STEGANOGRAPHY

Submitted by,

Name	Roll No.	Registration No.
Shivani Shikha	**CSE/2018/067**	**11700118045**
Surabhi Rani	**CSE/2018/072**	**11700118023**
Ankit Kumar Singh	**CSE/2018/088**	**11700118118**
Sauarbh Kumar Jha	**CSE/2018/091**	**11700118054**

Under the guidance of
Ms. Alokananda Dey

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING
RCC INSTITUTE OF INFORMATION TECHNOLOGY

Session 2018-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
RCC INSTITUTE OF INFORMATION TECHNOLOGY
[Affiliated to West Bengal University of Technology]
CANAL SOUTH ROAD, BELIAGHATA, KOLKATA-700015

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RCC INSTITUTE OF INFORMATION TECHNOLOGY

TO WHOM IT MAY CONCERN

I hereby recommend that the Project entitled **“STEGANOGRAPHY”** prepared under my supervision by

Shivani Shikha(CSE/2018/067,11700118045),
Surabhi Rani(CSE/2018/072,11700118023),
Ankit Kumar Singh(CSE/2018/088,11700118118),
Saurabh Kumar Jha(CSE/2018/091,11700118054)

of B. Tech (8th Semester), may be accepted in partial fulfillment for the degree of **Bachelor of Technology in Computer Science & Engineering** under Maulana Abul Kalam Azad University of Technology (MAKUT).

…………………………………

Project Supervisor
Department of Computer Science and Engineering
RCC Institute of Information Technology

Countersigned:

………………………………………
Head
Department of Computer Sc. &Engg.,
RCC Institute of Information Technology
Kolkata – 700015

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RCC INSTITUTE OF INFORMATION TECHNOLOGY

CERTIFICATE OF APPROVAL

The foregoing Project is hereby accepted as a credible study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project only for the purpose for which it is submitted.

FINAL EXAMINATION FOR
EVALUATION OF PROJECT

1. ____
2. ____
3. ____
4. ____
5. ____
6. ____

(Signature of Examiners)

ACKNOWLEDGEMENT

We acknowledge our overwhelming gratitude & immense respect to our revered guide, **Ms. Alokananda Dey** under whose scholarly guideline, constant encouragement & untiring patience; we have proud privilege to accomplish this entire project work. We feel enriched with the knowledge & sense of responsible approach we inherited from our guide & shall remain a treasure in our life.

Signature:-

Shivani Shikha (CSE/2018/067)	
Surabhi Rani (CSE/2018/072)	
Ankit Kumar Singh (CSE/2018/088)	
Saurabh Kumar jha (CSE/2018/091)	

ABSTRACT

Innovation of technology and having fast Internet make information to distribute over the world easily and economically. From sharing information, to communicating with each other, to exchanging electronic documents, to checking bank balances and paying bills, everything is done within a few clicks using the internet. This has made people to worry about their privacy and security of their works. There are significant interests in security approaches that aim to protect information and digital data, since the growing increase in uses of the internet and multimedia. One of those approaches among the many is Steganography, to be more specific image steganography which is the key objective of this paper. *Steganography* is a technique that prevents unauthorized users to have access to the important data. The main objective of image steganography is to hide the secret data in different embedding medium called as carriers.

In this paper, we have discussed about image steganography, variety of different methods and schemes in image steganography, comparative review for these methods, advantages, disadvantages, the merits and demerits of these techniques as well as provided few examples of data hiding using image steganography.

CONTENTS

CHAPTER-1 Page No

CHAPTER-2

CHAPTER-3

CHAPTER-4

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1.1 Introduction

In recent years, communication between two people or between groups of people has become easy due to the enormous growth of internet. So, security and confidentiality of highly sensitive data has become an issue of supreme importance and concern. To protect the information from an unauthorized person data hiding method has been developed. Three methods are interlinked to data hiding are

- **Cryptography**
- **Watermarking and**
- **Steganography.**

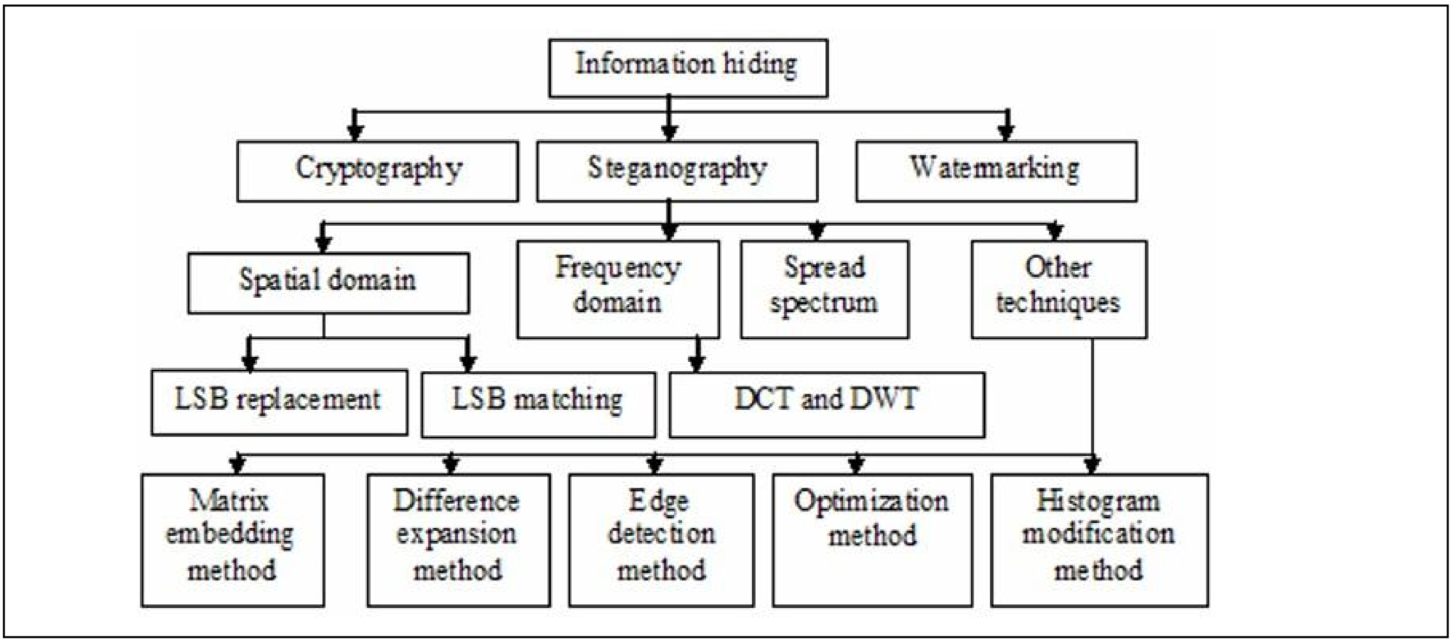


Fig 1 Flowchart of Information Hiding

Steganography comes from the combination of the Greek words **Stegano** means "**covered"** and **Graphy** referring to "**writing"** which collectively makes it "**covered writing**"[1][2] . Steganography is a very old art of embedding personal information into other data by using some rules and techniques[18]. As a result, unauthorized users are not able to see and recognize the embedded information. Steganography is managing a secret path for sending information invisibly.

As shown in **Figure 2**,[12] the aim of steganography is to conceal the message under cover files, concealing the very existence of information exchange.

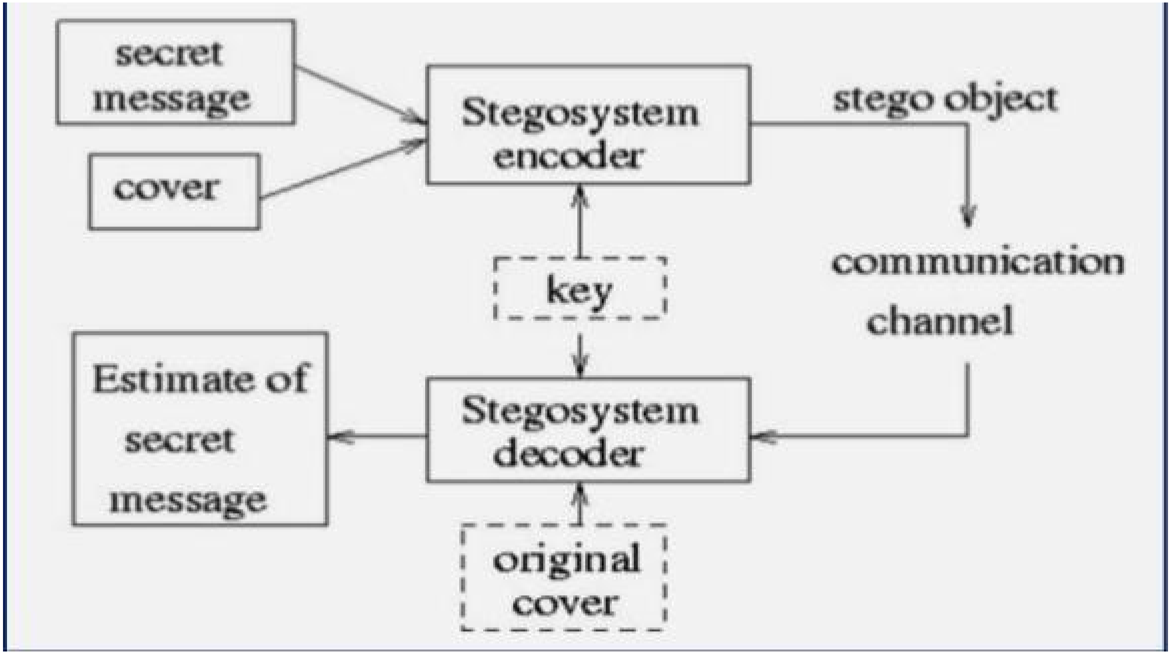


Figure 2 : A typical Stegangraphy Technique

In General, Steganography is classified into four types as follows [1]:

1. Image steganography: It is a process of concealing the secret image inside the cover image in such a way that the existence of the secret image is disappeared and the cover image seems to be original.

2. Audio steganography: Digital sound files are used to hide a secret message by vaguely changing the binary sequence of a sound file, which is known as audio steganography.

3. Video steganography: Video files can be defined as a collection of images and sounds combined together, thus, most of the introduced images and audio can be used and applied to the digital video files. In fact, large amount of secret data that can be embedded inside the video files, since the video file is a moving stream of images and sounds.

4. Text steganography: Text steganography basically refers to the information that is hidden in text files. The text steganography includes everything from manipulating and changing text formatting, word changing within the text, producing and generating random sequences or using context-free language grammars to generate readable texts.

Steganography requires three main components, namely carrier object, secret data, and steganographic algorithm.

Steganography can be used for many useful applications, such as: secure transmission of top-secret data between national and international governments, online banking security, military and intelligent agencies security and safe circulation of secret documents among defense organizations [1][3].

The main theme of cryptography is to secure communication by changing the data into a form that an unauthorized person cannot understand.

Digital watermarking is the act of embedding a watermark permanently into digital data in which the watermark can be easily detected (or) extracted later in order to ensure the authenticity of the digital data. The digital data may be audio, video, images and text. The embedded watermark is a signature that refers to the ownership of the data for copyright protection[19].

Steganography differs from cryptography and watermarking. Image steganography is the method of hiding highly sensitive information in a cover image and the resulting stego image is transferred securely through an unsecure channel. In steganography, sensitive information is invisible to the unauthorized person, while in cryptography the unauthorized person cannot able to understand the sensitive information. So, in some cases sending encrypted data may draw attention while invisible data may not. Watermarking embed the information related to the digital data, while in steganography there is no relation between information and digital data. Among these methods, steganography provides a high level of security to the secret data.

1.2 Literature Review

According to the recent research, eight different approaches are discussed in this work. They are [13]

1 Spatial domain

2 Frequency domain

3 Spread spectrum method

4 Matrix embedding method

5 Difference expansion (DE) method

6 Edge detection method

7 Optimisation method

8 Histogram modification method.

1. **Spatial Domain Techniques**: In spatial domain techniques, carrier object pixels, like image and video objects, are directly manipulated and changed in order to hide secret data inside it. The following techniques belong to spatial domain [1][4-6]:

i. Least Significant Bit (LSB): It is a simple strategy for implementing steganography.It embeds the data into the cover, so that it cannot be detected by a casual observer. The technique works by replacing some of the information in a given pixel with information from the data in the image.

ii. Gray-Level Modification (GLM) : Gray level Modification (GLM) is defined as a technique in which the grey level values of the image pixels are modified in accordance with a mathematical function to represent binary data. Each pixel has a distinct grey level value which can have an odd or even value. This odd or even value of the grey level is appropriately modified to represent binary data. [7]

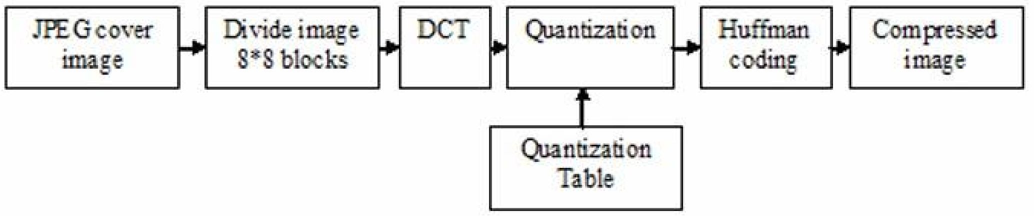
iii. Pixel Value Differencing (PVD) : Pixel-value differencing (PVD) scheme uses the difference value between two consecutive pixels in a block to determine how many secret bits should be embedded. It provides high imperceptibility to the stego image by selecting two consecutive pixels and designs a quantization range table to determine the payload by the difference value between the consecutive pixels. Besides, it offers the advantage of conveying a large number of payloads, while still maintaining the consistency of an image characteristic after data embedding. [8]

2 **Frequency domain** : In frequency domain, the image is transformed from time domain into frequency domain using some transform and secret data is hidden in the transform coefficient. The frequency domain embed less amount of information into image but however depends on the number of pixel and depth of the colour the embedding capacity of the secret data was determined. The frequency domain provides more robustness against image compression, cropping, rotation, etc.

The common frequency domain methods used in steganographic system are 2-D DCT and DWT. These methods are explained below.

Discrete cosine transform : The discrete cosine transform (DCT) is used for JPEG images. DCT transformation are performed by dividing the image into 8 * 8 pixel blocks and these blocks are transformed into 64-DCT coefficient and followed by quantisation and Huffman coding step to get the compressed image. The block diagram of JPEG image compression using DCT is shown in Figure 3.

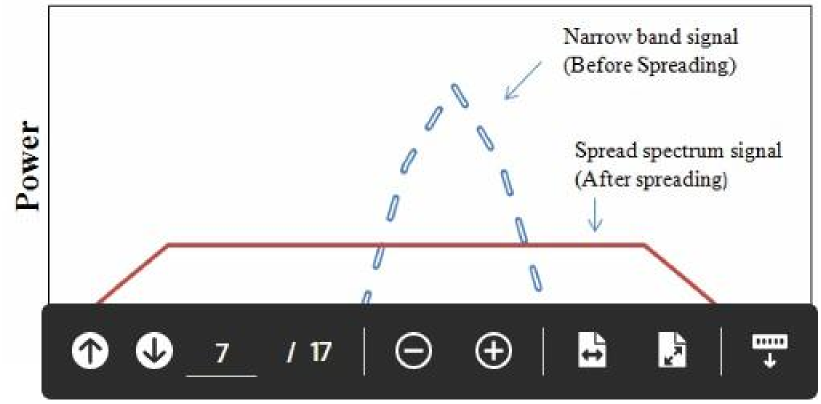
Figure 3 Block diagram of jpeg compression using DCT



Discrete wavelet transform: The discrete wavelet transform (DWT) efficiently transform the image from spatial domain into frequency domain. The wavelet transform is the widely used transform for much research under steganography system. The main advantage of wavelet transform is it divides the high- frequency and low-frequency data on a pixel by pixel basis clearly. It also approximates the data with sharp discontinuities [14]. It transform the image into various level depends on the application.

3**. Spread spectrum technique**: Spread spectrum steganographic technique was proposed by Marvel [15]. Spread spectrum is the process of spreading the bandwidth of narrow band signal across a wideband of frequencies. This can be obtained by modulating the narrowband signal with a wideband signal. After spreading, the energy of the narrowband signal at any frequency becomes low and thus hard to detect. The concept behind the spread spectrum is the secret data is embedded or hided as noise to cover image captured by photo electronic devices. The noise may be white Gaussian noise. Then, the secret data is modulated with pseudorandom generated sequence and added to the cover image. The explanation of spread spectrum is shown in Figure 4.

Figure 4 Concept of spread spectrum (see online version for colours)

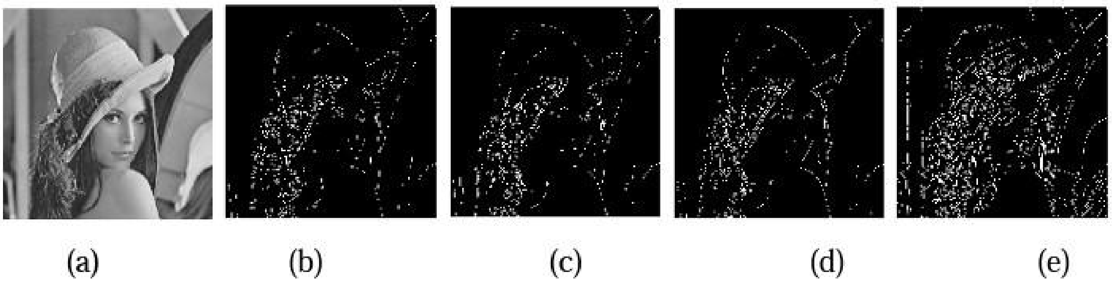


Matrix encoding was introduced by Crandall [16]. Matrix encoding embeds the secret message and cover image with some error correction code and then according to the coding result it modifies the cover image. Matrix embedding provides a high embedding efficiency and lower embedding distortions.

5**. DE method**: DE method was proposed by Tian [17]. DE is a method of hiding the secret data reversibly in pixel values. It utilises the high correlation of the digital image (or) cover image, thus highly correlated cover image are suggested decreasing the distortions and improving the embedding capacity.

6 Edge detection method: The edges of digital image can embed or hide more secret data without dissuading the quality of an image as the distortion at edges also cannot be detected easily by human eye. This is the reason why the edge detection method has been established. Edge detection is the method of identifying the points in an image at which the brightness of image changes sharply or formally has discontinuities. Several edge detector approaches have been shown in Figure 5.

Figure 5 Various edge detector methods, (a) cover image (b) Sobel method (c) Prewitt method (d) Roberts method (e) Canny method



7. **Optimisationmethod** : The four fields of soft computing are fuzzy computing, artificial neural networks, evolutionary computing and probabilistic computing. Genetic algorithm (GA) and particle swarm optimisation (PSO) are the two subcategories of evolutionary computing which have been used to solve the steganography problems. GA is different from the PSO. GA follows the randomised search procedure to solve the optimisation problem. It uses the operators such as selection, recombination, crossover and mutation. PSO algorithm begins with a population of haphazard location and search for an optimum via updating generations and it does not have crossover and mutation operators. GA and PSO is utilised to find the best scanning order, starting point in the spatial domain and to find the best coefficient or best directions in transform domain, to hide the secret message in selected pixel. This will decrease the distortion in steganography and also increases the embedding capacity. The optimisation method can combine with the spatial domain or with the frequency domain.

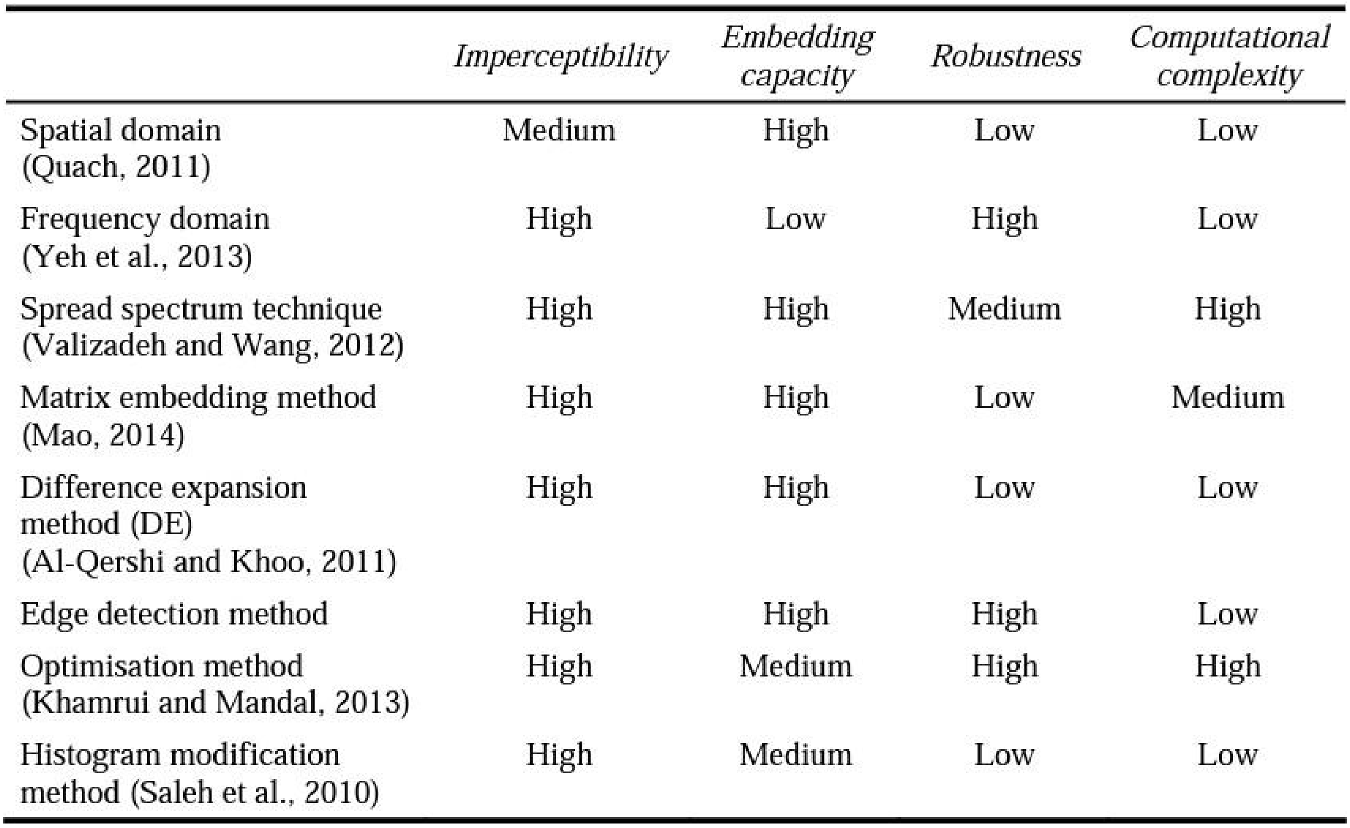
8. **Histogram modification method**: The histogram of a digital image specifies the entire tonal distribution of the image. Histogram modification-based steganographic method utilises this concept to embed the secret data. Histogram modification utilises the peak and minimum value within the histogram of image to hide the secret bit. However, the security is not guaranteed in this work.

Comparative analysis of different steganographic techniques

All the above mentioned steganographic techniques are not void of strong and weak points. It is very important to select the suitable techniques to be applied according to the applications. The parameters to measure the performance of steganographic system are:

1 Imperceptibility: This parameter represents the skill to avoid distortions, where the human eye fails to notice it.
2 Robustness: This parameter measures ability of payload to survive during embedding and extraction phase even in the manipulation of stego-image such as image compression, image manipulation and image filtering.
3 Embedding capacity: This parameter represents the maximum amount of secret data that can be embedded and extracted successfully.
4 Computational complexity: This parameter represents the complexity of implementing the techniques. The comparative analysis of various above mentioned steganographic techniques in terms of the competing parameters are shown in Table 1.

Table 1 A comparative analysis of various steganographic techniques



	Imperceptibility	*Embedding capacity*	*Robustness*	*Computational complexity*
Spatial domain (Quach, 2011)	Medium	High	Low	Low
Frequency domain (Yeh et al., 2013)	High	Low	High	Low
Spread spectrum technique (Valizadeh and Wang, 2012)	High	High	Medium	High
Matrix embedding method (Mao, 2014)	High	High	Low	Medium
Difference expansion method (DE) (Al-Qershi and Khoo, 2011)	High	High	Low	Low
Edge detection method	High	High	High	Low
Optimisation method (Khamrui and Mandal, 2013)	High	Medium	High	High
Histogram modification method (Saleh et al., 2010)	High	Medium	Low	Low

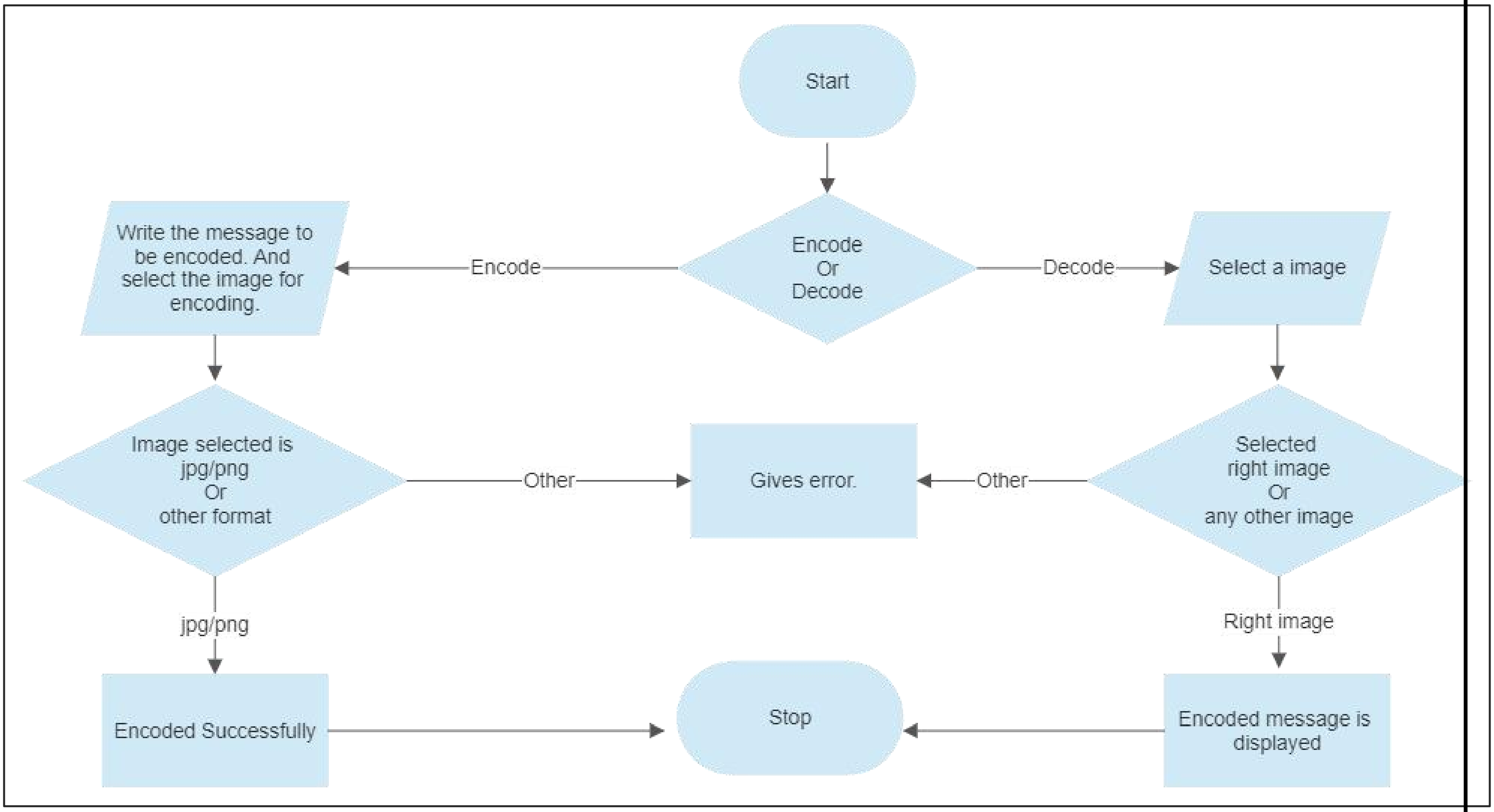
Table 2: Literature Review [2][9-11]

NO	Research Name	Domain	Methodology	Advantages	disadvantages	Evaluation
1	A Modified Image Steganography Method based on LSB Technique s [20].	Spatial	The message is expressed in 6 binary bits using LSBraille method, rather than ASCII format. Three message's bits are embedded in one pixel as follows: two bits are embedded in blue layer, and one bit is embedded in green layer.	• Security of LSB steganographic technique is slightly improved. • It provides a high performance, since it based on LSB1 and LSB2 algorithms. • It supports all image formats.	• Limited space for embedding a message. • It provides weak robustness because it based on LSB1 and LSB2 methods, which are easy to extract the original message. • It does not providean encryption.	PSNR using MATLAB 11.1.0.
2	MLSB Technique based on 3D Image Steganography Using AES Algorithm [21]	Transfor m	A secret message is encrypted using Advanced Encryption Standard (AES). Then, they followed a set of rules: i. Selecting 3D Image and Previewing ii. Embedding and Encrypting Data iii. Decrypting and Extra cting Image.	• Security of LSB technique is improved by applying an encryption to a secret message. • Its design supports multilayer approach. • It supports all image formats and sizes.	• It uses DWT method, which has a negative impact on performance. • It based on traditional LSB, which is easy to recover the original message. • It does distribute a sharable key in a secure manner.	MATLAB (not PSN R results)
3	A Proposed Algorithm for Steganogr aphy in Digital Image based on Least Significan t Bit (LSB) [22].	Spatial	Instead of using the LSB-1, LSB-2 is utilized to maximize the robustness. It provided more protection to the message bits, since a Stego-Key has been used to permute the message bits before embedding it.	• It provides a high performance, since it based on LSB1 and LSB2 algorithms, and by making small modifications to its pixels. • It supports all image formats.	• Limited to image size 256×256, which is easy to recover the embedded message. • It provides weak robustness because it based on LSB1 and LSB2 methods. • It does not provide an encryption.	PSNR

4	A Comprehensive Image Steganography Tool using LSB Scheme [23].	Spatial	IMStego tool helps a user to hide and extract secret data using 1-LSB or 2-LSB algorithm into color images, and only limited to BMP and PNG image formats.	• It creates a Java-based tool called IMStego to embed a secret message into images using 1-LSB and 2-LSB. • It provides a high performance, since it based on LSB1 and LSB2 algorithms.	• Limited to only BMP and PNG image formats. • It does distribute a sharable key in a secure manner. • It does not provide an encryption.	IMStego Java-based Tool (not PSNR results)

CHAPTER 2

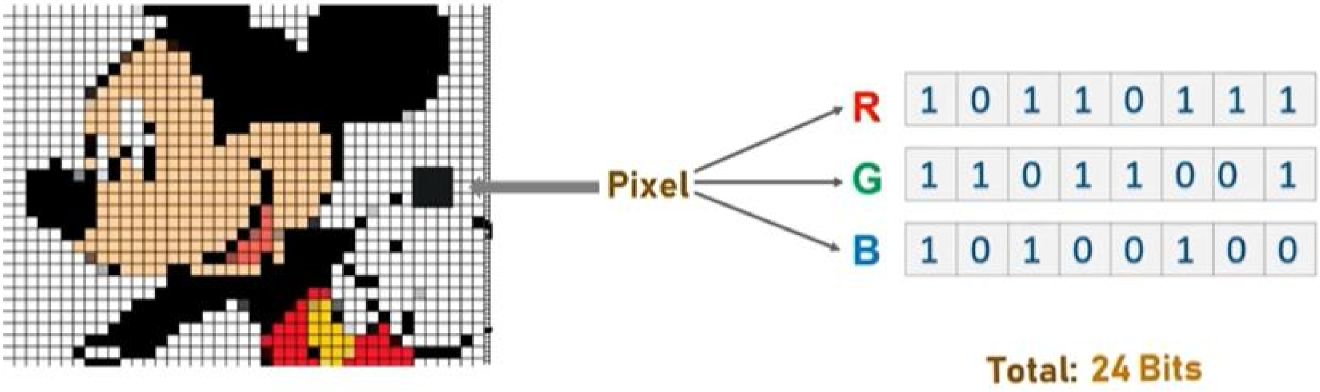
2.1 Work Flow

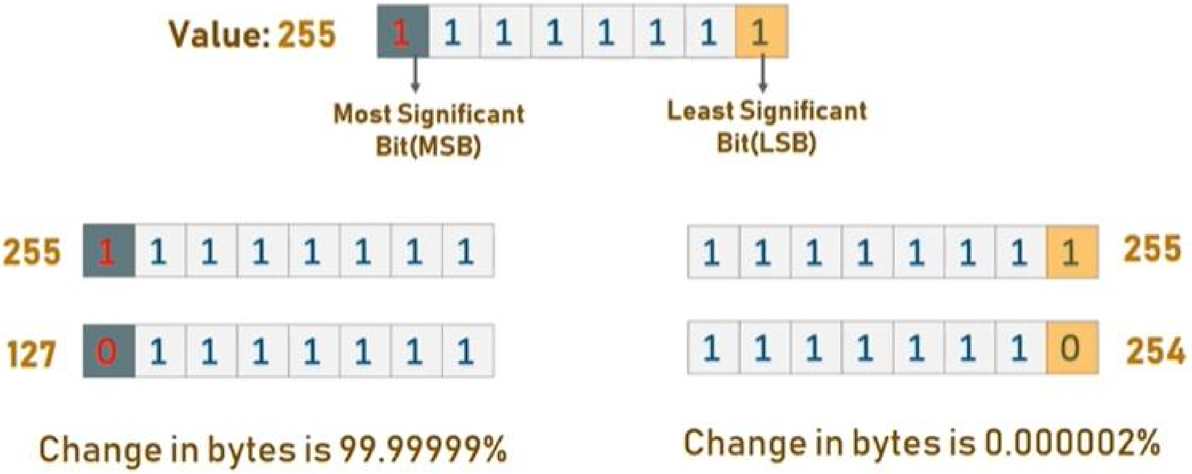


2.2Proposed Work

We have tried to implement image steganography using LSB technique, i.e; Least Significant Bit Technique.
LSB Steganography is an image steganography technique in which messages are hidden inside an image by replacing each pixel's least significant bit with the bits of the message to be hidden. To understand better, let's consider a digital image to be a 2D array of pixels. Each pixel contains values depending on its type and depth. We will consider the most widely used modes — **RGB(3x8-bit pixels, true-color)** and **RGBA(4x8-bit pixels, true colorwith transparency mask).** These values range from 0–255, (8-bit values). We can convert the message into decimal values and then into binary, by using the ASCII Table. Then, we iterate over the pixel values one by one, after converting them to binary, we replace each least significant bit with that message bits in a sequence. To decode an encoded image, we simply reverse the process. Collect and store the last bits of each pixel then split them into groups of 8 and convert it back to ASCII characters to get the hidden message.

Least Significant Bit (LSB) is a technique in which the last bit of each pixel is modified and replaced with the secret message's data bit.




Value: 255 1 1 1 1 1 1 1 1
Most Significant Bit(MSB)
Least Significant Bit(LSB)
255 1 1 1 1 1 1 1 1
127 0 1 1 1 1 1 1 1
Change in bytes is 99.99999%
1 1 1 1 1 1 1 1 255
1 1 1 1 1 1 1 0 254
Change in bytes is 0.000002%

From the above image it is clear that, if we change MSB it will have a larger impact on the final value but if we change the LSB the impact on the final value is minimal, thus we use least significant bit steganography.

Code:-

```
#importing modules for python Image Steganography project
from tkinter import *
from tkinter.filedialog import *
from PIL import ImageTk,Image
from stegano import exifHeader as stg
from tkinter import messagebox

import cv2
import numpy as np
from PIL import Image

#it convert data in binary formate
def data2binary(data):
if type(data) == str:
p = ''.join([format(ord(i), '08b')for i in data])
elif type(data) == bytes or type(data) == np.ndarray:
p = [format(i, '08b')for i in data]
return p

# hide data in given img
def hidedata(img, data):
data += "$$"
#'$$'--> secrete key
d_index = 0
b_data = data2binary(data)
len_data = len(b_data)

#iterate pixels from image and update pixel values
for value in img:
for pix in value:
r, g, b = data2binary(pix)
if d_index < len_data:
pix[0] = int(r[:-1] + b_data[d_index])
d_index += 1
if d_index < len_data:
pix[1] = int(g[:-1] + b_data[d_index])
d_index += 1
if d_index < len_data:
pix[2] = int(b[:-1] + b_data[d_index])
d_index += 1
if d_index >= len_data:
break
return img

# decoding
def find_data(img):
bin_data = ""
for value in img:
for pix in value:
r, g, b = data2binary(pix)
bin_data += r[-1]
bin_data += g[-1]
bin_data += b[-1]
all_bytes = [bin_data[i: i + 8] for i in range(0,
len(bin_data), 8)]
readable_data = ""
for x in all_bytes:
readable_data += chr(int(x, 2))
if readable_data[-2:] == "$$":
```

```
break
return readable_data[:-2]

def decode():
img_name = input("\nEnter Image name : ")
image = cv2.imread(img_name)
img=Image.open(img_name,'r')
msg = find_data(image)
return msg

# decoding the file for python Image Steganography project
def Decode():
Screen.destroy()
DecScreen = Tk()
DecScreen.title("Decoding Screen")
DecScreen.geometry("500x500+300+300")
DecScreen.config(bg="pink")
def OpenFile():
global FileOpen
FileOpen=StringVar()
FileOpen =
askopenfilename(initialdir="/Desktop",title="Select the
File",filetypes=(("only png files","*png"),("all type of
files","*.*")))

def Decoder():
image = cv2.imread(FileOpen)
img=Image.open(FileOpen,'r')
msg = find_data(image)

label3 = Label(text=msg)
label3.place(relx=0.3,rely=0.3)

SelectButton = Button(text="Select the
file",command=OpenFile)
SelectButton.place(relx=0.1,rely=0.4)

EncodeButton=Button(text="Decode",command=Decoder)
EncodeButton.place(relx=0.4,rely=0.5)

def Encode():
Screen.destroy()
EncScreen = Tk()
EncScreen.title("Encoding Screeen")
EncScreen.geometry("500x500+300+300")
EncScreen.config(bg="yellow")
label = Label(text="Confidential Message")
label.place(relx=0.1,rely=0.2)
entry=Entry()
entry.place(relx=0.5,rely=0.2)
label1 = Label(text="Name of the File")
label1.place(relx=0.1,rely=0.3)
SaveEntry = Entry()
SaveEntry.place(relx=0.5,rely=0.3)

def OpenFile():
global FileOpen
FileOpen=StringVar()
FileOpen = askopenfilename(initialdir = "/Desktop" ,
title = "SelectFile",filetypes=(("jpeg files","*jpg"),("all
files","*.*")))
```

```
label2 = Label(text=FileOpen)
label2.place(relx=0.3,rely=0.3)

def Encoder():
Response= messagebox.askyesno("PopUp","Do you
want to encode the image")
if Response == 1:
image = cv2.imread(FileOpen)
img = Image.open(FileOpen, 'r')
w, h = img.size
data = entry.get()
if len(data) == 0:
raise ValueError("Empty data")
enc_img = SaveEntry.get()+".png"
enc_data = hidedata(image, data)
cv2.imwrite(enc_img, enc_data)
img1 = Image.open(enc_img, 'r')
img1 = img1.resize((w, h),Image.ANTIALIAS)
# optimize with 65% quality
if w != h:
img1.save(enc_img, optimize=True, quality=65)
else:
img1.save(enc_img)
messagebox.showinfo("Pop Up","Successfully
Encoded the image")
else:
messagebox.showwarning("Pop
Up","Unsuccessful,please try again")

SelectButton = Button(text="Select the
file",command=OpenFile)
SelectButton.place(relx=0.1,rely=0.4)

EncodeButton=Button(text="Encode",command=Encoder)
EncodeButton.place(relx=0.4,rely=0.5)

#Initializing the screen for python Image Steganography
project
Screen = Tk()
Screen.title("Image Steganography Project 2022")
Screen.geometry("500x500+300+300")
Screen.config(bg= "blue")
#creating buttons
EncodeButton = Button(text="Encode",command=Encode)
EncodeButton.place(relx=0.3,rely=0.4)

DecodeButton = Button(text="Decode",command=Decode)
DecodeButton.place(relx=0.6,rely=0.4)

mainloop()
```

Code Explanation:

Steps to follow to develop Image Steganography Project using Python:
1. Installation of Tkinter and Stegano modules
2. Importing modules
3. Initializing the screen
4. Function to decode the image
5. Function to encode the image

1. Installation of Tkinter and Stegano modules:
If these modules are not installed on your system, install these modules by running the following command on the command prompt or the terminal.

2. Importing modules:
a. Tkinter: It is the Graphical user interface package.
b. tkinter.filedialog : This module offers a set of classes and functions that can be used to work with files.
c. PIL: It helps to save many different formats of images.
d. Stegano: It’s the python steganography module.
e. Messagebox: It is used to display the message boxes.

3. Initializing the screen:
a. Tk(): Main window is created with the help of Tk().
b. title(): The title on the main window is displayed with the help of this function.
c. geometry(): The geometry of the screen(length,height,width) is set with the help of this function.
d. config(): It helps to access the attributes of the object after initialisation.
e. bg: It sets the background colour of the screen.
f. Button(): It creates the button on the screen.
g. place: It is used to set the position and size of a window.
h. relx: It is the fraction of the width of the parent widget.
i. Rely: It is the fraction of the height of the parent widget.

4. Function to decode the image:
Decode function is made to decode the image. Decscreen variable creates the decode window screen. OpenFile function is used for selecting files and Decoder function is used to decode the image and display the text on the screen.
a. destroy(): It is used to destroy the widget.
b. StringVar(): It holds a string.
c. askopenfilename(): The selected file name is returned by this function.
d. reveal(): It is used to reveal the hidden message.
f. Label(): It specifies the container box where the text can be placed.

5. Function to encode the image:
Encode function takes the secret message that will be displayed after decoding the image and takes the image’s file name. Encscreen variable is the encoded window screen. OpenFile function is used to select the file that is to be encoded and Encoder function encodes the image.
a. Entry(): Single line text strings are accepted by this function from the user.
b. askyesno: It shows a dialog box for confirmation from the user.
c. showinfo(): It is used to display an appropriate message.
d. showwarning(): It displayed the warning message to the user.
e. mainloop(): In tkinter, the event loop is runned by this function.

CHAPTER 3

3.1 Test Cases & System Validation

Test cases - we have to store the confidential message behind the image or in the form of image, audio, video and text document. Here, we are doing the image steganography that's why we are sending the confidential message in the form of Image Only.

Step – 1: Execute the program for Image steganography.
Step – 2: Select the Encode Button first.
Step – 3: Leave the confidential message whatever you want and select the file(image).
Step – 4: Click on the Encode button for encodinig the message in the form of selecting Image.
Step – 5: After doing all this you will see a pop-up of successfully encoded.

Another part of the execution of the program is Decoding part also. If we want to decode the any image then we can do by this program.

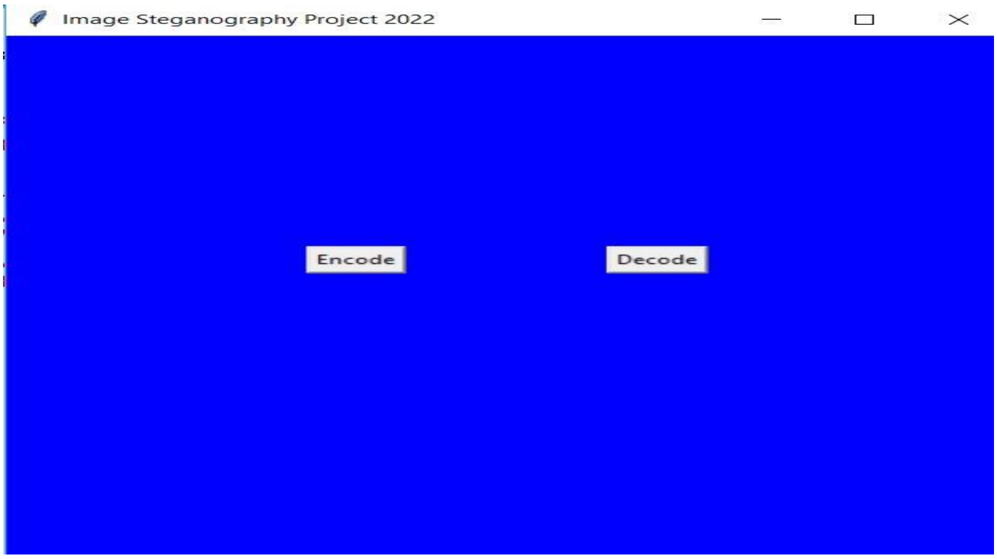
Step – 1: Firstly, Execute the program for decoding.
Step – 2: Select the file(Image) which file you want to decode.
Step – 3: After selecting the file click on the decode button.
Step – 4: After doing all this you will see the that confidential message which image was encoded.

3.2 Observed Output

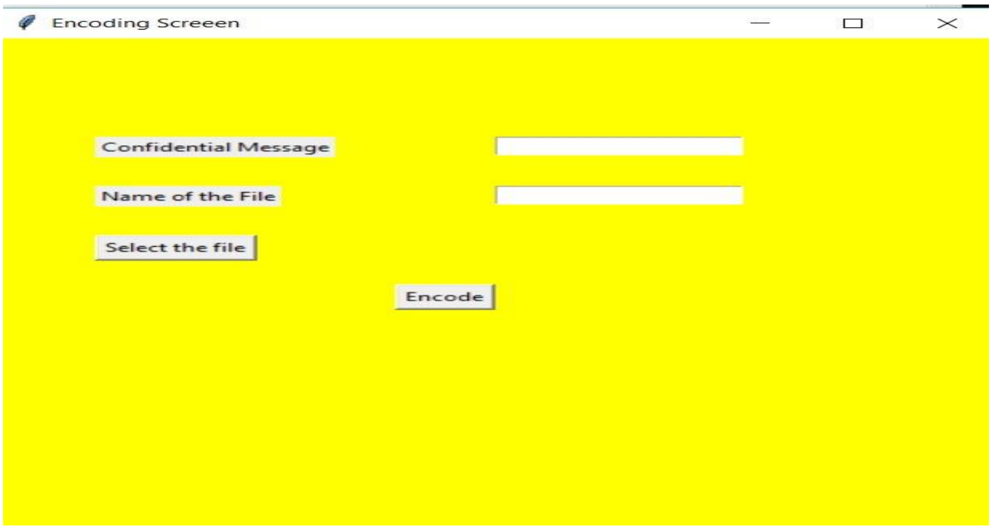
Case 1 :- Successfully Encoded and Decoded

For Encoding:

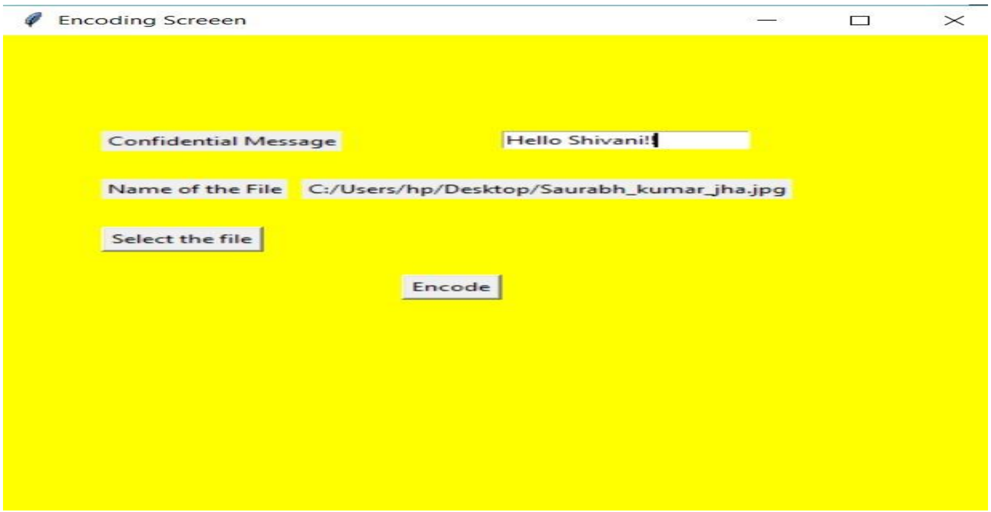
Step 1 - The opening screen where we will choose encoding.



Step 2 - Write the confidential message that is to be encoded and choose a .jpg/.png file to perform the required image steganography.



Step 3 - Writing message and choosing image.

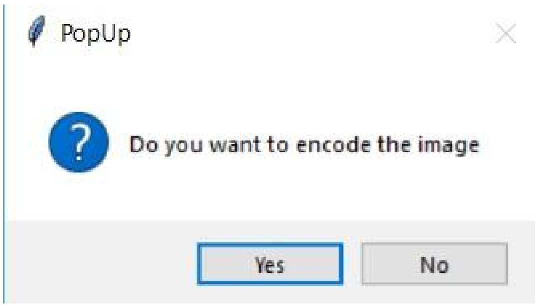


Encoding Screeen

Confidential Message: Hello Shivani!

Name of the File: C:/Users/hp/Desktop/Saurabh_kumar_jha.jpg

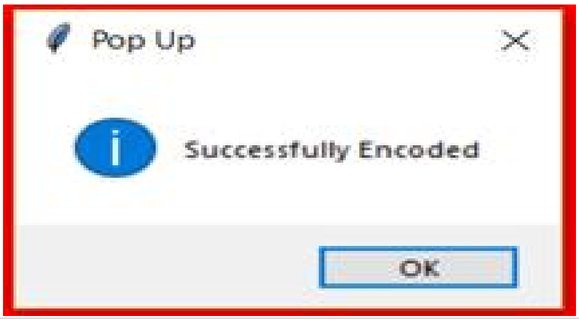
Step 4 - After clicking on the button encode this pop up appears.



PopUp

Do you want to encode the image

Step 5 - If the encoding is successful then the below pop up appears.

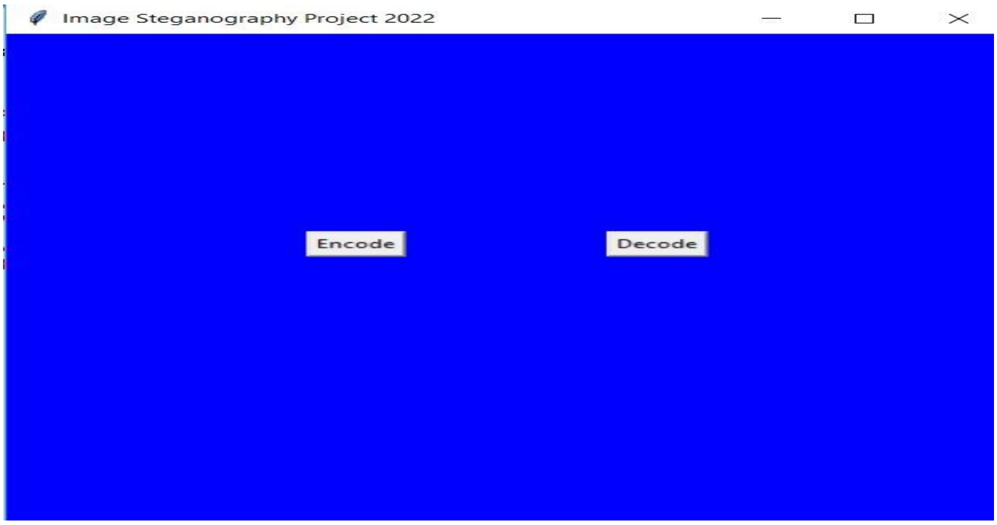


Pop Up

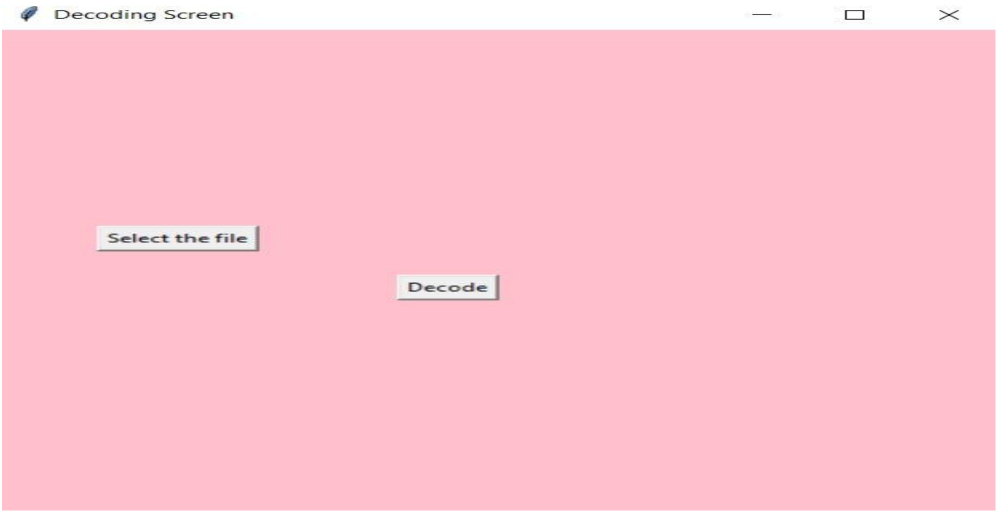
Successfully Encoded

For Decoding:

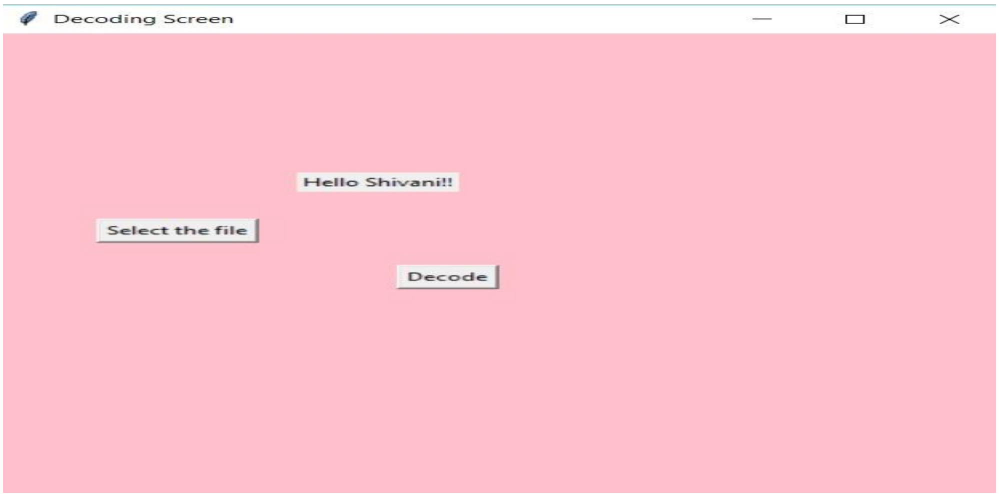
Step 1 - From the opening screen this time we will choose decoding.



Step 2 - Select the file which has the encoded message and click on the button deocde.



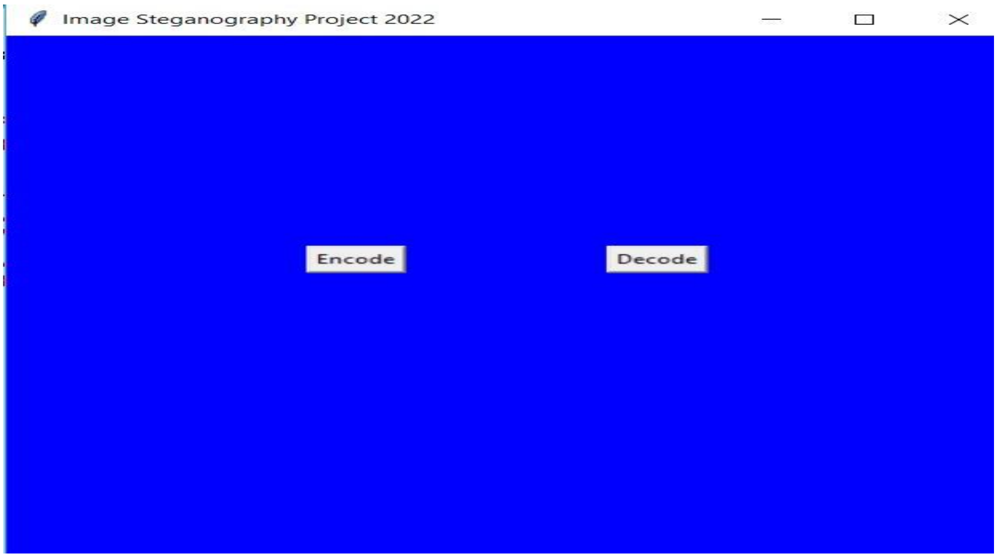
Step 3 - If the right file is selected, the secret message is displayed on the screen as shown below.



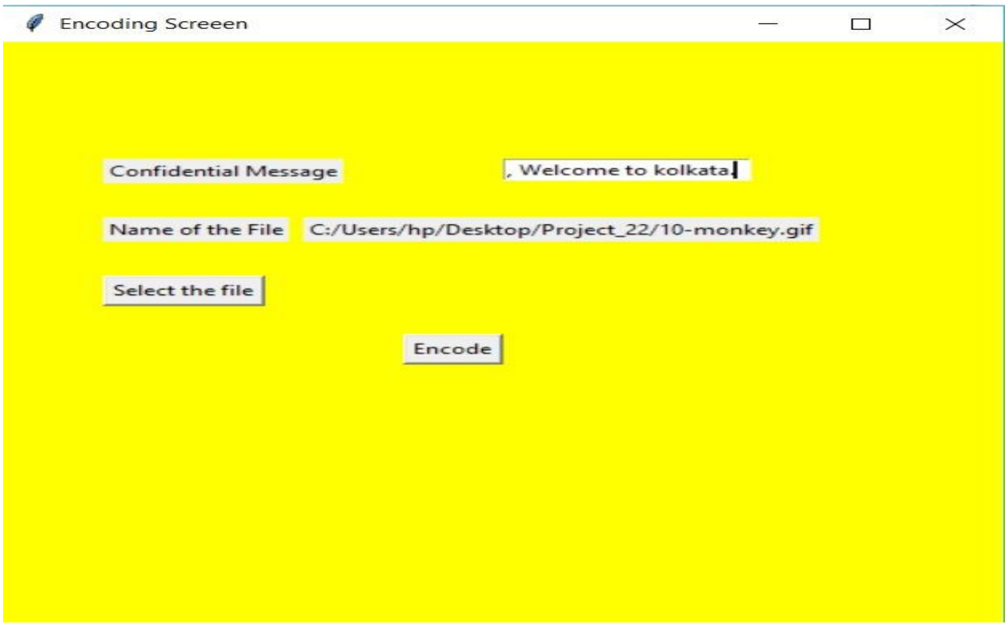
Case 2 :- Unsuccessful encoding

For Encoding:

Step 1 - The opening screen where we will choose encoding.



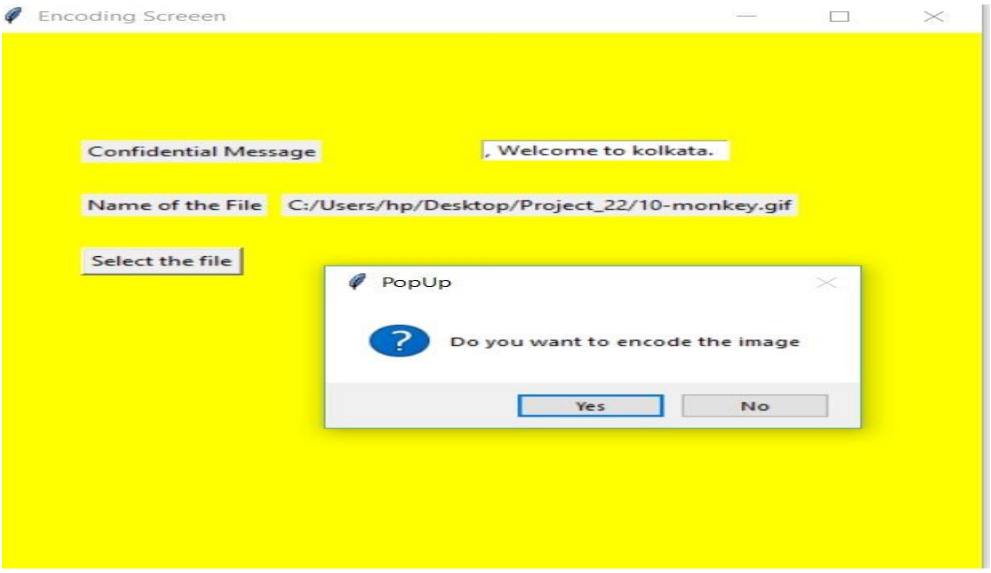
Step 2 - In this case we select a .gif file to encode the given message.



Confidential Message , Welcome to kolkata

Name of the File C:/Users/hp/Desktop/Project_22/10-monkey.gif

Step 3 - After clicking the encode button , the pop up appears for confirmation and we choose yes.

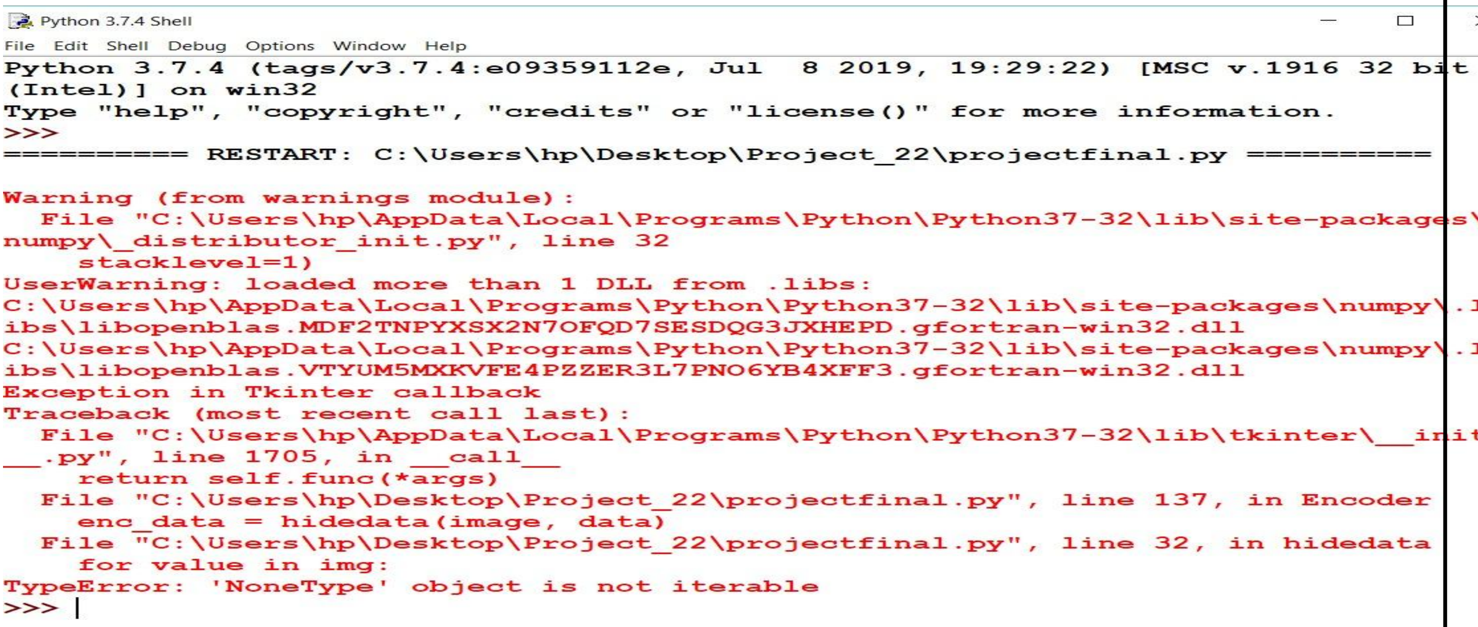


Confidential Message , Welcome to kolkata.

Name of the File C:/Users/hp/Desktop/Project_22/10-monkey.gif

Do you want to encode the image

Step 4 - As the choosen file is .gif and not .jpg/.png we face a error and the encoding remains unsuccessful.



```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\hp\Desktop\Project_22\projectfinal.py ==========

Warning (from warnings module):
  File "C:\Users\hp\AppData\Local\Programs\Python\Python37-32\lib\site-packages\
numpy\_distributor_init.py", line 32
    stacklevel=1)
UserWarning: loaded more than 1 DLL from .libs:
C:\Users\hp\AppData\Local\Programs\Python\Python37-32\lib\site-packages\numpy\.l
ibs\libopenblas.MDF2TNPYXSX2N7OFQD7SESDQG3JXHEPD.gfortran-win32.dll
C:\Users\hp\AppData\Local\Programs\Python\Python37-32\lib\site-packages\numpy\.l
ibs\libopenblas.VTYUM5MXKVFE4PZZER3L7PNO6YB4XFF3.gfortran-win32.dll
Exception in Tkinter callback
Traceback (most recent call last):
  File "C:\Users\hp\AppData\Local\Programs\Python\Python37-32\lib\tkinter\__init
__.py", line 1705, in __call__
    return self.func(*args)
  File "C:\Users\hp\Desktop\Project_22\projectfinal.py", line 137, in Encoder
    enc_data = hidedata(image, data)
  File "C:\Users\hp\Desktop\Project_22\projectfinal.py", line 32, in hidedata
    for value in img:
TypeError: 'NoneType' object is not iterable
>>> |
```

CHAPTER4

4.1 Conclusion

A structured survey of various techniques in image steganography-based data hiding has been presented. A key issue in the steganography is the difficulty to improve the embedding capacity and quality of stego image. In this survey paper, we have given basic ideas and overview of existing steganography techniques and provided a sample example of doing the same with help of a python code. In addition, digital image processing research in image steganography could benefit from the large training set of images and change the embedding capacity bits that could be accessed by observers for evaluation of their techniques.

4.2 Future Scope

- To perform image steganography for messages of any length
- To perform image steganography of any size of images.
- Implementation of complete client server based application program.
- Implementation of error correction and detection methods at receiver end.

References

[1]. S. Kurane, H. Harke, and S. Kulkarni, “TEXT AND AUDIO DATA HIDING USING LSB AND DCT A REVIEW APPROACH,” Natl. Conf. “Internet Things Towar. a Smart Futur. “Recent Trends Electron. Commun., 2016.

[2]. E. Nandhini, M. Nivetha, S. Nirmala, and R. Poornima, “MLSB Technique Based 3D Image Steganography Using AES Algorithm,” J. Recent Res. Eng. Technol. ISSN, vol. 3, no. 1, p. 2936, 2016.

[3]. K. Muhammad, “A Secure Cyclic Steganographic Technique for Color Images using Randomization,” Tech. Journal, Univ. Eng. Technol. Taxila, 2014.

[4]. M. H. and M. Hussain, “A Survey of Image Steganography Techniques,” Int. J. Adv. Sci. Technol., vol. 54, pp. 113–124, 2013.

[5]. N. Hamid and R. B. Ahmad, “Image Steganography Techniques: An Overview,” no. 6, pp. 168–187, 2012.

[6]. J. Kour and D. Verma, “Steganography Techniques –A Review Paper,” Int. J. Emerg. Res. Manag. &Technology, vol. 9359, no. 35, pp. 22789359, 2014.

[7]. E. C. Vidyasagar M. Potdar, “Grey Level Modification Steganography for Secret Communication,” 2004. [Online]. Available: https://www.researchgate.net/publication/4137627_Grey_level_modification_steganography_for_secret_communi cation.

[8]. H.-W. T. and H.-S. Leng, “A Steganographic Method Based on Pixel-Value Differencing and the Perfect Square Number,” 2013. *Online+. Available: http://www.hindawi.com/journals/jam/2013/189706/.

[9]. M. M. Emam, A. A. Aly, and F. A. Omara, “A Modified Image Steganography Method based on LSB Technique,” Int. J. Comput. Appl., vol. 125, no. 5, p. 9758887, 2015.

[10]. A. E. Mustafa, A. M. F. Elgamal, M. E. Elalmi, and A. Bd, “A Proposed Algorithm For Steganography In Digital Image Based on Least Significant Bit,” Res. J. Specif. Educ., no. 21, 2011.

[11]. Sahar A. El_Rahman, “A Comprehensive Image Steganography Tool using LSB Scheme,” I.J. Image, Graph. Signal Process., 2015

[12] “Image Steganography Techniques - A Review Paper “ by Mohammed A. Saleh ,College of Sciences and Arts in Ar Rass, Qassim University, Kingdom of Saudi Arabia, IJARCCE International Journal of Advanced Research in Computer and Communication Engineering Vol. 7, Issue 9, September 2018

[13] S Uma Maheswari - Coimbatore Institute of Technology and Jude Hemanth D Karunya University , “Different methodology for image steganography-based data hiding: Review paper”, 2015

[14] Kruus, P., Scace, C., Heyman, M. and Mundy, M. (2003) ‘A survey of steganography techniques for image files’, Advanced Security Research Journal, Vol. 5, No. 1,pp.41–52.

[15] Marvel, L.M., Boncelet, C.G. and Retter, C.T. (1999) ‘Spread spectrum image steganography’, IEEE Transactions on Image Processing, Vol. 8, No. 8, pp.1075–1083.

[16] Crandall, R. (1998) ‘Some notes on steganography’, Steganography Mailing List *online+ http://os.inf.tu- dresden.de/ westfeld/crandall.pdf.

[17] Tian, J. (2002) 'Reversible watermarking by difference expansion', Proc. Workshop on Multimedia and Security, pp.19–22.

[18]Jin-Suk, K., Yonghee, Y., & Mee Young, S. (2007, 7-9 Nov. 2007). Steganography using block-based adaptive threshold. Paper presented at the Computer and information sciences, 2007. iscis 2007. 22nd international symposium on.

[19] Ramadhan J. Mstafa - University of Zakho and Christian Bach - University of Bridgeport , "Information Hiding in Images Using Steganography Techniques" , 2013.

[20] J. Talbot and D. Welsh, "Complexity and Cryptography," pp. 1–9, 2006.

[21] A. Hasan, "Computer Security," 2010.

[22] Sarciszewski, "Guide to Cryptography," 2015.

[23] E. R. Harold, "What is an Image," 2006.