**Project Report**

**Names: Anthony Golubski-Allen,**

**Ibrahim Sardar,**

**Yousef Alsaadi**

**Project: Steganography**

**Date: 4/20/18**

* **Introduction**
* **Background**

We did our research into image steganography. We looked at different image steganography techniques that would allow us to embed images and text into another image. There were list of methods that we found which were classified into two domains. The spatial domain and the frequency domain. The spatial domain is where you directly change the values of the pixels. That is where we found the least significant bit algorithm. The frequency domain is where the images are changed by transforms and then the message is embedded. In that domain, we found out about the discrete Fourier transform and discrete cosine transform algorithms. We looked at each algorithm that we found to see which algorithm to use for our program within the semester. We would decide to go with the least significant algorithm because it was the most common one to use.

We would learn the weakness of the least significant bit algorithm does not take much to ruin the hidden object inside the cover image. The hidden image can be destroyed if you rotate, crop, draw, resize, and other ways that manipulate the stego image. This means that this algorithm will limit the stego image to be uncompressed or be in a format with lossless compression. Lossless compression keeps the image data intact while lossy compression will lose some data. We decided that we should use bitmap for our program for the mean time and add more formats later on.

The programming language that we decided to use for our project is C++. Everybody was familiar with C++. We decided to use free source library called WxWidgets for our graphic user interface. We used another free source library called CImg to be able to read and write to the bitmap image. We decided to go with CImg library because it was simple to add to our program. Also, CImg could use more image formats along the way once we installed the libraries to our computer. CImg has macros built in their code that will use those libraries if they are installed.

* **Implementation details**

We would set up the graphic user interface first. There are four forms in our program. The four forms You can access these forms via file menus.

* **Testing/Results**

**Text to Image**

**Encrypting**

**Before**

****

**After**

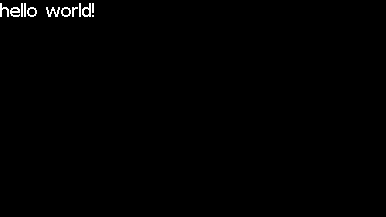
****

**Decrypting**

**Before**

****

**After**

****

**Image to Image**

**Encrypting**

**Before**

****

Figure 1:Image to Hide



Figure 2: Cover Image

**After**

****

**Decrypting Image**

**Before**

****

**After**

****

**File to Image**

**Encrypting**

**Before**

**After**

**Decrypting**

**Before**

**After**

**Video is in this folder directory**

* **Summary/Future Work**
* **References**

.