IMAGE STEAGANOGRAPHY

By

MARADIYA DARSHAN DINESHKUMAR
17BCE057

MAVANI PANAH ASHOKKUMAR
17BCE058

DEPARTMENT OF COMPUTER ENGINEERING
Ahmedabad 382481

IMAGE STEGANOGRAPHY

Mini Project – I

Submitted in fulfillment of the requirements

For the degree of

Bachelor of Technology in Computer Engineering

By

MARADIYA DARSHAN DINESHKUMAR
17BCE057

MAVANI PANAH ASHOKKUMAR
17BCE058

Guided By
Dr. Om Prakash
DEPARTMENT OF COMPUTER ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING
Ahmedabad 382481

CERTIFICATE

This is to certify that the project/Seminar entitled "IMAGE STEGANOGRAPHY" submitted by MARADIYA DARSHAN DINESHKUMAR (17BCE057) and MAVANI PANAH ASHOKKUMAR (17BCE058), towards the partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Engineering of Nirma University is the record of work carried out by him/her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination.

Dr. Om Prakash
Assistant Professor
Department of Computer Engineering,
Institute of Technology,
Nirma University,
Ahmedabad

Dr. Madhuri Bhavsar
HOD, Dept. of Information Technology
Institute of Technology,
Nirma University,
Ahmedabad

ACKNOWLEDGEMENT

We are thankful to our guide Dr. Om Prakash to give us encouragement, support, and guidance at the time we needed the most. His expertise in the field of image processing helped us a lot in exploring this rather new field and developing more interest in it.

We thank our family members without whom none of this would have been possible.

ABSTRACT

Image steganography is the technique of hiding information in an image. The said image can later be decoded by the receiver to obtain the message all the while remaining hidden in plain sight to any third party. It can also be used as an alternative to data storage because such images can also store data in them not requiring a separate file to manage the data. The main purpose of image steganography is communication between parties without any third party getting the message. In this paper, we will see how image steganography is used in a modern time while giving an understanding of what image steganography is and how we can accomplish it.

TABLE OF CONTENTS

1 Introduction

1.1 General

The report is on the project "Image Steganography" that takes the text message as an input, hides it into the cover image and produces stegoimage, which is later used in steganalysis to retrieve the hidden message. We have used MATLAB to implement the process.

1.2 Objective of Study

Unlike in cryptography, steganographic techniques hide the secret message in plain sight while one can totally see a message and determine whether it is cypher text or not (due to presence of meaningless sequence of letters and digits in the message). In image steganography, these secret messages are hidden within the data (pixels) of an image making it virtually impossible for human eye to detect the change in the image due to alteration because of steganography. There are many different types of steganographic techniques including:

- Least Significant Bit (LSB)
- Encrypt and Scatter
- Masking and Filtering
- Redundant Pattern Encoding
- Transformation Algorithms (DCT, DWT, etc.)

In this report, Discrete Cosine Transformation (DCT) Steganography is explained in great detail.

1.3 Scope of Work

Some places where we can use image steganography include:

- Law and Enforcement Agencies – Criminal data can be stored inside the mugshot of the criminal.

- Medical Agencies – Information about the patients can be stored in their photo IDs.
- Secured communication – Two parties can communicate with each other by using steganographic images while any attacker would only conclude that they are exchanging pictures.

2 Fundamental terminologies

2.1 Discrete Cosine Transformation (DCT)

Discrete Cosine Transform (DCT) is a technique which is used transform an image from spatial to frequency domain. It divides the image into multiple 8x8 blocks with respect to its frequency domains, i.e., high, middle and low. In this technique, DCT coefficients of the given image are obtained, then the secret message is later inserted in the image of DCT coefficients.

Given below is the general equation of 2D DCT:

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

...(2.1)

For u,v = 0 to N-1 (Integer numbers). Here, the size of image is NxM. c(i, j) is the intensity of the pixel in row i and column j; C(u,v) is the DCT coefficient in row u and column v of the DCT matrix. DC component of the image is at low frequency; it appears in the upper left corner of the block. Compression can be achieved because the lower right values represent higher frequencies, and generally small enough to be neglected as it is not possible for human eye to detect such high frequency values. DCT is used in steganography as the image is divided into 8x8 blocks of pixels. Working from left to right, the DCT is applied to each block. Each block is divided by the standard quantization table and the message is inserted in DCT coefficients as a secret message.

2.2 Quantization table

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig. 2.1 Standard Quantization Table

The table shown in Fig. 1.1 is the standard quantization table used in JPEG compression. We use the same block in our implementation for the sake of simplicity. There is no specific reason as to why we use the standard quantization block.

3 Overall Description

3.1 DCT steganography algorithm

DCT based steganography has been used in this paper to hide secret message into cover image and the following steps show the algorithm of encrypting and decrypting the image.

3.1.1 Encryption

Step 1: Reading cover image.

Fig. 3.1.1 Input image

98	96	94	88	84	90	90	80
106	105	102	93	85	88	86	76
111	112	110	99	88	87	82	71
100	104	108	101	91	90	84	73
93	101	111	112	108	110	107	97
83	92	104	108	109	116	114	104
75	79	85	86	85	91	90	79
90	89	89	83	78	82	78	66

Fig. 3.2.2 Original block of an input image

Step 2: Splitting cover image into 8×8 block of pixels and applying 2D DCT on each block.

745.7500	32.2409	-18.9259	1.9269	-20.7500	10.5177	1.3450	-0.9021
11.7959	32.1873	12.0685	-0.3767	-0.0426	-0.1722	-0.2055	0.0073
-47.2231	4.1071	14.8728	6.8590	0.0396	0.6743	-0.2286	0.0533
27.8846	-45.2387	-6.7722	-0.0480	0.3779	0.1426	-0.3624	-0.0310
-4.5000	6.8906	-0.1353	0.1224	0.5000	0.3308	0.3266	0.1399
-21.2750	0.1043	-0.0945	0.3194	-0.0224	0.2113	-0.0736	0.0655
14.6897	0.3472	-0.4786	0.3196	-0.5576	-0.3897	0.3772	0.0621
0.2372	0.1841	0.3905	-0.1320	0.4795	0.0501	-0.6310	0.1493

Fig. 3.3.3 DCT of the block

Step 3: From left to right of cover image, each block is divided by the standard quantization table (Fig. 1.1).

Step 4: The entered message is reduced to a lesser bit number by using Huffman encoding on the message. This is done so that large messages are converted to lesser bit-length while encrypting so that more message can be accumulated in lesser space.

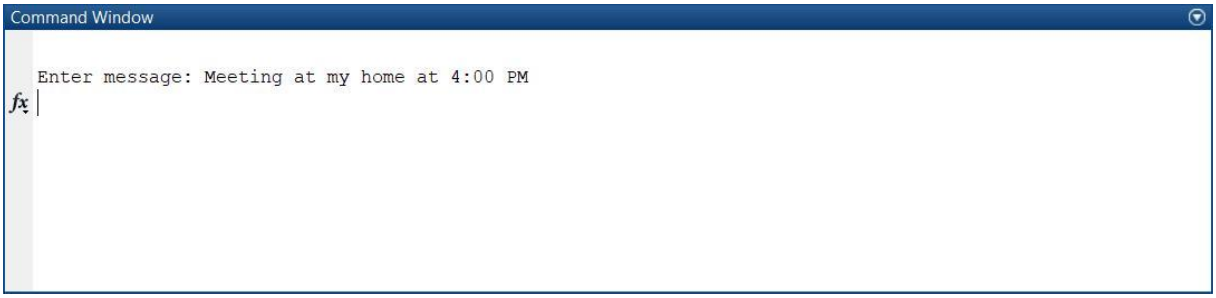

Fig. 3.4.4 Secret message

Step 5: The Huffman encoded text is then embedded to the DCT coefficients of the image.

47	3	-2	0	-1	0	0	0
1	3	1	0	0	0	0	0
-3	0	1	0	0	0	0	0
2	-3	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Fig. 3.5.5 DCT + quantized block

Step 6: Writing stego image.

3.1.2 Decryption

Step 1: Reading stego image.

Fig. 3.6.6 Stego image

Step 2: Dividing stego image into 8×8 block of pixels and Multiplying each block with the standard quantization table.

752	33	-20	0	-24	0	0	0
12	36	14	0	0	0	0	0
-42	0	16	0	0	0	0	0
28	-51	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-24	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Fig. 3.7.7 Unquantized block

Step 3: Applying inverse DCT to each block.

95.2803	98.7298	94.8600	85.4523	83.8684	90.3494	91.9791	87.3174
109.4550	112.6052	107.5519	95.3648	89.1983	89.9912	86.3237	78.4538
109.3437	113.5483	109.4985	96.7748	87.6685	83.5658	74.7374	63.5630
100.3370	107.6900	108.6535	100.9081	95.1830	92.3498	83.2897	71.5549
93.4638	104.1955	111.0818	110.4482	111.6210	114.4216	109.1938	99.3597
80.4207	91.9577	100.5436	102.5452	107.0992	113.5123	111.3667	103.3152
74.6011	83.8788	88.7819	86.9332	88.5473	93.3787	90.7584	82.7161
84.7395	91.3747	91.7806	84.8094	81.8409	83.3273	78.7234	69.8162

Fig. 3.8.8 Inverse DCT block

Step 4: The message is then extracted from the inverse DCT coefficients of the image and the received Huffman encoded message is decoded.

Step 5: The message is the given as the output.

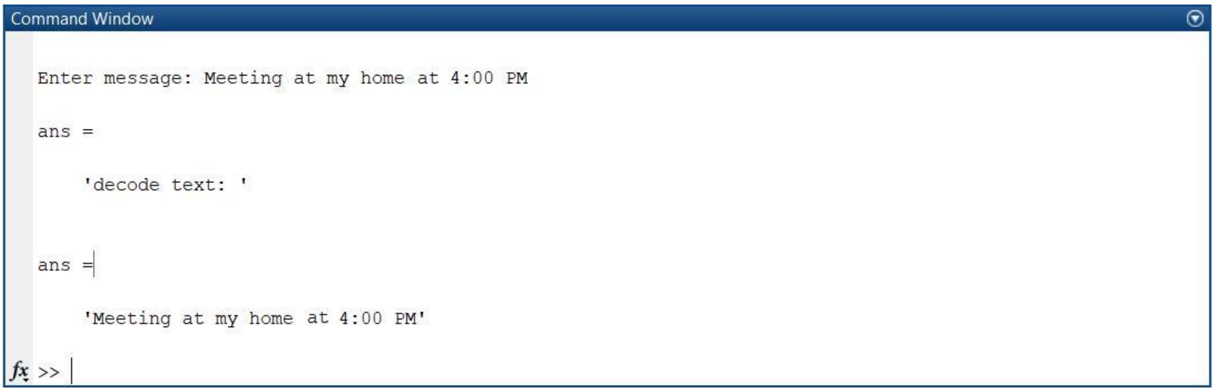

Fig. 3.9.9 Decoded message

F Conclusion & Summary

F.1 Summary

This report discussed the basic fundamentals of image steganography, implementations of image steganography in real world and some terminologies required to understand image steganography. Also, implementation of DCT steganography and how image compression takes a major part in it.

F.2 Conclusion

After successful completion of the project we learned to:

- There are many different ways we can exchange image secretly in this day and age.
- How an image is processed by a computer.
- Different ways to hide message inside images.
- How we can understand the fundamentals of different techniques and use them to achieve our goal.
- Importance of compression of images.

References

1. Ajay Nain. *"Implementation of steganographic techniques in matlab"*
2. Hossein Sheisi, Jafar Mesgarian, and Mostafa Rahmani. *"Steganography: Dct Coefficient Replacement Method and Compare With JSteg Algorithm"*
3. Monika Gunjal, and Jasmine Jha. *"Image Steganography Using Discrete Cosine Transform (DCT) and Blowfish Algorithm"*
4. AlaaAbdulhusseinDaleh Al-magsoosi. *"Comparison study between LSB and DCT Based Steganography"*

Appendix – List of Useful Websites

1. Advanced JPEG Steganography and Detection by John Ortiz. https://www.youtube.com/watch?v=BQPkRlbVFEs
2. Secrets Hidden in Images (Steganography) – Computerphile. https://www.youtube.com/watch?v=TWEXCYQKyDc
3. Image Compressing using Discrete Cosine Transform in Matlab- Part 1 https://www.youtube.com/watch?v=UU0tLHsMaOA
4. The Tickle Trunk: Guide to JPEG-89 Compression. http://cgjennings.ca/toybox/hjpeg/