CHAPTER 1

INTRODUCTION

1.1 AUDIO STEGANOGRAPHY:

Data transmission in public communication system is not secure because of interception and improper manipulation by eavesdropper. So the attractive solution for this problem is Steganography,which is the art and science of writing hidden messages in such a way that no one, apart from the sender and intend recipient, suspects the existence of the message, a form of security through obscurity. Audio Steganography is the scheme of hiding the existence of secret information by concealing it into another medium such as audio file.

In cryptography, the structure of a message is scrambled to make it meaningless and unintelligible unless the decryption key is available. It makes no attempt to disguise or hide the encoded message.

Cryptography offers the ability of transmitting information between persons in a way that prevents a third party from reading it. Cryptography can also provide authentication for verifying the identity of someone or something [3]. In steganography does not alter the structure of the secret message, but hides it inside a cover image so that it cannot be seen. A message in a cipher text, for instance, might arouse suspicion on the part of the recipient while an "invisible" message created with steganographic methods will not. In other word, steganography prevents an unintended recipient from suspecting that the data exists. In addition, the security of classical steganography system relies on secrecy of the data encoding system.

Once the encoding system is known, the steganography system is defeated. The trend towards electronic communications and humans desire to conceal messages from curious eyes. With rapid advancement in technology, steganographic software is becoming effective in hiding information in image, video, audio or text files

Audio Steganography hides the secret message in an audio signal called cover audio. Once the secret message is embedded in the cover audio, the resulting message is called stego message and stego message is transmitted to the receiver side. For any audio steganographic technique to be implementable it must satisfy three conditions [1]:

- ✓ **Capacity** means the amount of secret information that can be embedded within the host message
- ✓ **Transparency** evaluates how well a secret message is embedded in the cover audio
- ✓ **Robustness** measures the ability of secret message to withstand against attacks

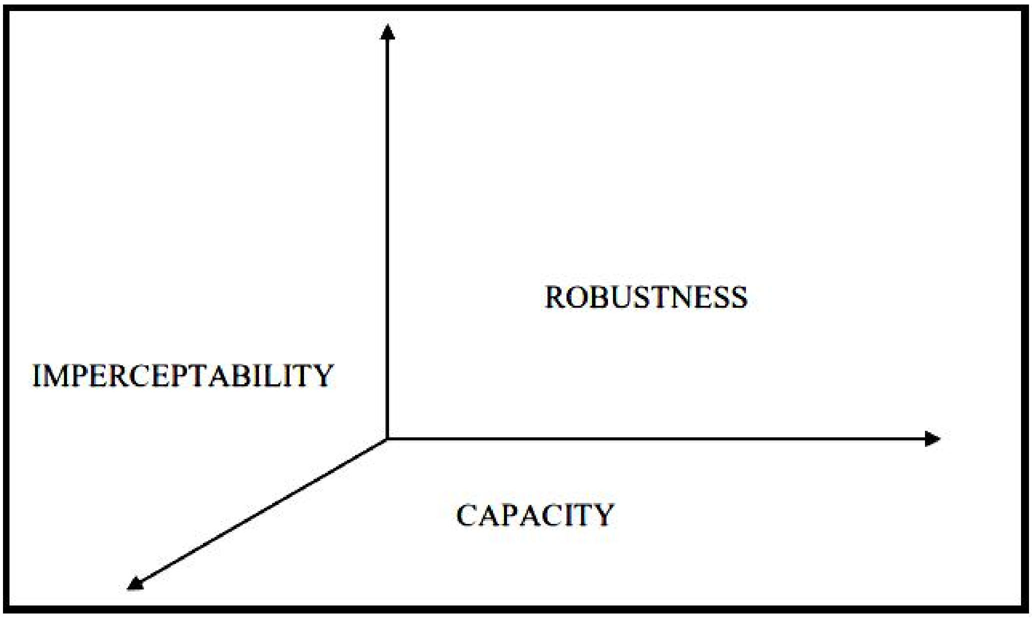


figure 1.1 The tradeoffs between imperceptability, capacity and robustness.

The basic block diagram of a steganographic system is shown in Fig.1. The secret message to be transmitted is embedded inside a cover file. A stego key is also used to provide security. Using a suitable embedding algorithm secret message is embedded into the carrier object. The resultant file is called stego file and this stego file is transmitted to the receiver side.

At the receiver side stego file is decoded using the stego key to extract the secret message. In the case of an audio steganographic system cover file is an audio file. At present, there is lot of research is being made on audio steganography.

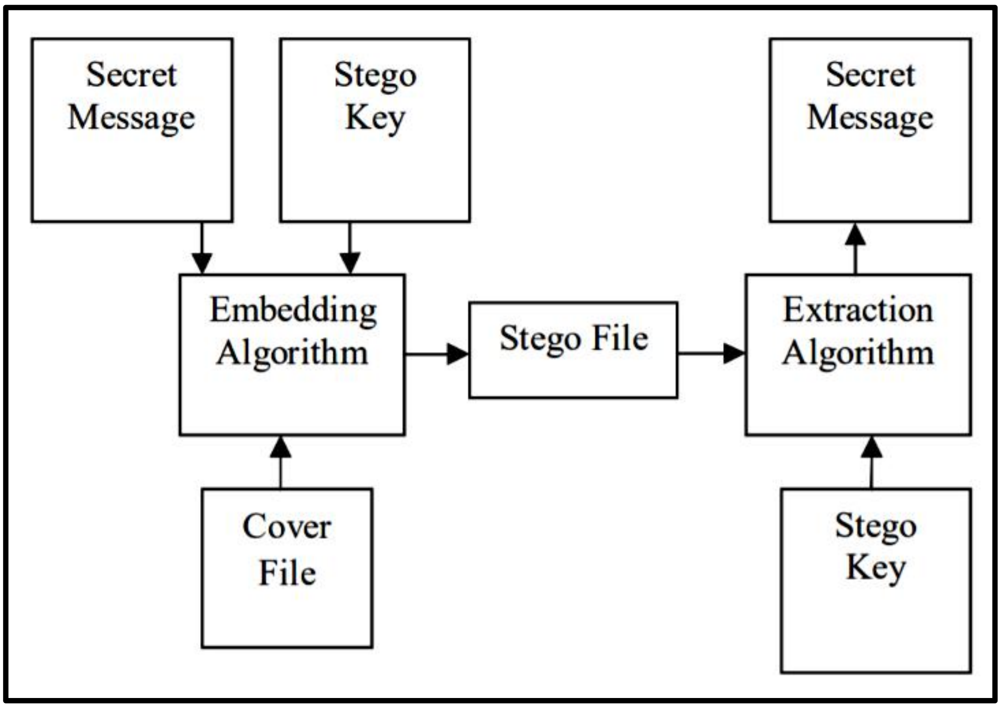
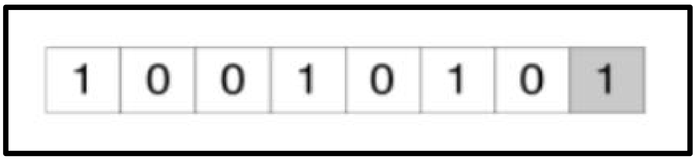


Figure 1.2 Steganography system

1.2 LSB CODING:

LPCM(Linear Pulse Coded Modulation) produces a fully discrete representation of the input signal that can be easily encoded as digital data for storage or manipulation. We exploit this ability of converting the audio file (cover media) into array of digital values for further manipulation using LPCM in the process of steganography. By accessing the audio file in its digital form(discrete form) we can access each sample's digital value and its equivalent binary word.

When we change the LSB(Least Significant Bit) of a word corresponding to a sample in the digital audio file, not much difference is perceived by human ears between the original and the encrypted audio file when reconstructed into analog format by demodulation(or digital to analog conversion in this case). This is one of the simplest methods of hiding data in steganography.

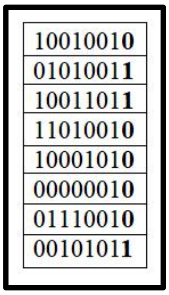


1	0	0	1	0	1	0	1

Figure 1.3 1 bit of message in 1 word sample of audio

The application decoding the cover reads the eight Least Significant Bits of those bytes to recreate the hidden byte—that is 0110001—the letter "a." As you may realize, using this technique let you hide a byte every eight bytes of the cover. Note that there's a fifty percent chance that the bit you're replacing is the same as its replacement, in other words, half the time, the bit doesn't change, which helps to minimize quality degradation.

For example, to hide the letter "a" (ASCII code 97, which is 01100001) inside eight bytes of a cover, you can set the LSB of each byte like this:



10010010
01010011
10011011
11010010
10001010
00000010
01110010
00101011

Figure 1.4 8:1 byte ratio for LSB substitution method

1.3 CAESAR CIPHER:

The Caesar cipher is one of the earliest known and simplest ciphers. It is a type of substitution cipher in which each letter in the plaintext is 'shifted' a certain number of places down the alphabet. For example, with a shift of 1, A would be replaced by B, B would become C, and so on.

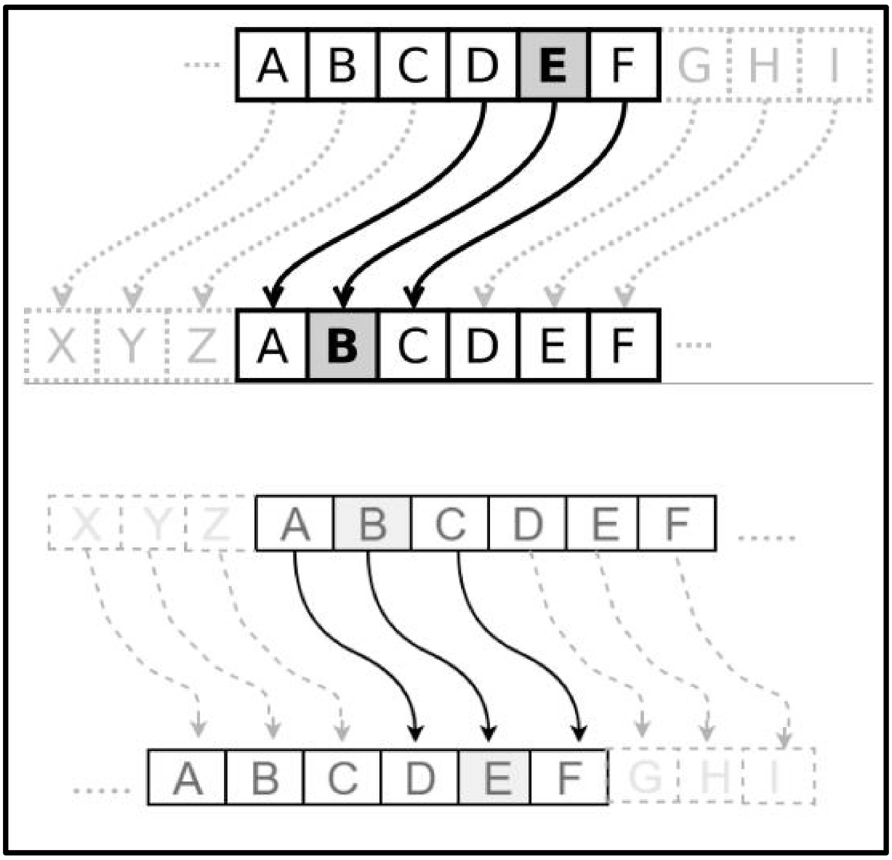


Figure 1.5 CAESAR CIPHER in action

1.4 OTHER METHODS USED IN STEGANOGRAPHY:

- ✓ PARITY CODING
- ✓ PHASE CODING
- ✓ SPREAD SPECTRUM
- ✓ ECHO HIDING

1.5 GUI(GRAPHICAL USER INTERFACE) USING PYTHON 3.7.2:

In software architecture, there may be many layers between the hardware and end user. Each can be spoken of as having a front end and a back end. The front is an abstraction, simplifying the underlying component by providing a user-friendly interface, while the back usually handles business logic and data storage. Tk/Tcl has long been an integral part of Python. It provides a robust and platform independent windowing toolkit, that is available to Python programmers using the tkinter package. The tkinter package is a thin object-oriented layer on top of Tcl/Tk. To use tkinter, you don't need to write Tcl code, but you will need to consult the Tk documentation, and occasionally the Tcl documentation.

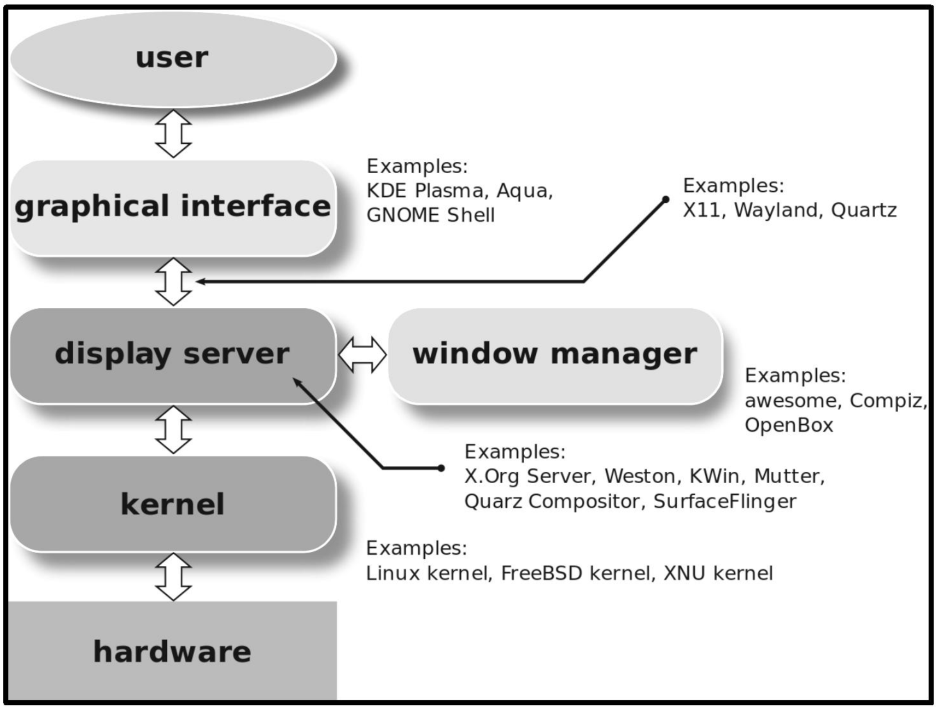


Figure 1.6 GUI role in computation of programs

Tkinter is a set of wrappers that implement the Tk widgets as Python classes. In addition the internal module tkinter provides a thread safe mechanism which allows Python and

Tcl to interact. Tlinter’s chief virtues are that it is fast, and that it usually comes bundled with Python. Although its standard documentation is weak, good material is available, which includes: references, tutorials, a book and others. tkinter is also famous for having an outdated look and feel, which has been vastly improved in Tk 8.5.

CHAPTER 2

LITRATURE SURVEY

2.1 SCIVERSE SCIENCE DIRECT PUBLICATION:

M.Baritha Beguma Y.Venkataramanib, "LSB Based Audio Steganography Based On Text Compression," SciVerse Science Direct, Procedia Engineering 30 (2012) 703 – 710, Saranathan College of Engineering Trichy 620012,India, 2012

2.1.1 CONCLUSION OF THIS PAPER:

This paper proposes a method of text transformation using Dictionary based encoding and audio steganography. In a channel, the reduction of transmission time is directly proportional to the amount of compression. If the input text is replaced by variable length codes with its length less than its average size, the size of input text can be reduced by using dictionary based compression. This proposed compression algorithm achieves good compression ratio, reduces bits per character. This audio Steganography is conducted for various compression algorithms with dictionary based compression. Audio Steganography based text compression achieves better SNR value.

2.1.2 IDEA DERIVED:

the idea of using audio as the cover media for steganography and the idea to use the error metrics such as SNR PSNR and MSE has been used.

2.2 ICIIP PUBLICATION:

V. Sharma and R. Thakur, "LSB modification based Audio Steganography using Trusted Third Party Key Indexing method" 2015 Third International Conference on Image Information Processing (ICIIP), Waknaghat, 2015, pp. 403-406. doi: 10.1109/ICIIP.2015.7414805

2.2.1 CONCLUSION OF THIS PAPER:

This paper proposes Random Key Indexing method to replace the LSBs of the carrier audio with secret message. The bit replacement is guided by primary key that is provided by TTP (Trusted Third Party) and secondary key that will be generated at encoder end during embedding process and is supplied to the decoder end. The proposed method also uses message retrieval code that adds another layer of protection to the process. The method is successfully tested on various 32 bit & 16 bit stereo wave files with different payloads. The SNRdB values comes out be in the range 139 dB to 142 dB for 32 bit and 67 dB to 85 dB for 16 bit stereo files. The Bit Error Rate (BER) comes out be in the range 0.23 to 0.32 percent for 32 bit and 0.018 to 0.028 percent for 16 bit files.

2.2.2 IDEA DERIVED:

the idea of LSB substitution has been used.

2.3 ISCMI PUBLICATION:

A. Binny and M. Koilakuntla, "Hiding Secret Information Using LSB Based Audio Steganography," 2014 International Conference on Soft Computing and Machine Intelligence, New Delhi, 2014, pp. 56-59.doi: 10.1109/ISCMI.2014.24

2.3.1 ABSTRACT OF THIS PAPER:

Audio signals have a characteristic redundancy and unpredictable nature that make them ideal to be used as a cover to hide secret information. A Steganographic technique for embedding text information in audio using LSB based algorithm is presented in this paper. In the proposed method each audio sample is converted into bits and then the text data is embedded. In embedding process, first the message character is converted into its equivalent binary. By using proposed LSB based algorithm, the capacity of stego system to hide the text increases. The performance of the proposed algorithm is computed using SNR values for various audio input.

2.3.2 IDEA DERIVED:

the idea of taking SNR(Signal To Noise Ratio) for validation has been referred.

CHAPTER 3

SCOPE OF THE PRESENT WORK

The amount of information that companies must communicate securely is increasing. As a result of technological advances, companies are constantly gaining more data about their clients and customers. They must ensure that data security and privacy remain a priority to protect against breaches

Banks are one of the best example considered to analyze the threat underlying in the process of communication.Conventional control techniques are not secure enough for transmissions that involve highly sensitive information like bank authentication and military communication.

Steganograhpic technique comes into picture when transmission of highly sensitive information is concerned.The amount of data to be encrypted however depends on the length of audio and the sampling rate of the digital computer's ADC(Analog to Digital Converter) and corresponding DAC(Digital to Analog Converter).

Steganalysis techniques are available to crack weak steg systems. There fore the so steg system must be strong in a way that the stego object is not vulnerable to steganalysis By increasing the entrophy of the message to be transmitted before the process of steganography the strength of the process of encryption is increased several folds when compared to usage of steganography solely.

Another limitation is that the size of the data to be transmitted is the size of the original data scaled by the the format of the cover file. For instance, when the audio used is in 8 bit word length format, the size of the data to be transmitted is 8 times as the original data itself. It is often the trade-off between the level of security and the feasibility of the process that is to be considered while transmitting a message from one end of the communication to the other end

CHAPTER 4

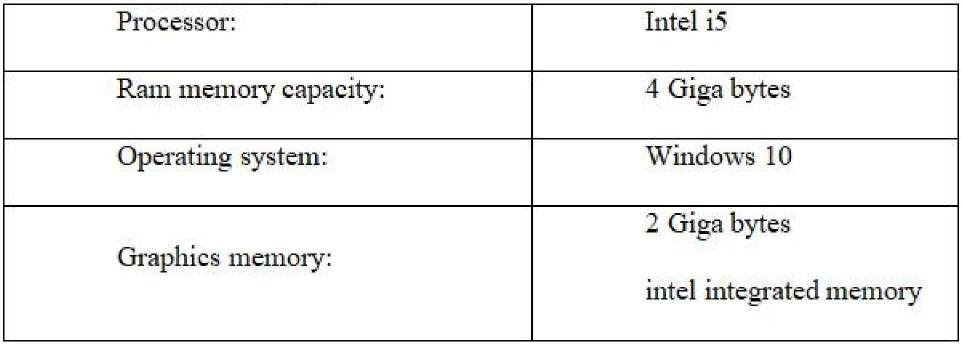
EXPERIMENTAL PROCEDURES

4.1 RESOURCES UTILIZATION:

4.1.1 HARDWARE COMPONENTS:

A computer with minimum configuration to support *Python4.7.2* and support ‘*.wav’* audio file format is the primary hardware requirement. The configuration of the hardware used for this project is as follows:

Table 3.1 hardware properties



Processor:	Intel i5
Ram memory capacity:	4 Giga bytes
Operating system:	Windows 10
Graphics memory:	2 Giga bytes intel integrated memory

4.1.2 SOFTWARE COMPONENTS:

PYTHON 4.7.2 programming language package is used in the for the purpose of realizing the process of steganography in a digital platform such as a computer.Both the GUI(Graphical User Interface) which is the front end and the steganography system algorithm at the back end of the project were developed using this programming. The built-in library package gives us the advantage of maximum possible abstraction at a code level for a user friendly programming experience.

This project makes use of the *‘tkinter’ , ‘wave’ , ‘struct’ , ‘pygame’* and *‘numpy‘* package appreciating the inbuilt functions for *‘.wav*’ audio file format to ‘*numpy array’* format and further to binary file format the reverse conversion of the entire formatting after embedding of the secret message .

4.2 PROGRAM FLOW CHART:

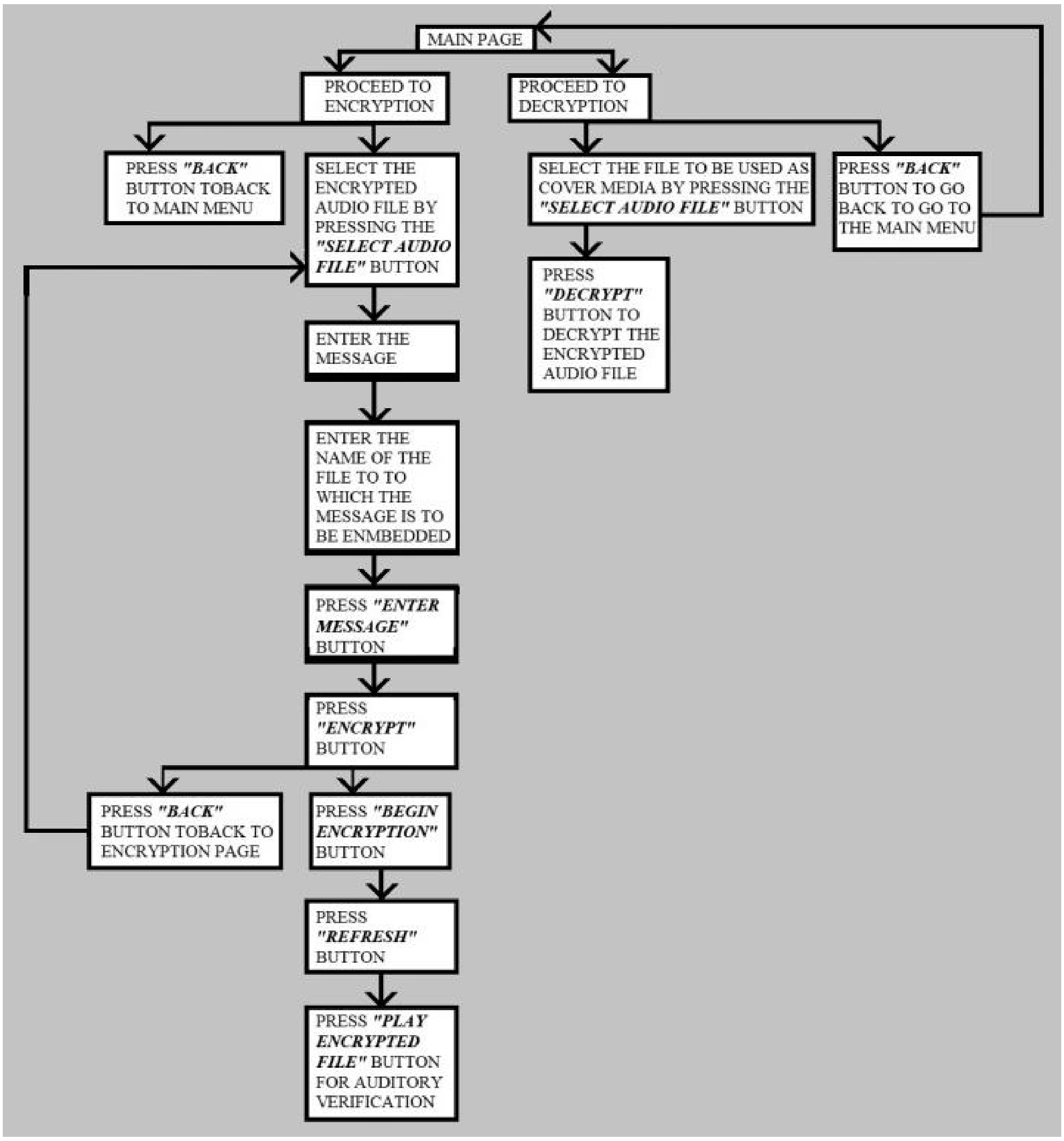

MAIN PAGE
PROCEED TO ENCRYPTION
PROCEED TO DECRYPTION
PRESS "BACK" BUTTON TOBACK TO MAIN MENU
SELECT THE ENCRYPTED AUDIO FILE BY PRESSING THE "SELECT AUDIO FILE" BUTTON
SELECT THE FILE TO BE USED AS COVER MEDIA BY PRESSING THE "SELECT AUDIO FILE" BUTTON
PRESS "BACK" BUTTON TO GO BACK TO GO TO THE MAIN MENU
PRESS "DECRYPT" BUTTON TO DECRYPT THE ENCRYPTED AUDIO FILE
ENTER THE MESSAGE
ENTER THE NAME OF THE FILE TO TO WHICH THE MESSAGE IS TO BE ENMBEDDED
PRESS "ENTER MESSAGE" BUTTON
PRESS "ENCRYPT" BUTTON
PRESS "BACK" BUTTON TOBACK TO ENCRYPTION PAGE
PRESS "BEGIN ENCRYPTION" BUTTON
PRESS "REFRESH" BUTTON
PRESS "PLAY ENCRYPTED FILE" BUTTON FOR AUDITORY VERIFICATION

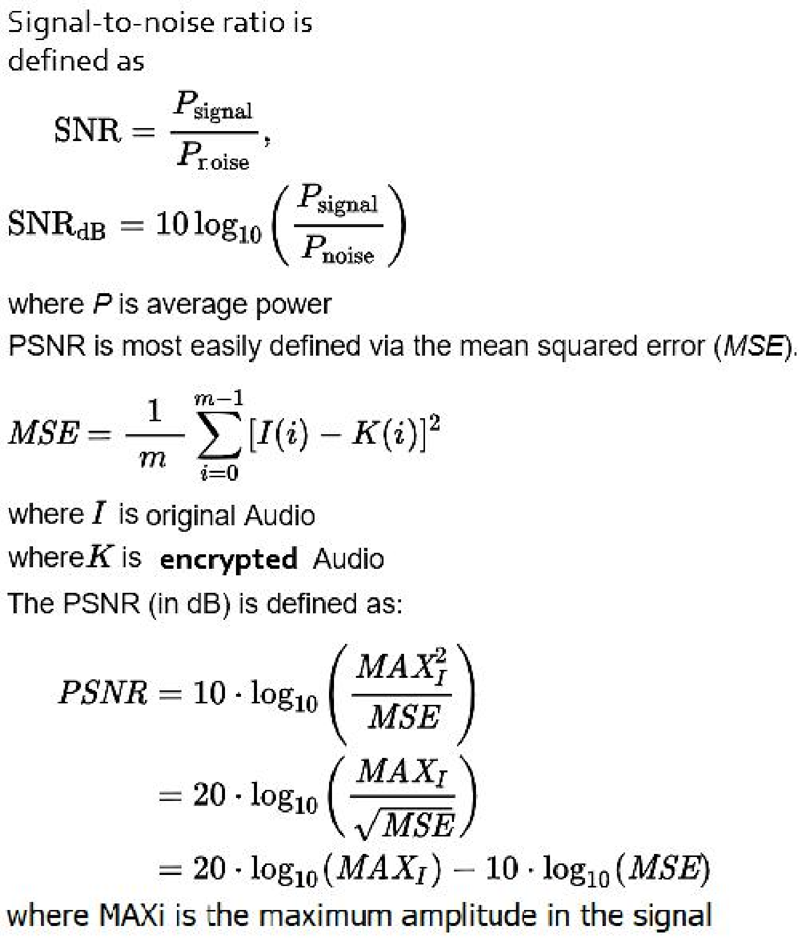
Figure 4.2.1 flow chart

4.3 STEGANOGRAPHY VALIDATION:

By calculating the change in the stego object(audio file) with respect to the original cover audio one can easily validate the process of steganography. This can be done by calculating the following parameters using there corresponding formulas between the two audio medias:

- ✓ SNR(Signal To Noise Ratio)
- ✓ PSNR(Peak Signal To Noise Ratio)
- ✓ MSE(Mean Square Error)

The formulas being:



Signal-to-noise ratio is defined as

$$\mathrm{SNR} = \frac{P_{\mathrm{signal}}}{P_{\mathrm{r.oise}}},$$

$$\mathrm{SNR_{dB}} = 10\log_{10}\left(\frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}}\right)$$

where P is average power

PSNR is most easily defined via the mean squared error (*MSE*).

$$MSE = \frac{1}{m}\sum_{i=0}^{m-1}[I(i) - K(i)]^2$$

where I is original Audio

where K is **encrypted** Audio

The PSNR (in dB) is defined as:

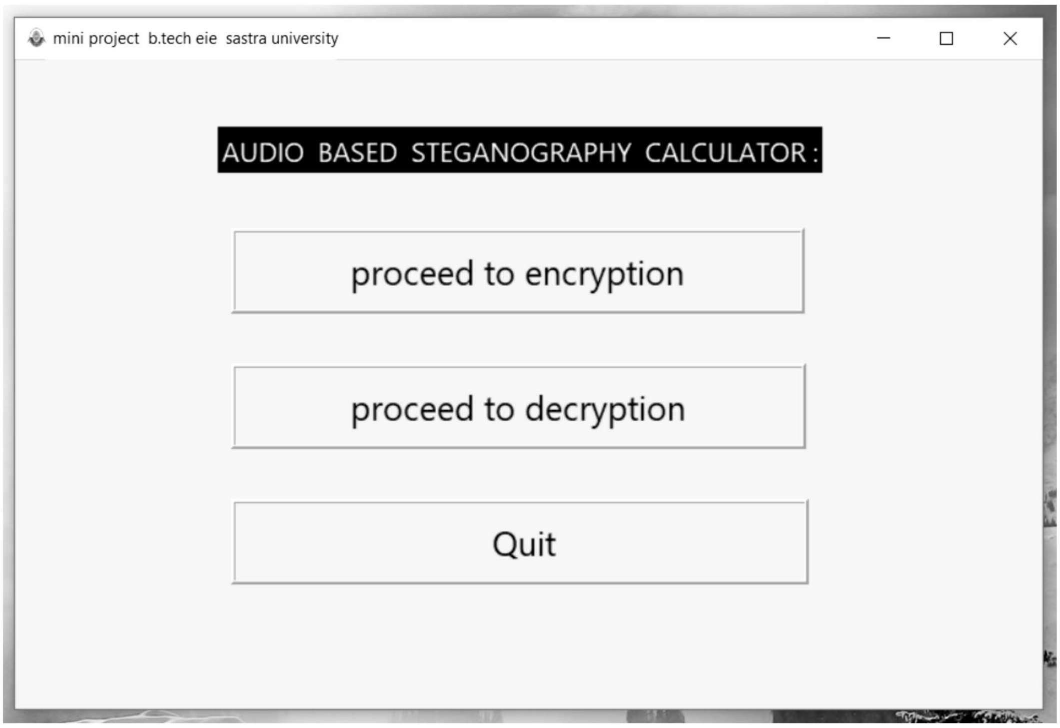
$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned}$$

where MAXi is the maximum amplitude in the signal

CHAPTER 5

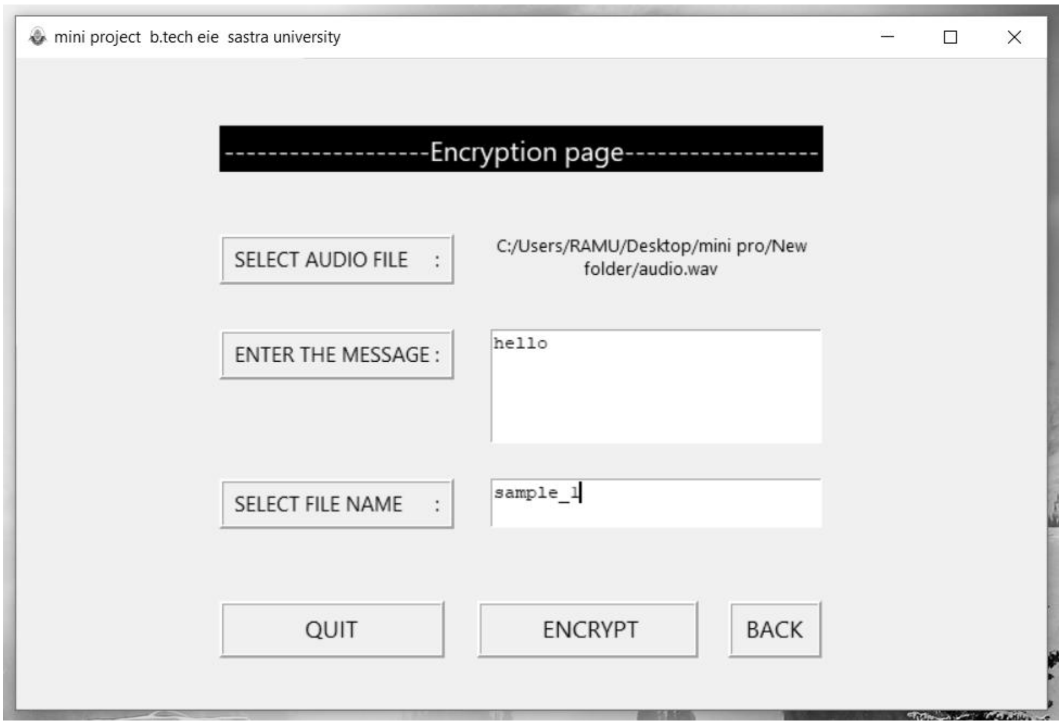
RESULTS AND DISCUSSION

5.1 SCREENSHOT OF GUI:



mini project b.tech eie sastra university
AUDIO BASED STEGANOGRAPHY CALCULATOR :
proceed to encryption
proceed to decryption
Quit

Figure 5.1.1 main menu



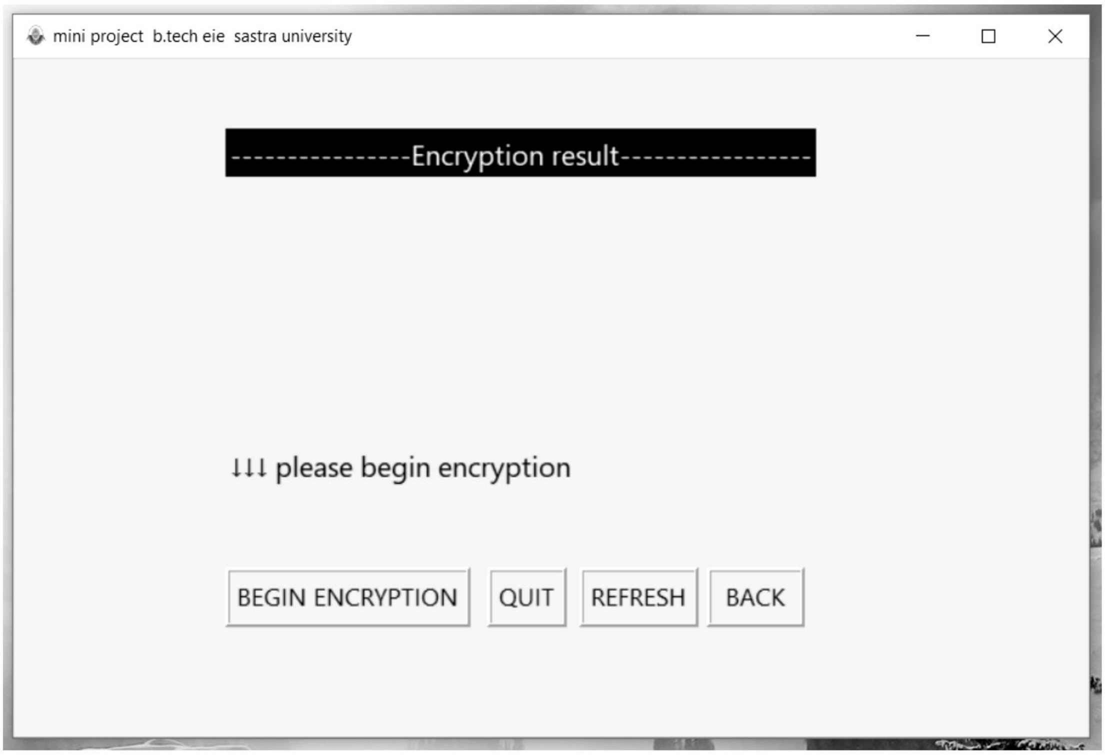
mini project b.tech eie sastra university
-------------------Encryption page-----------------
QUIT
ENCRYPT
BACK

SELECT AUDIO FILE : C:/Users/RAMU/Desktop/mini pro/New folder/audio.wav

ENTER THE MESSAGE : hello

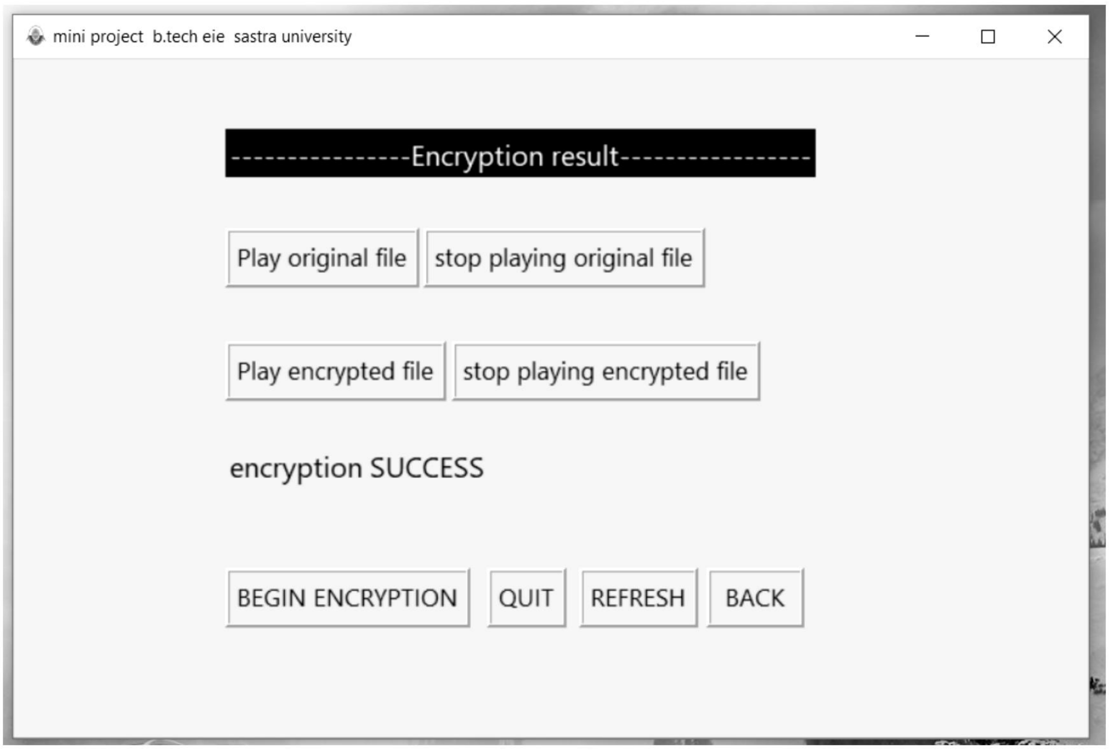
SELECT FILE NAME : sample_1

Figure 5.1.2 encryption page


mini project b.tech eie sastra university
---------------Encryption result----------------
BEGIN ENCRYPTION
QUIT
REFRESH
BACK

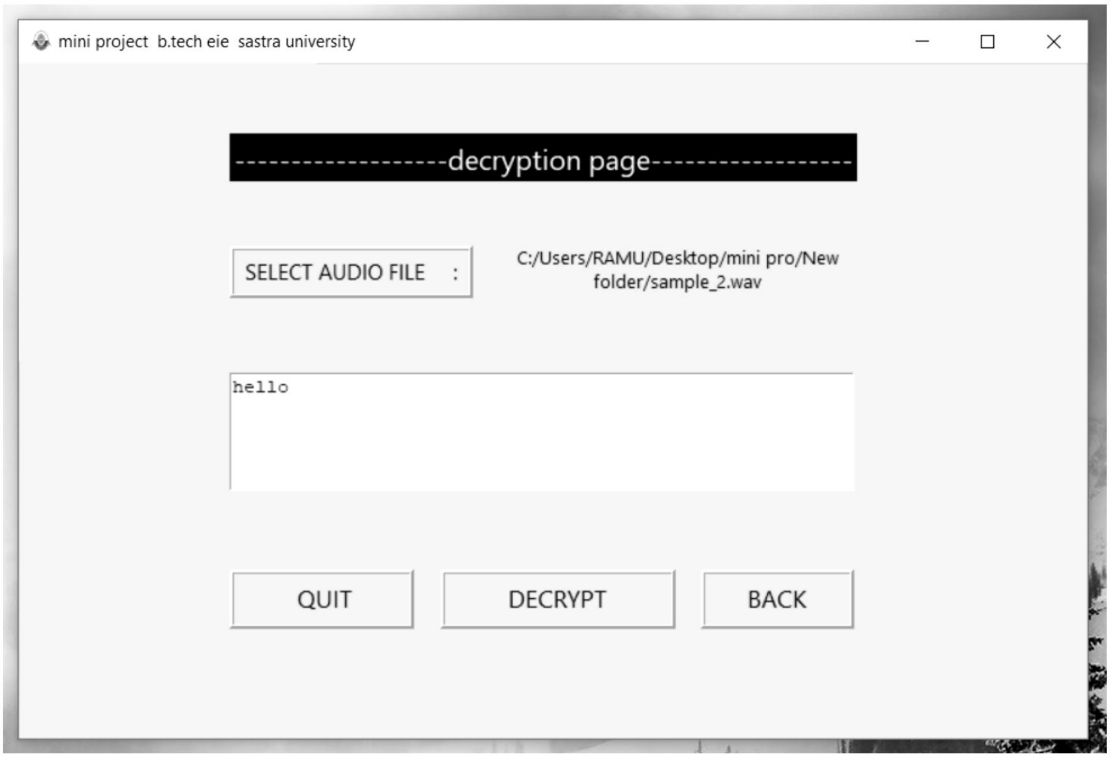
↓↓↓ please begin encryption

Figure 5.1.3 encryption result


mini project b.tech eie sastra university
---------------Encryption result----------------
Play original file
stop playing original file
Play encrypted file
stop playing encrypted file
BEGIN ENCRYPTION
QUIT
REFRESH
BACK

encryption SUCCESS

Figure 5.1.4 encryption result after encryption


mini project b.tech eie sastra university
-------------------decryption page------------------
SELECT AUDIO FILE :
QUIT
DECRYPT
BACK

C:/Users/RAMU/Desktop/mini pro/New folder/sample_2.wav

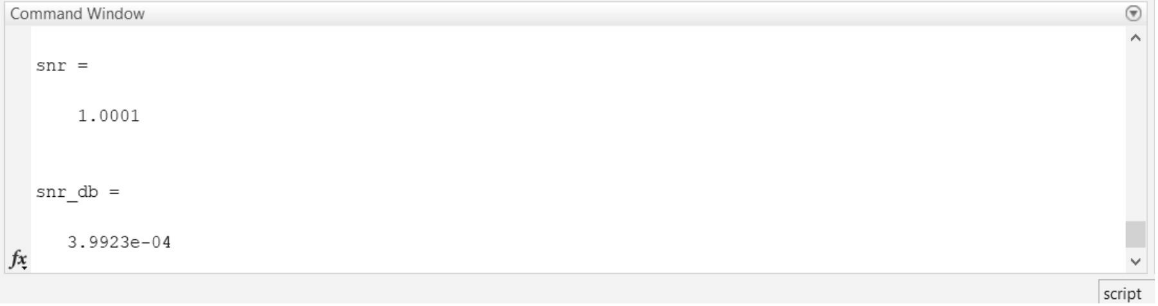
hello

Figure 5.1.5 decryption page with result

5.2 STEGANOGRAPHY SYSTEM VALIDATION RESULT:

5.2.1 SNR:

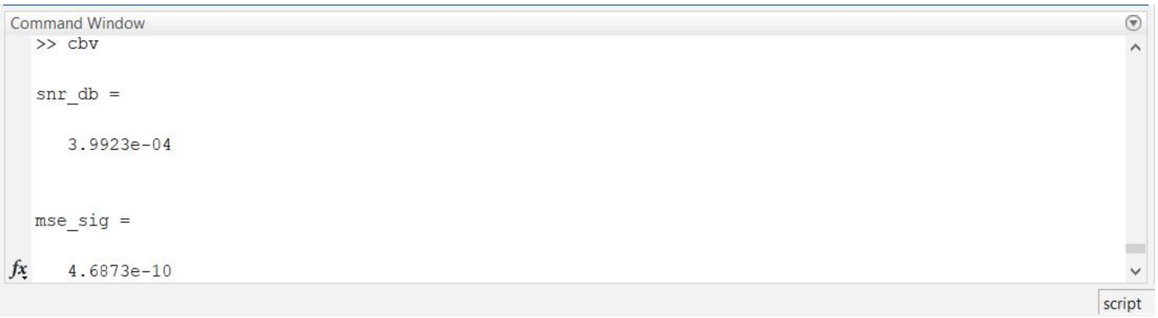
Matlab program was utilized to calculate snr, psnr and mse value:



```
snr =

    1.0001


snr_db =

   3.9923e-04
```

Figure 5.2.1 SNR

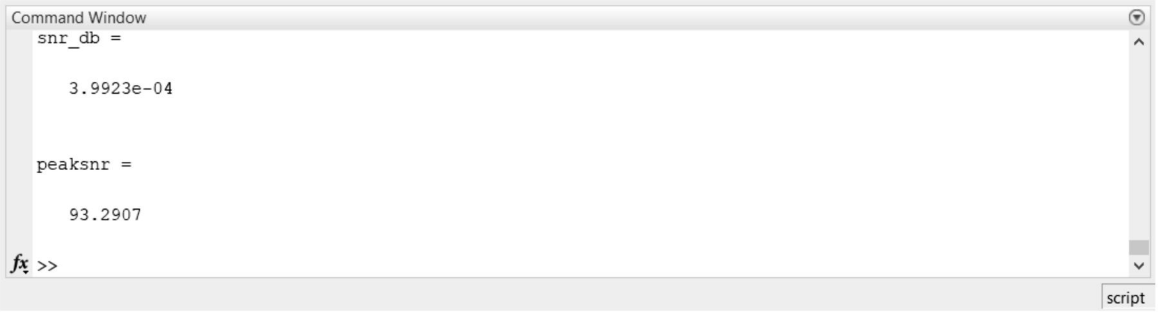
5.2.2 MSE:



```
>> cbv

snr_db =

   3.9923e-04


mse_sig =

   4.6873e-10
```

Figure 5.2.2 MSE

5.2.3 PSNR:



```
snr_db =

   3.9923e-04


peaksnr =

   93.2907

>>
```

Figure 5.2.3 PSNR

CHAPTER 5

CONCLUSIONS

Using the technique of steganography is one of the most highly secure way of transmitting data without even a dust-amount of suspicion on the way of transmission the credits also being added to Caesar cipher. This fact was verified both mathematically and by auditory verification. GUI being the highest level of abstraction is a proven fact to be mush easier to handle when compared to its counterpart, the programming terminal(python 3.7.2 ILDE in this case). On the whole the project was successfully accomplished with no implicit or explicit singularities.