



# Image Steganography

Janmejay S Purohit  
DSU15CS0027

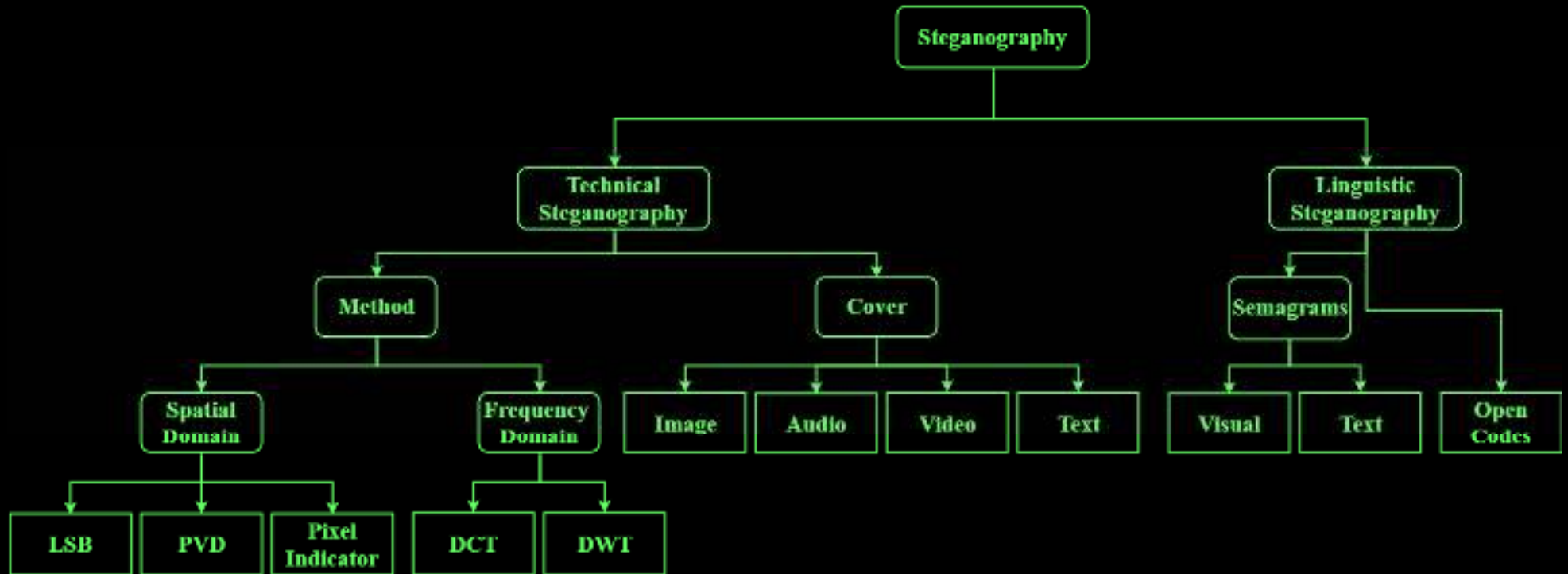
Guide: Dr. Rajesh T.M.

# Introduction

## What is Steganography?

- practice of concealing a file, message, image, or video within another file, message, image, or video
- replaces unneeded or unused bits in regular computer files with bits of different and invisible information
- steganography protects from pirating copyrighted materials as well as aiding in unauthorized viewing

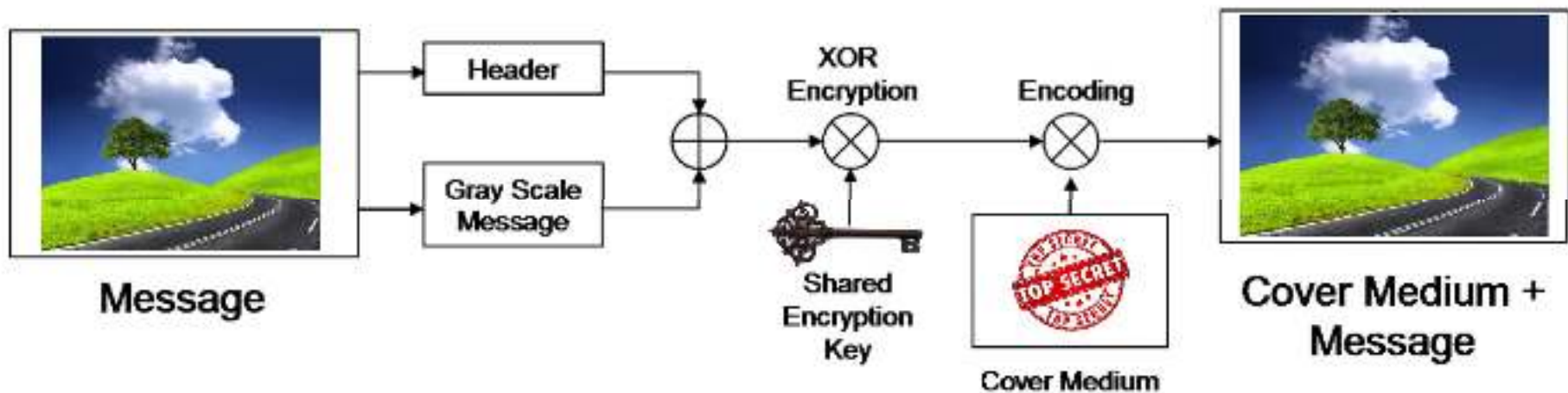
# Types of Steganography



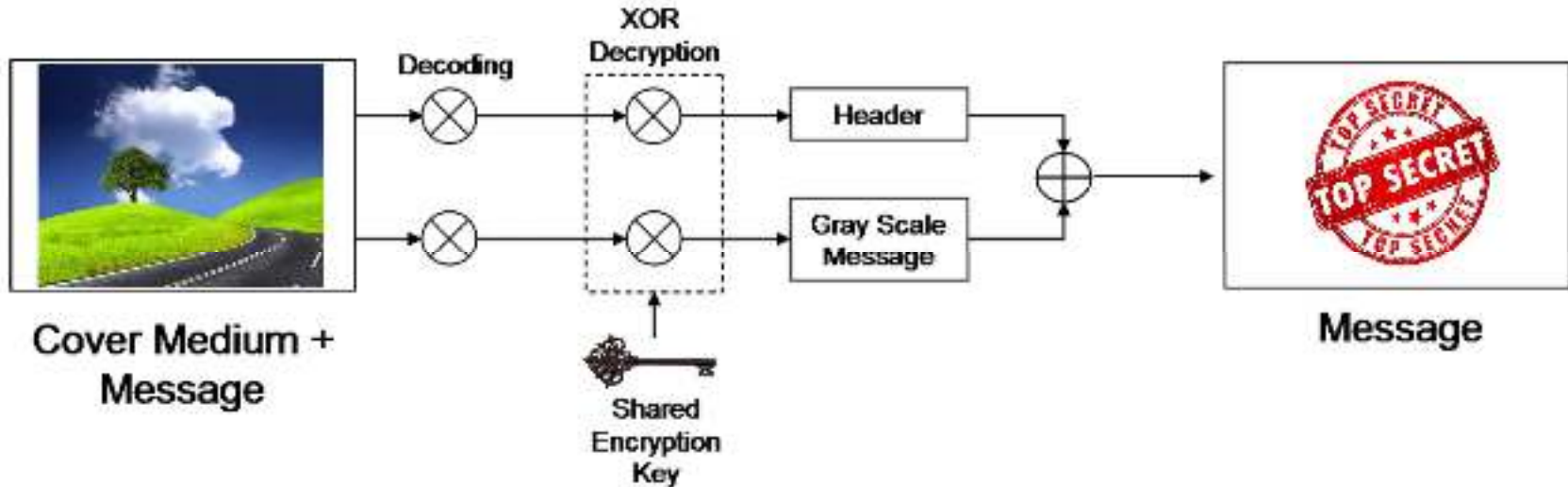
# Objective

- Provide safe and encrypted means of transmission
- Study and research on various steganography techniques
- Formulate more efficient algorithm than the native ones

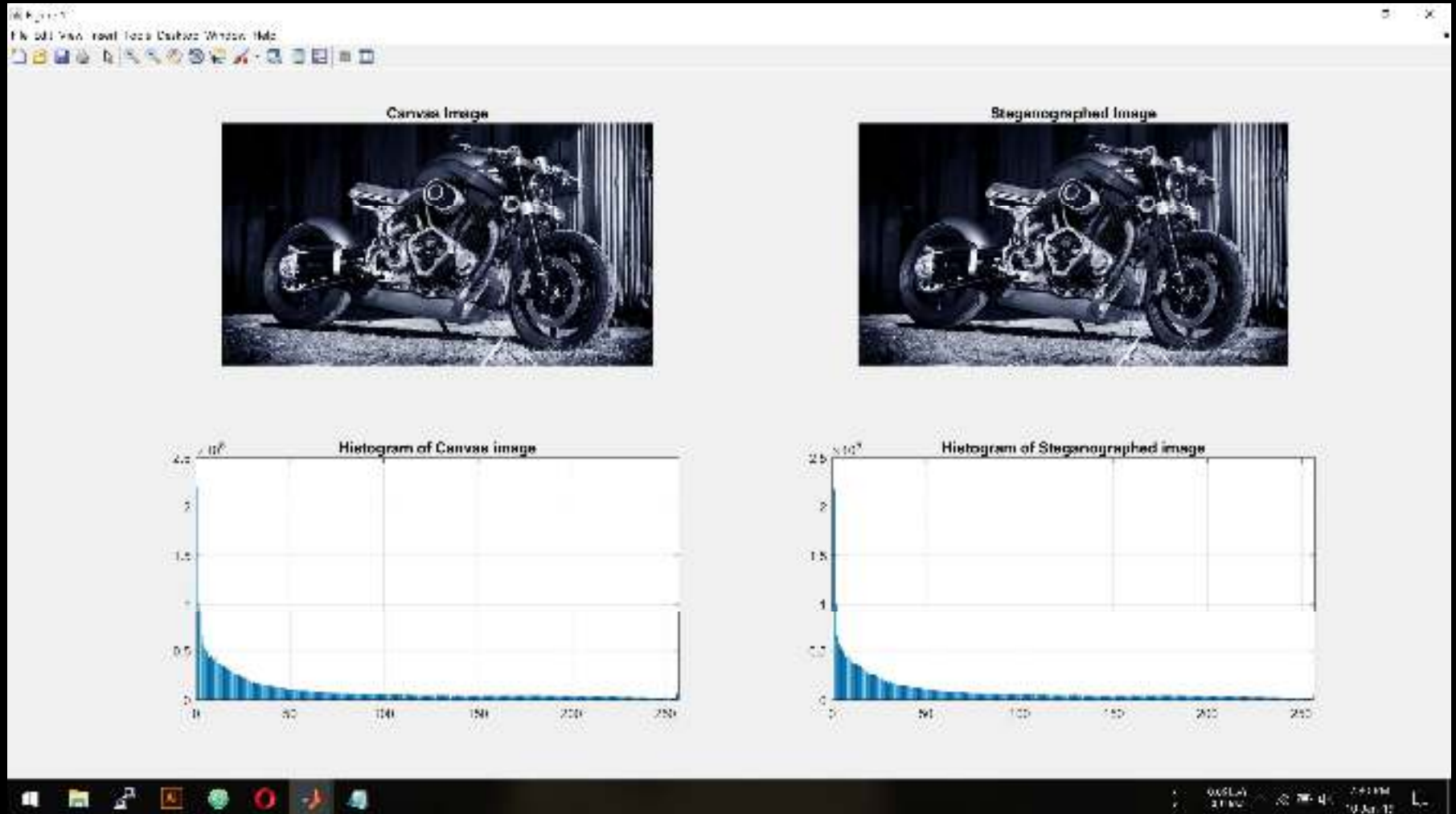
# Encoding Process



# Decoding Process



# Output



# Output

The screenshot displays a Windows desktop environment with several open windows:

- Code Editor:** Shows C++ code for a program named "stepography". The code includes a `main` function that reads a string `s` and a character `ch`, then iterates through the string to find and print characters that match `ch`.

```
int main() {  
    string s;  
    char ch;  
    cin >> s;  
    cin >> ch;  
    for (int i = 0; i < s.length(); i++) {  
        if (s[i] == ch) {  
            cout << s[i] << " ";  
        }  
    }  
    return 0;  
}
```
- Dialog Box:** A "Warning" dialog box is open, displaying text about the source code being generated by an AI model (GPT-4o) and a disclaimer about the accuracy of the information. It includes an "OK" button.
- Terminal Window:** A "Command Prompt" window is open, showing the execution of the program. The output is "stepography".



# Output

The screenshot displays the MATLAB environment. The main window shows a script editor with the following code:

```
clear all; close all; clc; % Clear workspace, close figures, and clear command window  
img = imread('img.jpg'); % Load the image  
[m,n,c] = size(img); % Get image dimensions  
% Create a random binary key of size 100x100  
key = randi(2,100,100);  
% Convert the key to a double matrix  
key = key/255;  
% Perform XOR operation between the image and the key  
steg_img = xor(img, key);  
% Save the steganographed image  
imwrite(steg_img, 'steg_img.jpg');  
% Display the original image and the steganographed image  
figure;  
subplot(2,2,1); imshow(img); title('Canvas Image');  
subplot(2,2,2); imshow(steg_img); title('Steganographed Image');  
subplot(2,2,3); hist(img, 255); title('Histogram of Canvas image');  
subplot(2,2,4); hist(steg_img, 255); title('Histogram of Steganographed image');
```

The workspace window shows the variable `img` with a value of `200x300x3 uint8`. The figure window displays the output of the script, showing the original image, the steganographed image, and their histograms. The histograms are nearly identical, indicating that the steganography process is successful and does not significantly alter the image's statistical properties.

| Command Window                                    |    |    |    |    |    |    |    |    |
|---|----|----|----|----|----|----|----|----|
| How to MATLAB? See resources for Getting Started. |    |    |    |    |    |    |    |    |
| 25  | 25 | 75 | 25 | 24 | 74 | 25 | 25 | 74 |
| 24  | 24 | 74 | 24 | 23 | 73 | 24 | 24 | 73 |
| 23  | 23 | 73 | 23 | 22 | 72 | 23 | 23 | 72 |
| 22  | 22 | 72 | 22 | 21 | 71 | 22 | 22 | 71 |
| 21  | 21 | 71 | 21 | 20 | 70 | 21 | 21 | 70 |
| 20  | 20 | 70 | 20 | 19 | 69 | 20 | 20 | 69 |
| 19  | 19 | 69 | 19 | 18 | 68 | 19 | 19 | 68 |
| 18  | 18 | 68 | 18 | 17 | 67 | 18 | 18 | 67 |

# Output

The screenshot shows a Windows desktop environment with several open windows. The primary window is a steganography application titled 'Steganography'. It features two image thumbnails: 'Steganographed image' (a car at night) and 'Hidden image' (a character's face). A code editor window is open, displaying the following code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Drawing;
using System.Drawing.Imaging;

namespace Steganography
{
    class Program
    {
        static void Main()
        {
            string message = "Message: I love you";
            Image image = Image.FromFile("image.jpg");
            byte[] data = message.GetBytes();
            byte[] imageData = image.ToByteArray();

            int index = 0;
            while (index < data.Length)
            {
                imageData[index] = (byte)(imageData[index] | (data[index] << 7));
                index++;
            }

            Image steganographedImage = Image.FromByteArray(imageData);
            steganographedImage.Save("steganographed.jpg");

            string hiddenMessage = "Hidden Message: I love you";
            Image hiddenImage = Image.FromFile("hidden.jpg");
            byte[] hiddenData = hiddenMessage.GetBytes();
            byte[] hiddenImageData = hiddenImage.ToByteArray();

            int hiddenIndex = 0;
            while (hiddenIndex < hiddenData.Length)
            {
                hiddenImageData[hiddenIndex] = (byte)(hiddenImageData[hiddenIndex] | (hiddenData[hiddenIndex] << 7));
                hiddenIndex++;
            }

            Image hiddenSteganographedImage = Image.FromByteArray(hiddenImageData);
            hiddenSteganographedImage.Save("hidden_steganographed.jpg");
        }
    }
}
```

Below the code, a hex dump of the image data is shown:

```
00000000: 75 41 76 76 101 76 101 000 101 00 101 101 101 101 101 101 101
00000010: 09 62 01 04 09 10 110 102 106 102 109 103 106 103 107 107
00000020: 202 07 75 50 113 100 110 107 109 102 108 109 107 107 107 107
00000030: 110 65 85 67 113 110 109 107 08 104 08 104 08 104 08 104
00000040: 110 00 76 00 100 100 08 104 08 104 08 104 08 104 08 104
```

# Shortcomings of the project

- Transforms the image to grayscale
- Not applicable to black & white images
- Gives output in bitmap format
- Canvas Image must be sufficiently larger than secret image

# Future Scope

- Dynamic Video Steganography

A video is a collection of still images called frames and audio. Video steganography for hiding secret data using Local Binary Pattern (LBP) is the desired. Segmentation is performed using k-means clustering. As each frame has a huge number of pixels and a pixel furthermore has color components, the k-means clustering serves the purpose also concerning the speed.