatting (i.e., changing from .GIF to .JPEG). Masking and filtering techniques for digital image encoding such as Digital Watermarking (i.e.- integrating a companies logo on there web content) are more popular with lossy compression techniques such as (.JPEG). This technique actually extends an images data by masking the secret data over the original data as opposed to hiding information inside of the data. The beauty of Masking and Filtering techniques are that they are immune to image manipulation which makes there possible uses very robust.

2.7 Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research [7]

Among various media types used, the popularity and availability of digital images are high and in this research work and hence, our focus is on implementing digital image steganography. The main challenge in designing a steganographic system is to maintain a fair trade-off between robustness, security, imperceptibility and higher bit embedding rate. Without a doubt, data security is the soul of data communication. Generally, information security systems are separated into two major categories, one is encryption and another is information hiding. Both of the categories are responsible for securing the information but their techniques differ. Steganography does not change the format of data or message and keeps the presence of its actual data, whereas cryptography keeps the data secret by converting it into an unreadable form. Cryptographic method's weakness lies in the presence of original data, even if the original data has been encrypted. Hence steganography techniques are an additive security to cryptographic techniques. With the combination of both, it provides an additional layer of security for the message during data communication. Whenever data security is an issue, Watermarking comes to light. It is one of the popular data security techniques that use authentication and copyright method to serve its purpose. The process of embedding digital data in the multimedia data is called a watermark. Watermarking is also pertinent to steganography in a broader sense, as both are intending to hide the data in media data. Both of the techniques possess properties such as security, capacity, robustness, and imperceptibility, but they prioritize them differently.

2.8 Literature Survey On Modern Image Steganographic Techniques [8]

Based on certain design criteria's such as security against RS steganalysis, invisibility, payload capacity, robustness against attacks, embedding scheme etc, different algorithms have been evaluated. This analysis explores the strengths and weaknesses of the modern image steganographic techniques which will enable us to design a better steganographic algorithm. Palette based images, such as GIF images, are popular image file format commonly used on the Internet. GIF images are indexed images where the colours used in the image are stored in a palette or a colour lookup table. The main issue with the palette based approach is that if one changes the least significant bit of a pixel, it could result in an entirely different colour since the index to the colour palette gets modified. One possible solution to this problem is to sort the palette so that the colour differences between consecutive colours are minimised. The strong and weak points regarding embedding information in GIF images using LSB is that since GIF images only had a bit depth of 8, the total amount of information that could be embedded will be less. Hiding Secret Message in Edges of the image introduced a new least significant bit embedding algorithm for hiding secret messages in non-adjacent pixel locations at the edges of images. Here the messages were hidden in regions which were least like their neighbouring pixels i.e.

regions that contain edges, corners, thin lines etc so that an attacker will have less suspicion of the presence of message bits in edges, because pixels in edges of an image appears to be much brighter or dimmer than their neighbours. One common disadvantage of LSB embedding was that it created an imbalance between the neighbouring pixels and the embedding capacity was relatively low. Here a new adaptive least-significant bit (LSB) steganographic method based on pixel-value differencing (PVD) [8] was proposed. The difference value of two consecutive pixels [9] estimates how many secret bits to be embedded into the two pixels. Pixels located in the edge areas were embedded with more secret bits than that located in smooth areas.

	LSB in GIF	RE-LSB	AE-LSB
Invisibility	Low	High	High
Payload Capacity	High	Low	High
Robustness against statistical attacks	High	Medium	High
Tolerance to RS Steganalysis	Low	High	Low
Robustness against image manipulation	Low	Medium	High
LSB replacement style asymmetry	Yes	Yes	Yes
Utilisation of edge areas	Low	Medium	Low

Table 2.1 Comparison of LSB Steganography Techniques

CHAPTER 3

METHODOLOGY

3.1 Methodology To Be Used

A methodology is defined as 'a system of practices, techniques, procedures, and rules used by those who work in a discipline.

In this Project we have used “Waterfall” methodology. The waterfall model is more suitable when the requirements are well known, product definition is stable, technology is understood, there are no ambiguous requirements, adequate resources with required expertise are available freely and the project is short.

The Waterfall model defines several consecutive phases that must be completed one after the other and moving to the next phase only when its preceding phase is completely done.

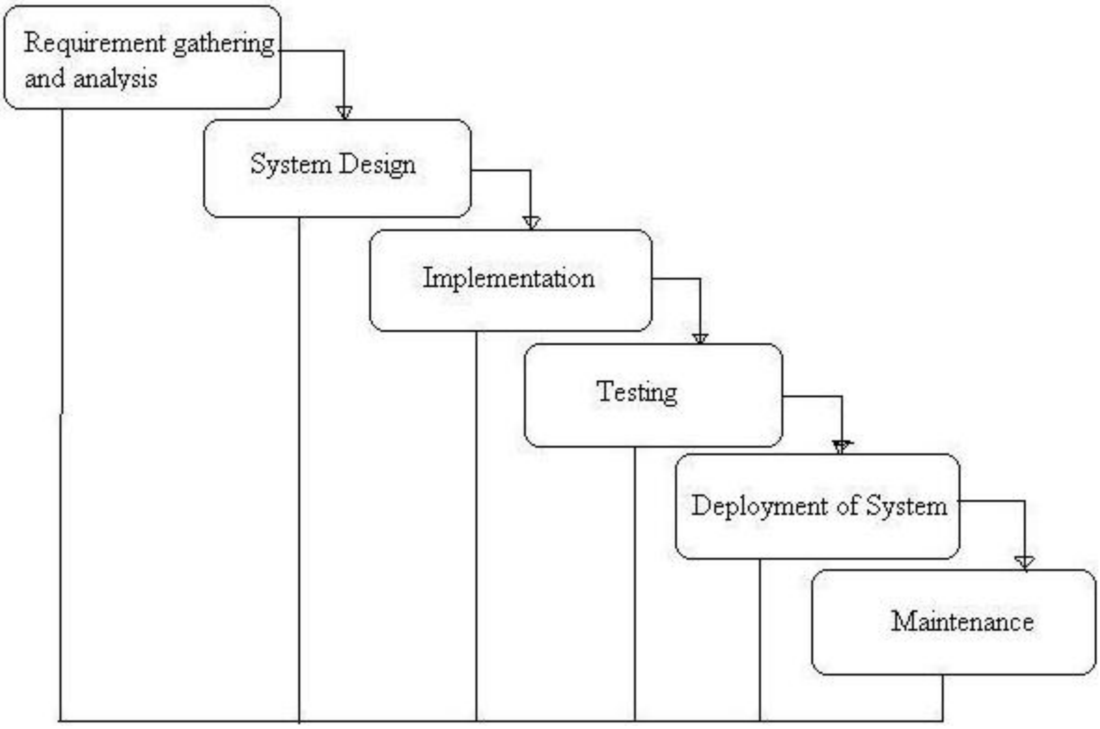


Fig. 3.1 Steps involved in waterfall model

3.2 Hardware Requirements

- GPU
- RAM : Minimum of 4GB
- Processor : Minimum Dual Core Processor

3.3 Software Requirements

- VISUAL STUDIO Code
- Google Collab
- Python IDE
- Framework used: Tkinter

3.4 Functional Requirement

The Functional Requirement document (also called Functional Specifications or Functional Requirement Specifications), defines the capabilities and functions that system must be able to perform successfully. Sone of them are:

- The system will allow an effective and reliable way of security for information through obscurity.
- The system will not allow the user's to modify encrypted image in any way.
- The system will allow for the use of any image file and information file to be used respectively for encryption.
- User's should have a basically fair knowledge in computer file types and computing in order to effectively make use of the system

3.5 Non - Functional Requirement

A Non-Functional Requirement is usually some form of constraint or restriction that must be considered when designing the solution. For the most part when people are talking about Constraints, they are referring to Non-Functional Requirements. Non-Functional Requirements have the same following characteristics:

- For user interfaces we take in consideration that they should has a standard look and being user friendly at the same time to make sure that users' attention will not be distracted and interface to provide more flexibility and scalability.
- The program will be in the English language
- The program must be fast in processing
- The program must to hide the image within the image and then extract image from the image properly.
- All function must be works well then system will be a high quality.

3.6 Design of the Steganography System

The tool is designed with two modules to serve the function of encrypting and decrypting.

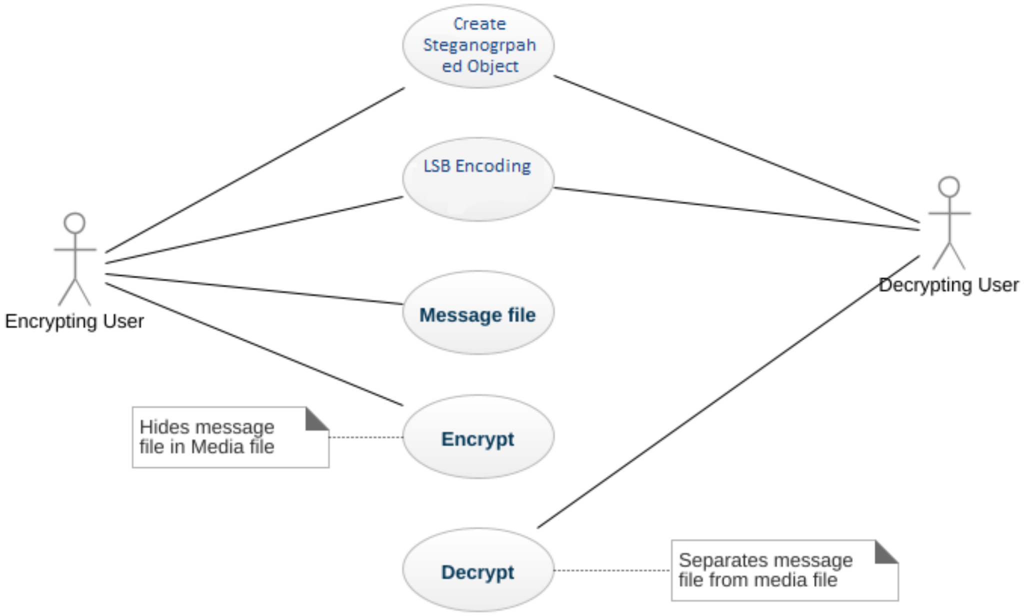

Create Steganogrpahed Object
LSB Encoding
Message file
Encrypt
Decrypt
Encrypting User
Decrypting User
Hides message file in Media file
Separates message file from media file

Fig. 3.2 Use Case Diagram

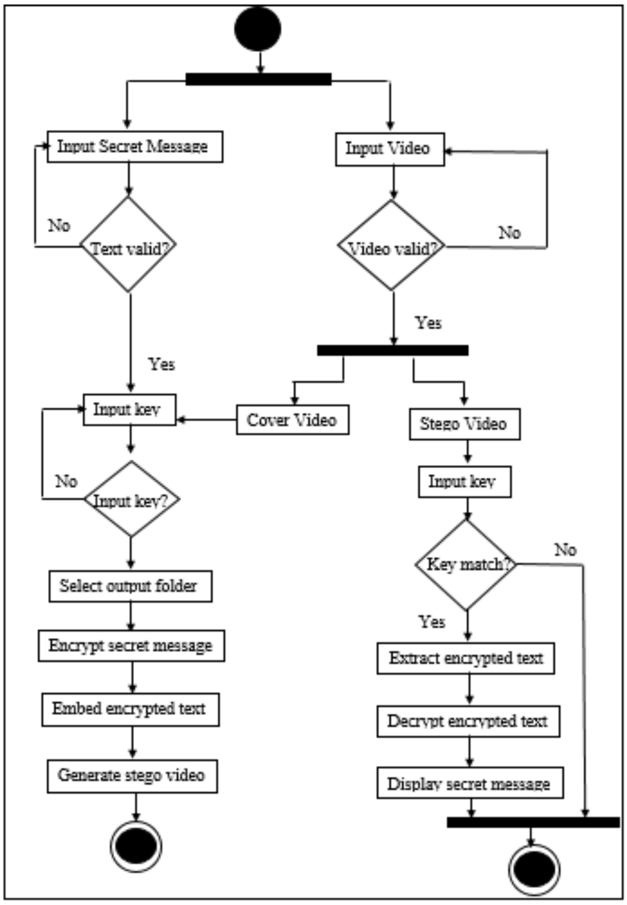


FIG 3.3 Activity Diagram

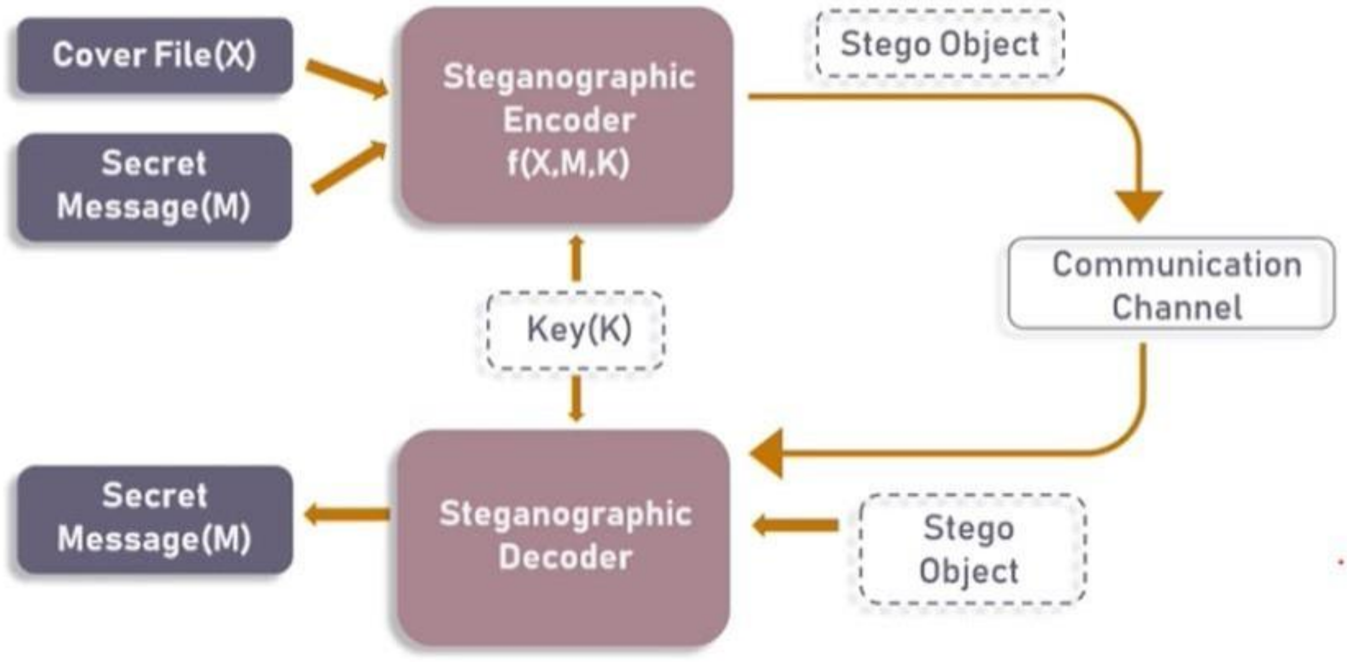


Fig 3.4 Block Diagram

3.6.1 The Encoding Process:

- The steganography technique used is LSB coding.
- The offset of the image is retrieved from its header.
- That offset is left as it is to preserve the integrity of the header, and from the next byte, we start our encoding process.
- For encoding, we first take the input carrier file i.e. an image file and then direct the user to the selection of the text file.

CHAPTER 4
IMPLEMENTATION

Structure of image files is that an image is created from pixels that any pixel created from three colors (red, green and blue said RGB) each color of a pixel is one byte information that shows the density of that color. Merging these three color makes every color that we see in these pictures. We know that every byte in computer science is created from 8 bit that first bit is Most-Significant-Bit (MSB) and last bit Least-Significant-Bit (LSB), the idea of using Steganography science is in this place; we use LSB bit for writing our security information inside pictures. So if we just use last layer (8th layer) of information, we should change the last bit of pixels, in other hands we have 3 bits in each pixel so we have 3*height*width bits memory to write our information. But before writing our data we must write name of data(file), size of name of data & this by assigning some first bits of memory (8th layer).

4.1 Code Snippet

4.1.1 Codes for Encryption and Decryption

```
import cv2
import numpy as np
import types
from google.colab.patches import cv2_imshow

#Here we are using 08b as we require 8 bit representation of binary digits.
#If we will be using only b then it will not add 0 to convert it into 8 bits
# and returns the binary converted value

def messageToBinary(message):
  if type(message) == str:
    return ''.join([ format(ord(i), "08b") for i in message ])

  elif type(message) == bytes or type(message) == np.ndarray:
    return [ format(i, "08b") for i in message ]
```

```
elif type(message) == int or type(message) == np.uint8:
    return format(message, "08b")
  else:
    raise TypeError("Input type not supported")

# Function to hide the secret message into the image

def Hide(image, secret_message):

  # calculate the maximum bytes to encode
  n_bytes = image.shape[0] * image.shape[1] * 3 // 8
  print("Maximum bytes to encode:", n_bytes)

  #Check if the number of bytes to encode is less than the maximum bytes in the image
  if len(secret_message) > n_bytes:
      raise ValueError("Error encountered insufficient bytes, need bigger image or less data !!")

  secret_message += "#####" # you can use any string as the delimeter

  data_index = 0
  # convert input data to binary format using messageToBinary() fucntion
  binary_secret_msg = messageToBinary(secret_message)

  data_len = len(binary_secret_msg) #Find the length of data that needs to be hidden
  for values in image:
      for pixel in values:
          # convert RGB values to binary format
          r, g, b = messageToBinary(pixel)
          # modify the least significant bit only if there is still data to store
          if data_index < data_len:
              # hide the data into least significant bit of red pixel
              pixel[0] = int(r[:-1] + binary_secret_msg[data_index], 2)
              data_index += 1
```

```
            if data_index < data_len:
                # hide the data into least significant bit of green pixel
                pixel[1] = int(g[:-1] + binary_secret_msg[data_index], 2)
                data_index += 1
            if data_index < data_len:
                # hide the data into least significant bit of  blue pixel
                pixel[2] = int(b[:-1] + binary_secret_msg[data_index], 2)
                data_index += 1
            # if data is encoded, just break out of the loop
            if data_index >= data_len:
                break
    return image

def showData(image):

    binary_data = ""
    for values in image:
        for pixel in values:
            r, g, b = messageToBinary(pixel)
            #extracting data from the least significant bits of red, green and blue
            binary_data += r[-1]
            binary_data += g[-1]
            binary_data += b[-1]
    # split by 8-bits
    all_bytes = [ binary_data[i: i+8] for i in range(0, len(binary_data), 8) ]
    # convert from bits to characters
    decoded_data = ""
    for byte in all_bytes:
        decoded_data += chr(int(byte, 2))
        if decoded_data[-5:] == "#####": #check if we have reached the delimiter which is "#####"
            break
    return decoded_data[:-5] #remove the delimiter to show the original hidden message
```

```
# Function to Encode data
def encode():
  imgfile1 = input("Enter the name of image(with extension): ")
  image = cv2.imread(imgfile1)

  print("\n\n~Original Image~\n")
  resized_image = cv2.resize(image, (500, 500))
  cv2_imshow(resized_image)

  data = input("Enter data to be encoded : ")

  if (len(data) == 0):
    print("Data is empty")

  imgfile2 = input("\nEnter the name of new encoded image(with extension): ")

# Function to Decode data
def decode():
  image_name = input("Enter the name of the image to be decode (with extension): ")
  image = cv2.imread(image_name)

  print("\n\n~ Encoded Image ~ \n")
  resized_image = cv2.resize(image, (500, 500))
  cv2_imshow(resized_image)
  print("\n\n--------------------------------------------")
  print("\tDecoded Message: " + showData(image))
  print("--------------------------------------------")

def steganography():
    a = 0
    while a!=3:
```

```
    a = input("\n\nImage Steganography \n 1. Encode the data \n 2. Decode the data \n 3. Exit \n\n Your input is: ")
    choice = int(a)
    if (choice == 1):
      print("\nEncoding....")
      encode()

    elif (choice == 2):
      print("\nDecoding....")
      decode()
    elif (choice == 3):
      print("\nExiting...")
      return
    else:
      print("Please enter correct input...\n")

#steganography()
```

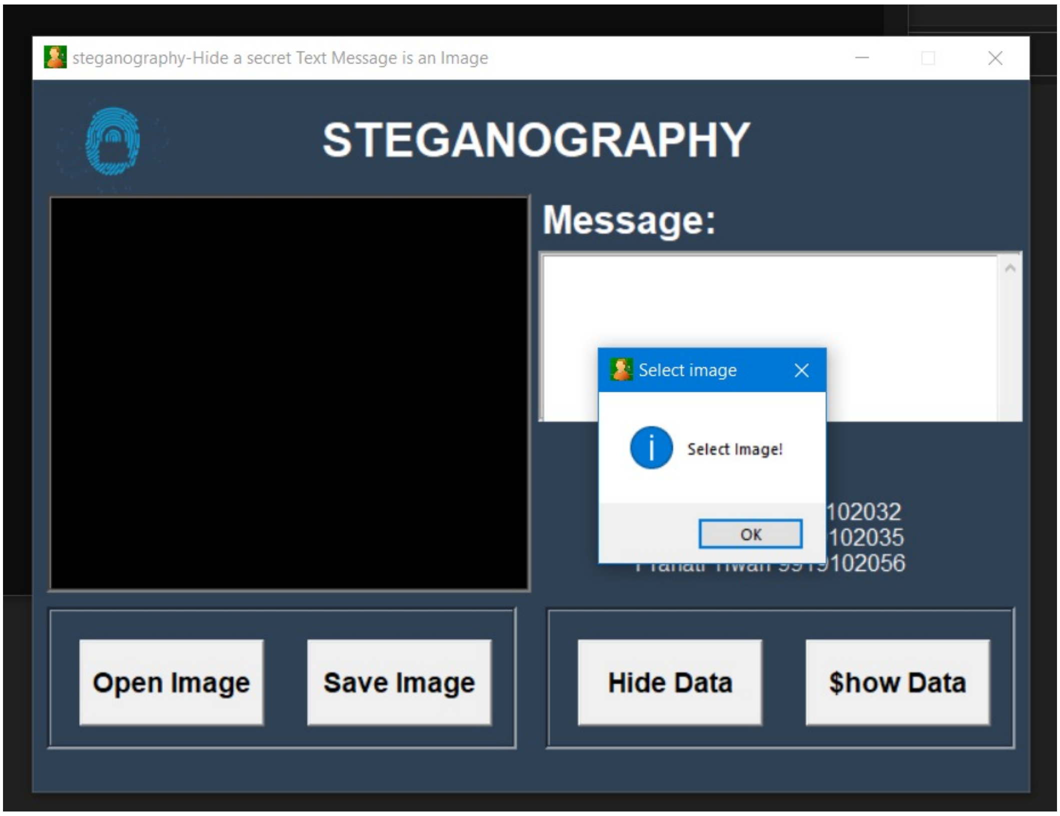
4.2.2 Code for GUI

```
from asyncio.windows_events import NULL
from tkinter import *
from tkinter import filedialog
import tkinter as tk
from PIL import Image, ImageTk
import os
from stegano import lsb
import ctypes
import tkinter.messagebox
root=Tk()
root.title("steganography-Hide a secret Text Message is an Image")
root.geometry("700x500+150+180")
root.resizable(False,False)
root.configure(bg="#2f4155")
```

```
def showimage():
    global filename
    filename=filedialog.askopenfilename(initialdir=os.getcwd(),
                                      title='Select Image File',
                                      filetype=(("PNG file","*.png"),
                                            ("JPG File",".jpg"),("All file",".txt")))
    img=Image.open(filename)
    img=ImageTk.PhotoImage(img)
    lbl.configure(image=img,width=250,height=250)
    lbl.image=img

def Hide():
    global secret
    message=text1.get(1.0,END)
    secret=lsb.hide(str(filename),message)

def Show():
    clear_message=lsb.reveal(filename)
    text1.delete(1.0,END)
    text1.insert(END, clear_message)

def save():
    secret.save("hidden.png")

#icon on popups
image_icon=PhotoImage(file="logo.jpg")
root.iconphoto(False,image_icon)

#logo
logo=PhotoImage(file="logo.png")
Label(root,image=logo, bg="#2f4155").place(x=10,y=0)
Label(root,text="STEGANOGRAPHY",bg="#2f4155",fg="white",font="arial 25 bold").place(x=200,y=20)

#Image Frame
f=Frame(root,bd=3,bg="black",width=340,height=280,relief=GROOVE)
f.place(x=10,y=80)

lbl=Label(f,bg="black")
lbl.place(x=40,y=10)

#Message Frame
frame2=Frame(root,bd=3,width=340,height=120,bg="white",relief=GROOVE)
frame2.place(x=355,y=120)

Label(root,text="Message:",bg="#2f4155",fg="white",font="arial 20 bold").place(x=355,y=80)
text1=Text(frame2,font="Robote 16",bg="white",fg="black",relief=GROOVE,wrap=WORD)
text1.place(x=0,y=0,width=320,height=295)

scrollbar1=Scrollbar(frame2)
scrollbar1.place(x=320,y=0,height=300)

scrollbar1.configure(command=text1.yview)
text1.configure(yscrollcommand=scrollbar1.set)

#third Frame
frame3=Frame(root,bd=3,bg="#2f4155",width=330,height=100,relief=GROOVE)
frame3.place(x=10,y=370)

def onClick1():
   tkinter.messagebox.showinfo("Select image",  "Select Image!")
button  =Button(frame3,text="Open  Image",width=10,height=2,font="arial  14  bold",command=lambda:
[onClick1(),showimage()]).place(x=20,y=20)

def onClick2():
    tkinter.messagebox.showinfo("Save", "Image Saved!")

button =Button(frame3,text="Save Image",width=10,height=2,font="arial 14 bold",command=lambda:
[onClick2(),save()]).place(x=180,y=20)

#Fourth Frame
frame4=Frame(root,bd=3,bg="#2f4155",width=330,height=100,relief=GROOVE)
frame4.place(x=360,y=370)

def onClick3():
    tkinter.messagebox.showinfo("Hide", "Message successfully hidden !")
button = Button(frame4,text="Hide Data",width=10,height=2,font="arial 14 bold",command=lambda:
[onClick3(),Hide()]).place(x=20,y=20)
Button(frame4,text="$how Data",width=10,height=2,font="arial 14 bold",command=lambda:
[Show()]).place(x=180,y=20)

Label(root,text="Mady By:\n\nMridnal Garg 9919102032\nRaghav Joshi 9919102035\nPranati Tiwari
9919102056",bg="#2f4155",fg="white",font="arial 12").place(x=420,y=255)
root.mainloop()
```

CHAPTER 5

SIMULATION RESULTS AND ANALYSIS



Select Image!

Fig. 5.1 View of how GUI looks initially

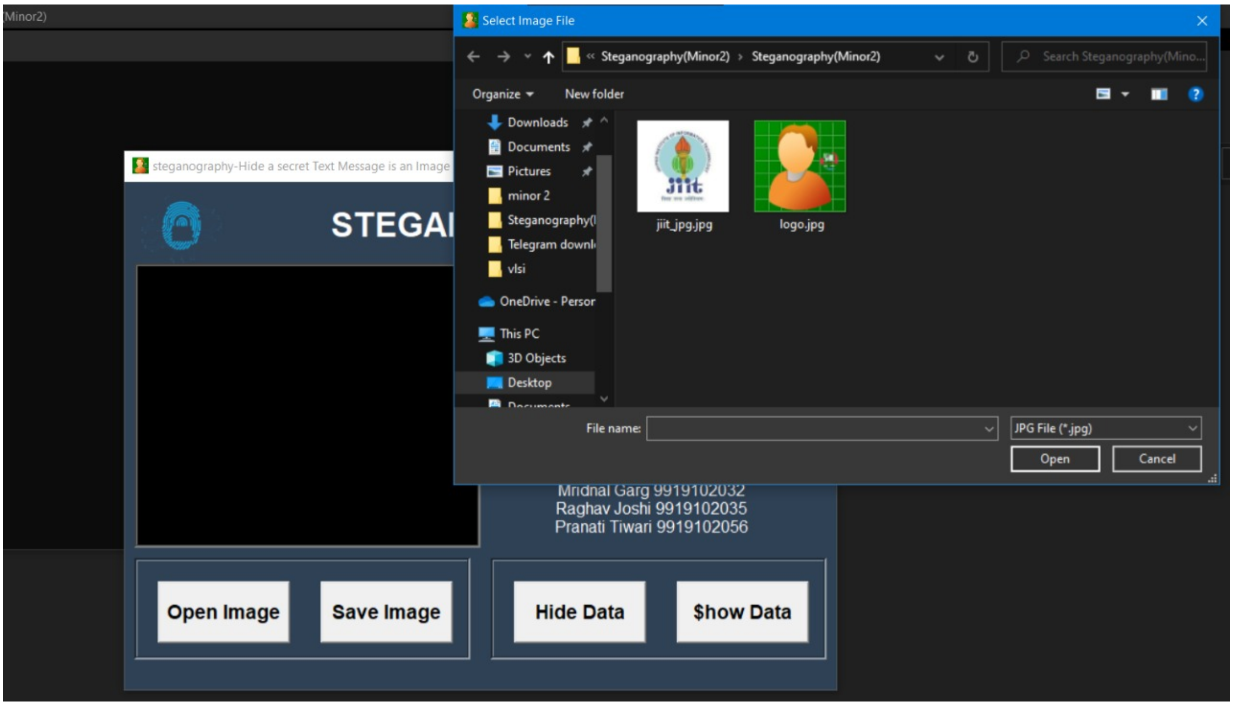
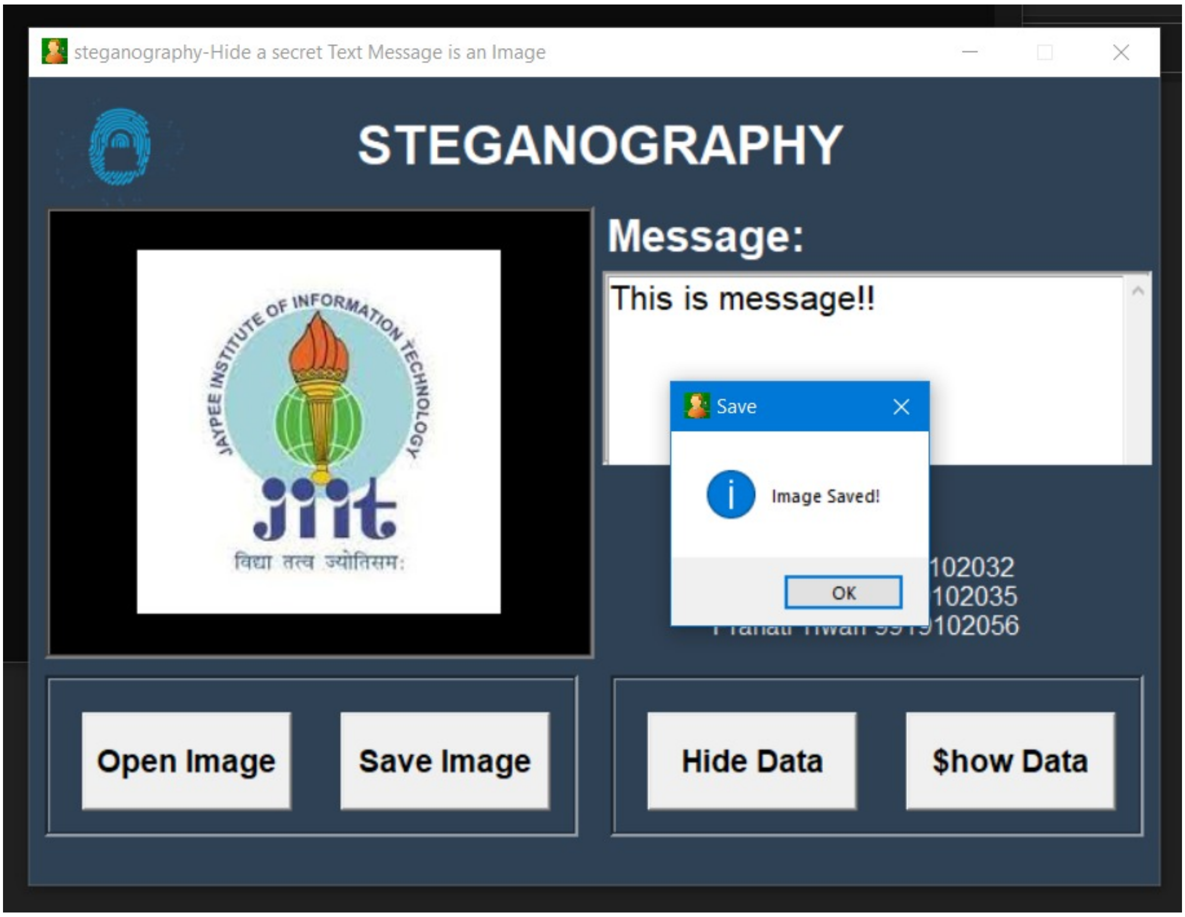


Fig. 5.2 Choosing an image for steganography

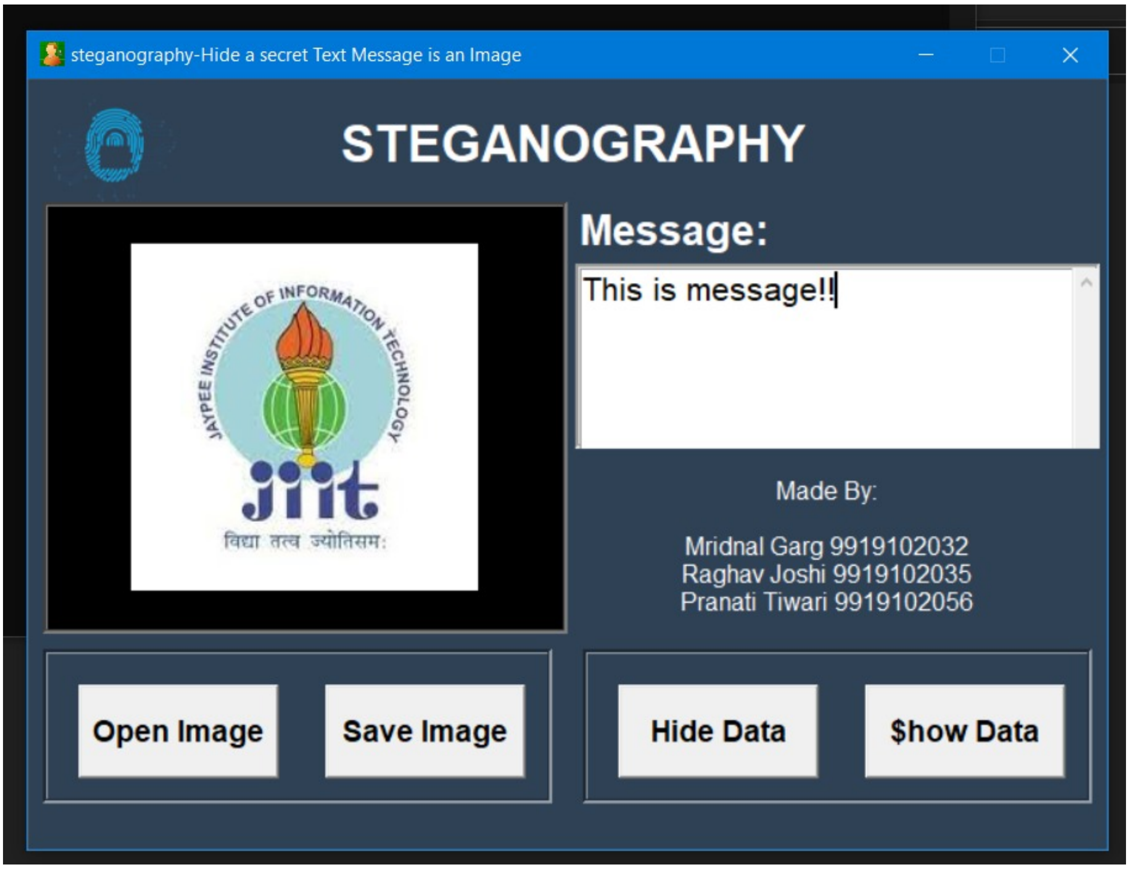


This is message!!

Image Saved!

102032
102035
Pranati Tiwari 9919102056

Fig. 5.3 Image ready for message encryption

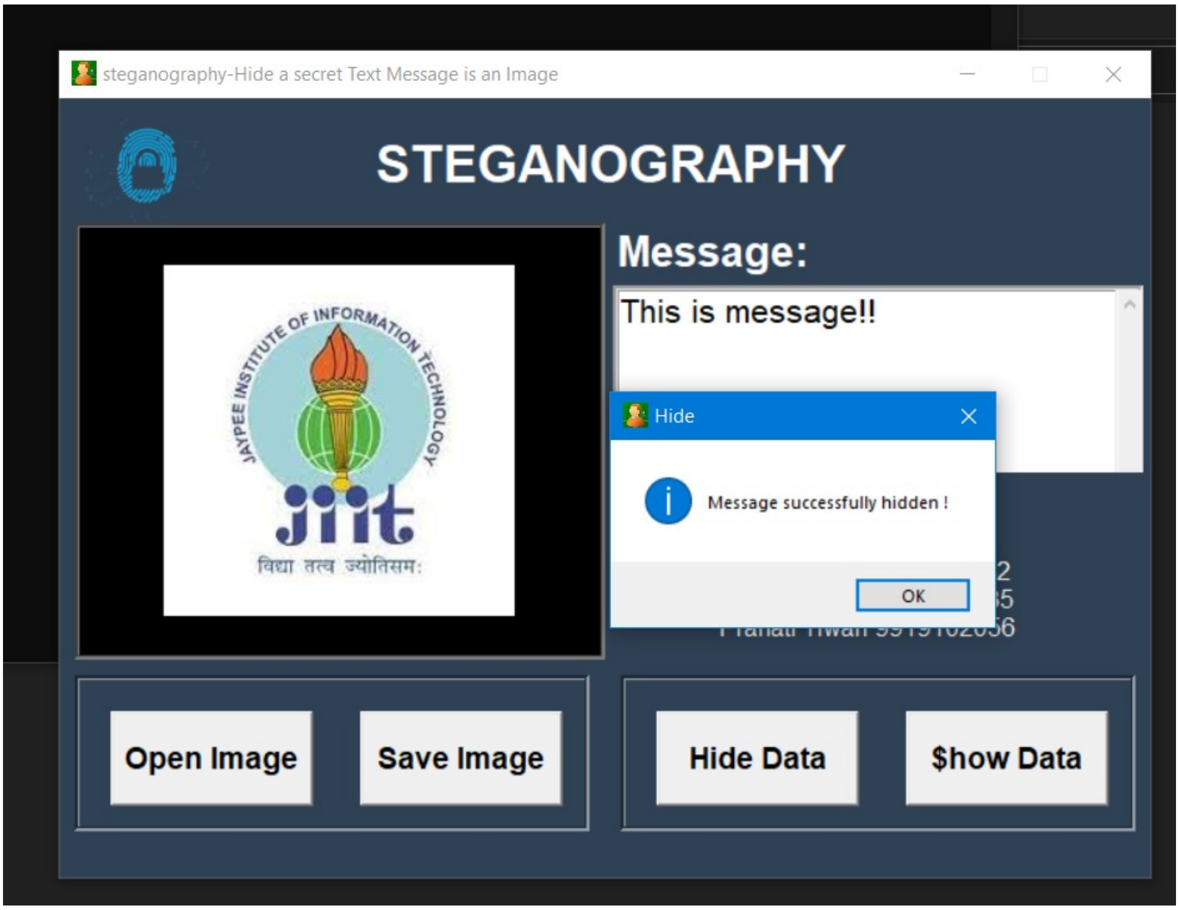


This is message!!

Made By:

Mridnal Garg 9919102032
Raghav Joshi 9919102035
Pranati Tiwari 9919102056

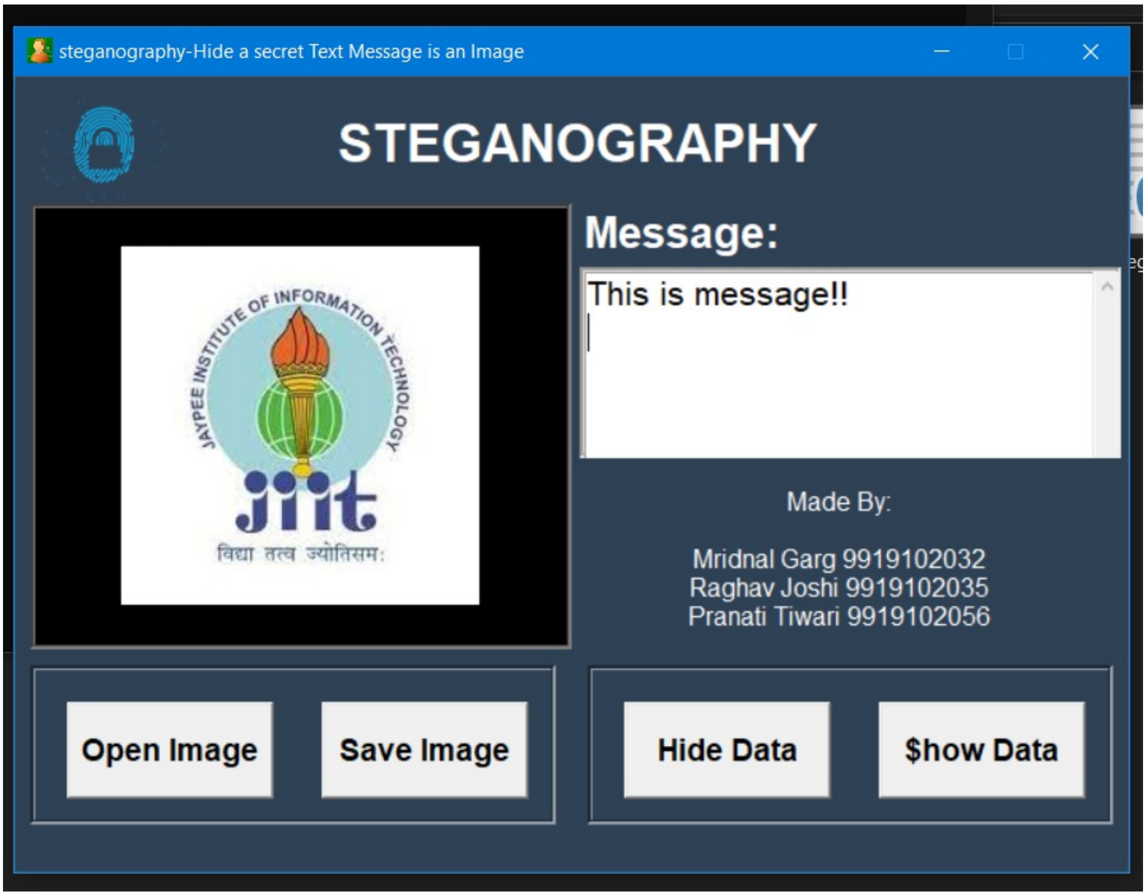
Fig. 5.4 Enter the message to be encrypted



This is message!!

Message successfully hidden !

Fig 5.5 the resolution of the image remains unchanged/ can’t be perceived by human eyes



This is message!!

Made By:

Mridnal Garg 9919102032
Raghav Joshi 9919102035
Pranati Tiwari 9919102056

Fig. 5.6 Message after Decryption

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

- 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 their.
- The goal of steganography is covert communication. So, a fundamental requirement of this steganography system is that the hider message carried by stego-media should not be sensible to human beings.
- The other goal of steganography is to avoid drawing suspicion to the existence of a hidden message. This approach of information hiding technique has recently became important in a number of application area.

- This project has following objectives:
  - To product security tool based on steganography techniques.
  - To explore techniques of hiding data using encryption module of this project
  - To extract techniques of getting secret data using decryption module.

- Normally, after embedding the data into the image, the image may lose its resolution. But in the proposed LSB approach, the image remains unchanged in its resolution as well in size.

REFERENCES

[1] Ramadhan J. Mstafa, Christian Bach "Information Hiding in Images Using Steganography Techniques" Northeast Conference of the American Society for Engineering Education (ASEE) Norwich University March 2013

[2] Omar Elharrouss, Noor Almaadeed, Somaya Al-Maadeed "An image steganography approach based on k-least significant bits (k-LSB)" ICIoT Doha, Qatar 11 May 2020

[3] Amritpal Singh ,Harpal Singh "An improved LSB based image steganography technique for RGB images" ICECCT Coimbatore, India 11 May 2020

[4] S. M. Masud Karim , Md. Saifur Rahman, Md. Ismail Hossain "A new approach for LSB based image steganography using secret key" ICCIT 2011 Dhaka, Bangladesh 09 March 2012,12592219

[5] Anupam Kumar Bairagi , Saikat Mondal, Rameswar Debnath "A robust RGB channel based image steganography technique using a secret key" 16th Int'l Conf. Computer and Information Technology Khulna, Bangladesh ,29 December 2014 14836714

[6] Bret Dunbar "A Detailed look at Steganographic Techniques and their use in an Open-Systems Environment" January 18, 2002

[7] Kadhim, Inas Jawad; Premaratne, Prashan; Vial, Peter James; Halloran, Brendan, “Comprehensive Survey of Image Steganography: Techniques, Evaluations, and Trends in Future Research.” Neurocomputing, vol. 335, pp. 299-326, 2019.

[8] Priya Thomas, “Literature Survey On Modern Image Steganographic Techniques.” International Journal of Engineering Research & Technology (IJERT), Vol. 2 Issue 5, May - 2013, ISSN: 2278-0181