Any obvious visual difference?
What is steganography

• **Concealing** important data within the carrier medium (video, audio, image, etc.)

• It is to *prevent* others from *thinking* there is secret data hidden in the carrier
Project Goals

• Store more information
• Reuse the image (support multi-user)
• More secure
Design

Let's start from image
Assign different layers to different users

- Support multiple users (up to 3)
- Avoid conflicts
- Reuse image
Design

Minimize the changes

pick one pixel

e.g. 150

0b10010110

Least Significant Bit
Design

• Every input character will be coded as 12 bits

• Layer available indicator 30 bits
  101100111100011111000011111100000

• Length of input will be coded as 20 bits embedded into image as well
Design

More secure

• Input message will be encrypted by RSA
• RSA Keys given by user (doesn’t store any key)
• Length of the input will also be encrypted
Design

Message Embedding

Start

parameter empty?

Yes

Notify user

No

Layer available?

Yes

Length over image size?

Yes

Encrypt input

Embed to image

End

No

No

Message Extraction

Start

parameter empty?

Yes

Notify user

No

Layer unused?

Yes

Extract & Decryption

Key correct?

Yes

Output

End

No
In this project, I will focus my research on image steganography, which is used to hide important data in the carrier medium to avoid detection from attackers while exchange messages. Not like encryption, steganography is not design to keep others knowing the hidden data, instead it is to prevent others from think there is secret data hidden in the carrier medium. That is to say, the goal of Steganography is to store private data in images and at the same time bring minimum changes to the carrier medium itself in order to lower the possibility of being suspected. It is an important technology for message exchange, and really interesting to me.

The existing steganography method can be categorized into two directions: spatial domain and frequency domain. For spatial domain method, people directly manipulate the value of pixels to hide additional data. For example, we can hide secret message into the least significant bits (LSB) of a pixel which is usually the last bit of the pixel. Because the changes of least significant bits would bring small impact on the entire image. It is difficult for people to tell the difference visually. In frequency domain based steganography methods, researchers decomposed carrier image by DCT(Discrete Cosine Transform) into several blocks and then insert secret data into the coefficients’ insignificant bits(based on LSB). The frequency domain methods is very effective for carriers like compressed audio or compressed image such as JPEG format. Due to the compression algorithm, minor changes to the carrier will cause obvious impact on carriers, being easy to detect. Researchers also introduced FFT(Fast Fourier Transform) and DWT(Discrete Wavelet Transform) into frequency domain methods to achieve better performance.

Since our application will allow one user or multiple users to hide information, therefore one problem need to be considered is how to avoid conflicts and guarantee that the information is retrievable. For instance user B’s hidden message can not overwrite that of previous user A. One approach came to me is that we could take the advantage of structure of color image to solve this problem. In color image, one pixel can be represented in various ways. Two common representations are HSL(hue-saturation-lightness) and HSV (hue-saturation-value) which provides us three layers to store data separately. Therefore, our application will support up to three users to hide data into same image and guarantee no conflicts among users. Regardless of multiple users, single user could hide up to three different encrypted messages into same image. Furthermore, before hiding data into image, our application will encrypted the message in the first place to improve information security. Even if the messages are exposed, attackers still have to decrypt to obtain plaintext.

User Interface

Open File

![Image Steganography](image)

Background

Reference

Thank you!