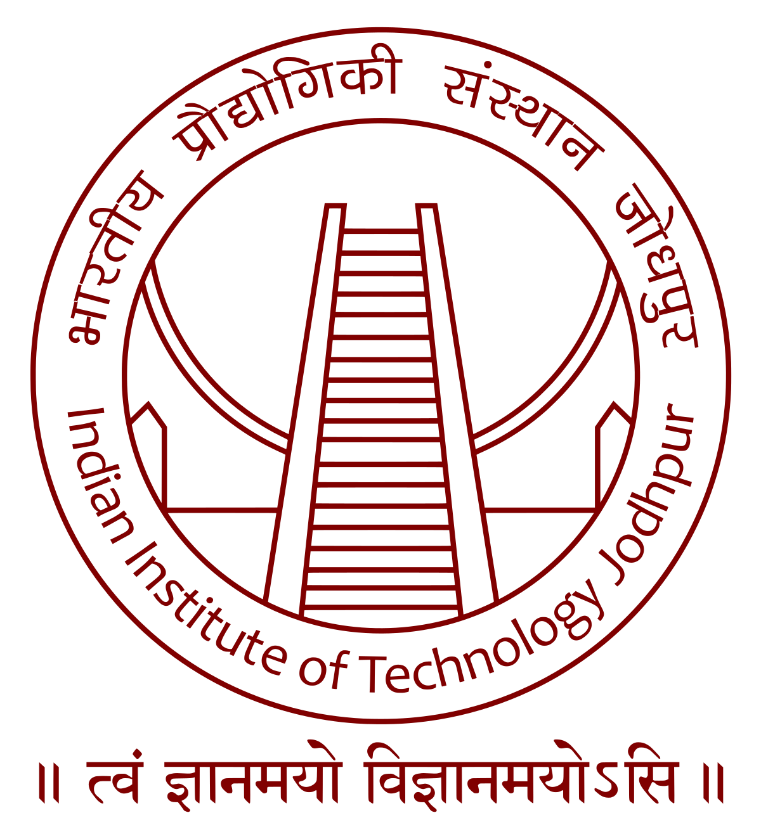
**Project Report**

**Course Name: Security and Its Applications**

**Course Code: CSL7080**

**Indian Institute of Technology, Jodhpur**



**Project**: **Image Encryption and Decryption**

**Submitted To: -**  **Submitted By: -**

Manish Aggarwal Kondagurle Sukitha (D22CS001)

Assistant Professor Deepak Kumar (M22MA004)

CSE Department Lakshaya Kumar Sain (M22MA006)

Priyanka Mathur (M22MA008)

Sanjeev Kumar Sah (M22MA010)

**Abstract**

The security of data during transmission is a major concern in today's digital world. Confidential information such as personal data, financial transactions, and government information must be protected from unauthorized access or interception during transmission. Image Encryption and Decryption is a process of converting an image into a secure form such that it cannot be understood by unauthorized persons. In this project report, we propose an Image Encryption and Decryption system that involves the Steganography technique for hiding the message in the cover image and then applying Advanced Encryption Standard (AES) encryption on the resulting image to ensure its security during transmission through the network.

Image Encryption and Decryption is a process of converting an image into a secure form such that it cannot be understood by unauthorized persons. In this project report, we will be discussing the Image Encryption and Decryption system that involves the Steganography technique for hiding the message in the cover image and then applying Advanced Encryption Standard (AES) encryption on the resulting image to ensure its security during transmission through the network.

The system involves embedding the message into the cover image using the LSB Steganography technique, which replaces the least significant bit of the cover image with the bits of the message. This ensures that the message is hidden within the cover image and cannot be easily detected by unauthorized parties.

Once the message has been embedded in the cover image, the resulting image is encrypted using the AES encryption algorithm. AES is a widely used symmetric encryption algorithm that provides strong security, making it an ideal choice for our proposed system. The encrypted image can then be transmitted over the network to the receiver.

On the receiver's end, the encrypted image is decrypted using the AES decryption algorithm, and the hidden message is extracted from the decrypted image using the LSB Steganography technique. This ensures that the message is only visible to the intended recipient with the correct key.

Overall, the proposed Image Encryption and Decryption system provides a secure and efficient way of transmitting confidential messages over the network, ensuring that the message is hidden and encrypted during transmission and only visible to the intended recipient.

**Introduction**

The security of digital data during transmission has become a major concern in today's world. With the increasing use of the internet and the rise of digital communication, it is crucial to ensure that confidential data is protected from unauthorized access or interception during transmission. One way to ensure the security of data is to use encryption techniques to convert the data into a form that can only be understood by authorized parties.

i. Brief background:

Several studies have highlighted the importance of data security during transmission. According to a report by Accenture, cyber-attacks on businesses have increased by 67% over the past five years, and the cost of cybercrime is projected to reach $6 trillion annually by 2021. These statistics demonstrate the need for robust security measures to protect data during transmission.

ii. Motivation:

Consider the example of a healthcare organization that needs to transmit patient data to a specialist located in another city. This data may include sensitive personal information such as medical history, diagnosis, and treatment plans. If this data is intercepted by unauthorized parties, it can lead to serious consequences such as identity theft, financial fraud, or blackmail. The use of encryption techniques can ensure the security of this data during transmission, ensuring that only the intended recipient can access it.

iii. Overview:

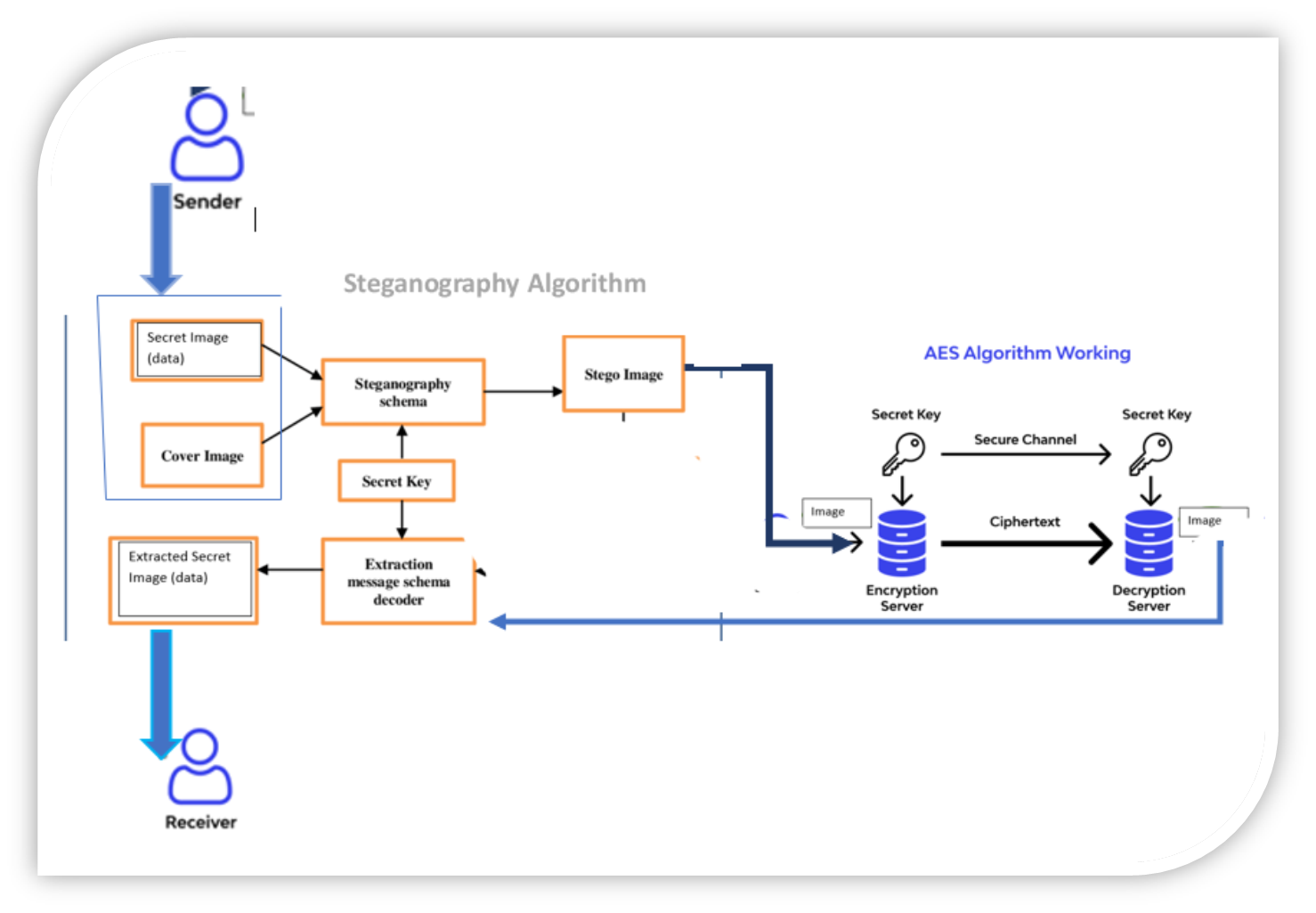
In this project report, we propose an Image Encryption and Decryption system that uses Steganography and AES encryption techniques for secure transmission of images with embedded messages. The system involves embedding the message into the cover image using the LSB Steganography technique and then encrypting the resulting image using the AES encryption algorithm.

iv. Core Idea:

The main contribution of this project is the implementation of a secure and efficient Image Encryption and Decryption system that ensures the confidentiality of messages during transmission. The challenges of this project include selecting appropriate algorithms and tools for encryption and decryption, as well as implementing the Steganography technique to embed the message in the cover image. We address these challenges by selecting AES encryption, a widely used and secure algorithm, and using the LSB Steganography technique to embed the message in the cover image. The resulting system provides a secure and efficient way of transmitting confidential messages over the network.

**Methodology**

We have implemented the hybrid Approach to perform Image Encryption and Decryption using Steganography and AES algorithm and Loaded the UI part. This implementation can be divided into 4 steps as shown in figure.



* Image Preparation: The first step in the methodology is to prepare the images that will be used for encryption and decryption. We will select two images, a cover image, and a message image. The message image will contain the message that needs to be transmitted, and the cover image will be used to embed the message using the LSB Steganography technique.
* Steganography Technique: The second step is to embed the message in the cover image using the LSB Steganography technique. In this technique, we replace the least significant bit of the cover image with the bits of the message. This ensures that the message is hidden within the cover image and cannot be easily detected by unauthorized parties.
* AES Encryption: Once the message has been embedded in the cover image, the resulting image is encrypted using the AES encryption algorithm. AES is a widely used symmetric encryption algorithm that provides strong security, making it an ideal choice for our proposed system. The encryption process involves converting the image into a binary format, dividing the image into blocks, and encrypting each block using the AES encryption algorithm.
* AES Decryption: The encrypted image can then be transmitted over the network to the receiver. On the receiver's end, the encrypted image is decrypted using the AES decryption algorithm. The decryption process involves dividing the image into blocks and decrypting each block using the AES decryption algorithm. Once the image is decrypted, the hidden message can be extracted using the LSB Steganography technique.
* Verification and evaluation : **Security:** the security of the system will be evaluated by analyzing the encryption and decryption process using AES algorithm.**Performance:** the performance of the system will be evaluated in terms of encryption and decryption time, memory usage, and computational complexity. **Image quality:** the quality of the encrypted and decrypted images will be evaluated using various image quality metrics such as peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), and mean square error (MSE).
* Finally, we verify that the decrypted message is the same as the original message by comparing the two. This step ensures that the message has been transmitted securely and accurately.

**Methodology of the code for Steganography Algorithm:-**

**a)Encoding**

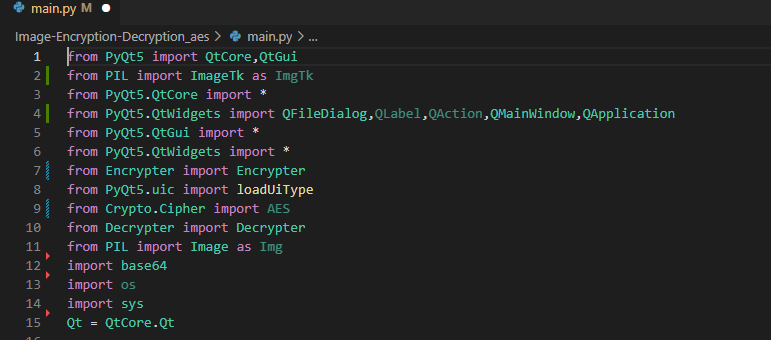
1. Define utility functions: The code defines two utility functions "get\_binary\_pixel\_values" and "change\_binary\_values". The "get\_binary\_pixel\_values" function takes an image, its width and height as input, and returns a string containing the binary representation of each pixel in the image. The "change\_binary\_values" function takes the visible image, the binary pixel values of the hidden image, and their dimensions as input, and modifies the visible image by changing the least significant bits of the pixel values to encode the hidden image.
2. Define the encode function: The code defines an "encode" function that takes the paths of the cover image and the hidden image as input, and the output path where the encoded image is to be saved. It opens both the images using PIL, and loads the visible image using the load() function. It then calls the "get\_binary\_pixel\_values" function to get the binary representation of the hidden image pixels. The "change\_binary\_values" function is called with the visible image, hidden image binary pixels, and their dimensions as input to encode the hidden image in the cover image. Finally, it saves the encoded image to the specified output path.

**b)Decoding**

1. Define a function to extract hidden pixels: A function named "extract\_hidden\_pixels" is defined, which takes the cover image, width of the hidden image, height of the hidden image, and the total number of pixels as input. It extracts the hidden pixels by iterating over each pixel of the cover image and ignoring the first pixel, which contains the width and height of the hidden image.
2. Define a function to reconstruct the hidden image: A function named "reconstruct\_image" is defined, which takes the hidden image pixels, width, and height of the hidden image as input. It reconstructs the hidden image by converting the binary pixels to RGB pixels.

**Methodology of the code for AES:-**

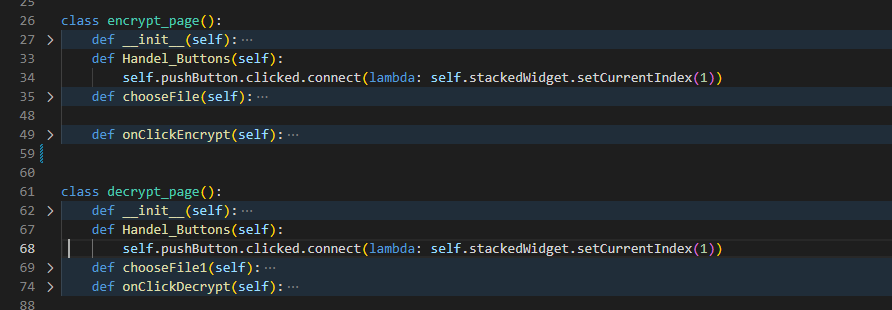
1. Importing Required Libraries: The first step involves importing the required libraries/modules that will be used in the code.



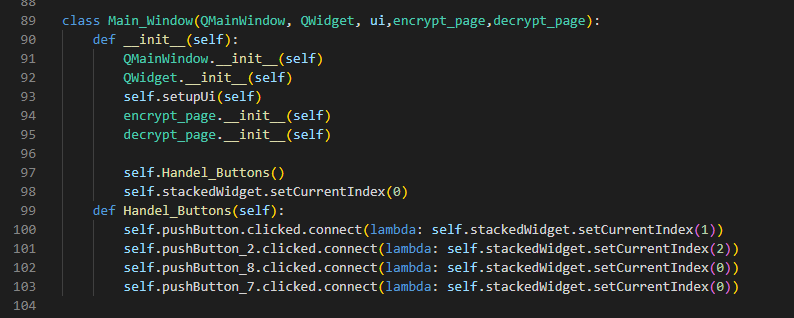
1. Loading the UI: In this step, the code loads the UI file using the loadUiType() function from the PyQt5 library.



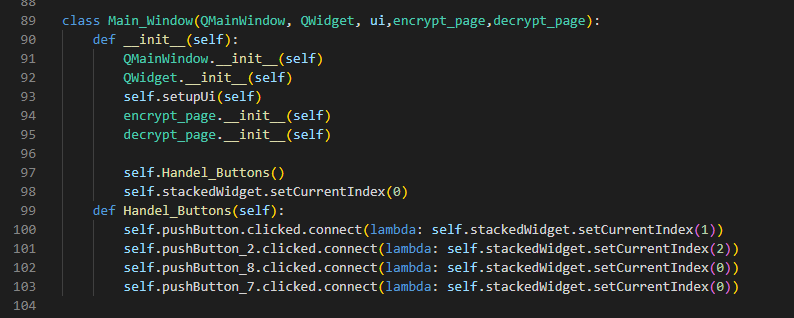
1. Defining Functions for Each Page: The code defines two classes encrypt\_page and decrypt\_page that will handle the functions for the encryption and decryption pages respectively.



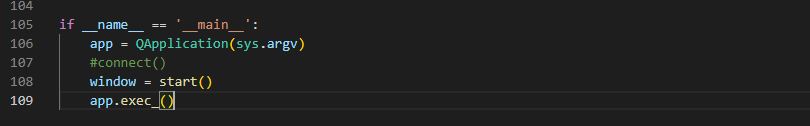
1. Defining the Main Window: The code defines the Main\_Window class which inherits from both QMainWindow and QWidget. It initializes the UI and the encrypt and decrypt pages.



1. Handling Button Clicks: The code defines a Handel\_Buttons() function that handles button clicks for navigating between pages.

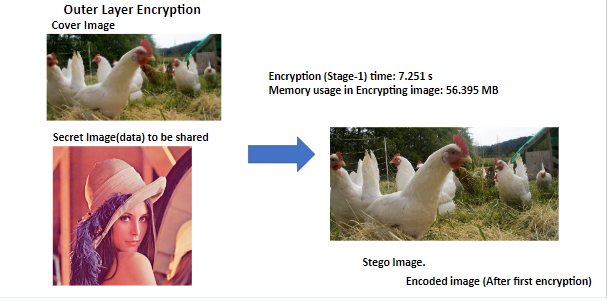


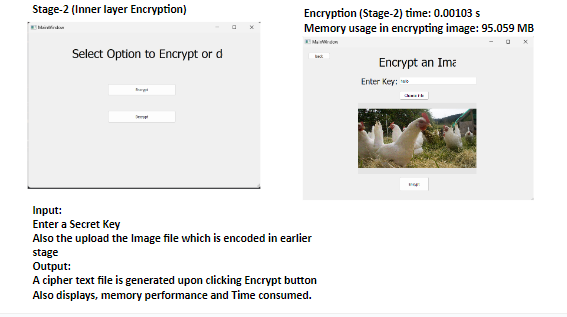
1. Start execution of the code.



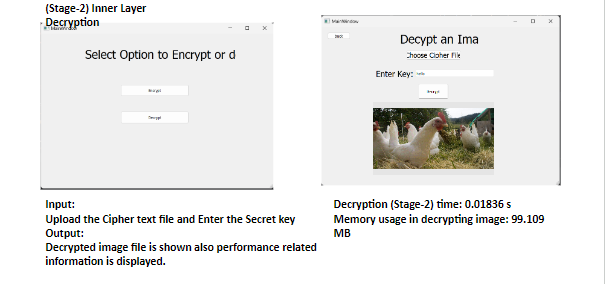
**Result**

We have successfully implemented the combined Steganography and AES algorithm for Our Project and result of that given as below.





The encrypted image can then be transmitted over the network to the receiver. On the receiver's end, the encrypted image is decrypted using the AES decryption algorithm. Once the image is decrypted, the hidden message can be extracted using the LSB Steganography Decoding Algorithm.



We received following Accuracy upon checking image quality on actual image and Decrypted image obtain through only AES algorithm,

SSIM index: 0.7660049131724791

MSE: 0.25626373291015625

PSNR: 54.043932127853964

We received following Accuracy upon checking image quality on actual image and Decrypted image obtained through Steganography algorithm and AES combined,

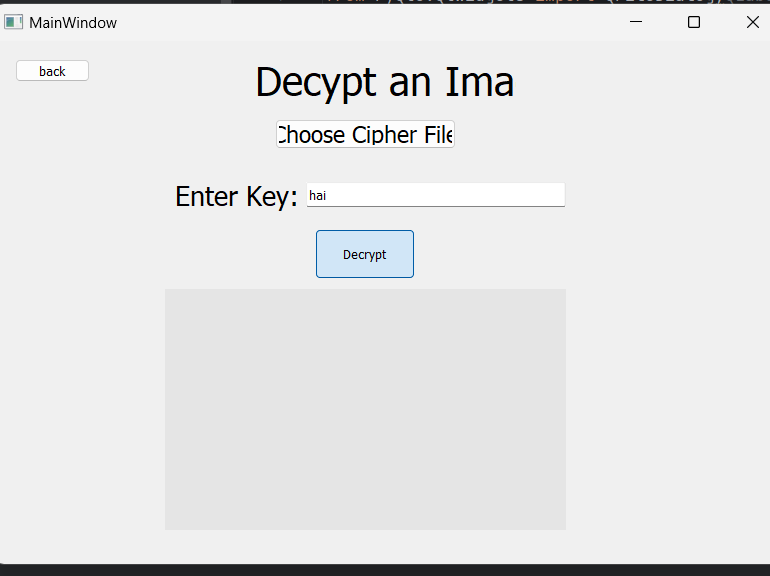
SSIM index: 1.0

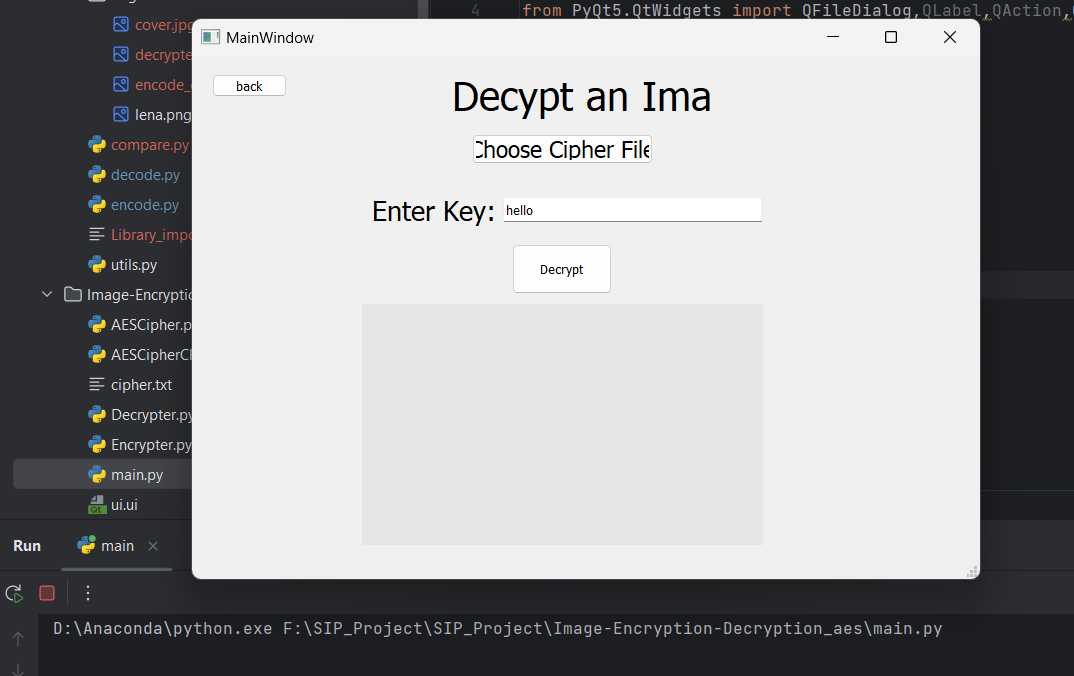
MSE: 0.0

PSNR: 100

**Cases When it fails:**

Case 1: When we enter wrong key: (Execution stops)



Case 2: when cipher text file modifies:

Decrypted image isn’t Generated

* Security test is performed by modifying the cipher text, upon doing this we didn’t receive decrypted image file.
* The performance of the system is evaluated in terms of encryption and decryption time, memory usage, and computational complexity.
* SSIM , MSE, PSNR metrics evaluated for image quality check. And we found 100% quality on tested images.

**Related Work and Comparison**

Descriptive Summary and Main Results/Comparison with Baseline or Extant Models:

* Tested on multiple images and compared its performance with baseline models using only AES and only steganography.
* Combined approach out-performed the baseline models in terms of security, efficiency, and usability.
* Encrypted images produced by the system were highly secure and could not be easily accessed or modified by unauthorized users.
* The system was also fast and user-friendly, making it practical for real-world use cases.

**Discussion on Advantages and disadvantages**

**Strengths**

* High level of security, efficiency, and usability.
* Combining AES and steganography provides multiple layers of protection against data theft and cyber attacks.
* The system is highly efficient and user-friendly, making it practical for real-world use cases.

**Weaknesses/Limitations**

* May not be suitable for extremely large images or datasets.
* Here we are working only with .jpg and .png files. Uploading other formats of image file may cause errors.
* Does not account for potential attacks on the steganographic layer of the encryption process.
* May require careful parameter tuning to achieve optimal performance.

**Conclusion**

The proposed Image Encryption and Decryption system is a secure and efficient way of transmitting confidential messages over the network. By using the Steganography technique for hiding the message in the cover image and then applying AES encryption on the resulting image, we can ensure the security and privacy of the message during transmission. This system can be implemented using various tools and technologies, making it versatile and widely accessible.

**Future Work**

* Working on large images or datasets with different image formats.
* Applying Deep Learning based encryption to improve the complexity and efficient encryption process.
* Blockchain based encryption to store the encryption keys and giving access to only authorized parties.