Analysis of DCT and DWT Based Steganography

A Project Report

submitted by

N. HARSHA VARDHAN REDDY
EE09B025

in partial fulfilment of the requirements
for the award of the degree

of

BACHELOR OF TECHNOLOGY

DEPT. OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

PROJECT CERTIFICATE

This is to certify that the project report titled **Analysis of DCT and DWT Based Steganography**, submitted by **N.HarshaVardhan Reddy**, to the Indian Institute of Technology, Madras, for the award of the degree of **BACHELOR OF TECHNOLOGY**, is a bona fide record of the project work done by him under our supervision. The contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Aravind R.
Project Guide
Professor
Dept. of Electrical Engineering
IIT Madras, 600 036

Place: Chennai

Date: 18th June 2013

Acknowledgments

I will be forever indebted to Prof. Aravind for giving me such a wonderful opportunity to work on this project and for introducing me to the fascinating world of image processing. He has given me the necessary freedom of choice and allowed me to explore my interests with necessary guidance at all stages of the project. I cannot express enough how it has changed my perception of engineering and hope to contribute more to the field in my future.

I take this stage to thank IIT Madras in completeness specifically the faculty who have made an indelible impact on my life. My four years here have been most humbling and thought provoking. I will always be proud of being associated with such a prestigious institution.

The constant support of the my friends who have helped me technically as well as personally in completing this project is much appreciated. Their valuable criticism has been very helpful in shaping the project.

Finally my parents and family, who have always believed and supported my choices. I owe everything i am today to them and hope i will be able to make them as proud of me as i am of them.

Abstract

Steganography is the art of concealing information within other information without arousing suspicion that secret communication is in fact taking place. Many cover or carrier mediums are used for the purpose of steganography with digital images being most popular because of their very high frequency on the Internet. For hiding secret information in images, there is an abundant variety of steganographic techniques at our disposal. Some are more complex than others with each of them have their respective merits and demerits. The choice of technique is very specific to the requirement at hand. Applications may require 'absolute' invisibility of the secret information, while others require a larger secret message to be hidden.

This project studies the most fundamental approaches to image steganography and intends to give an overview of its uses and techniques. It also tries to identify the requirements of a good steganographic algorithm and analyzes which techniques are more suitable for a particular application alongside the trade offs involved in implementing them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Information has become one of the most valuable resources available in the modern world. The sheer volume already in existence and the rate at which new information is being generated is simply overwhelming. Though its seemingly limitless capabilities are revolutionizing the way we live and interact, the inherent nature of storing and transferring this information has given rise to the largely unresolved issue of its security and privacy.

In an age where our physical lives are more in sync with our virtual lives, the ability to keep sensitive information out of prying eyes takes precedence and information security is now becoming the norm for all things virtual. Currently we do not have the infrastructure or the capacity to do this effectively or efficiently. The trade off between the accessibility for the owner and security against malicious elements also imposes new constraints for doing this.

Established techniques such as password enabled access, data encryption always have the pressure of becoming incapable of handling the brute force attacks of high computing power which is easily within reach for a reasonable price in the current age. Tackling this issue needs novel approaches that are more secure against such attacks and the ages old method of steganography may potentially provide us with a solution which is simple and efficient.

[1]*Steganography: The art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity.*

It relies on the imperceptibility of the secret message hidden inside a seemingly innocent cover information which acts as decoy for the actual message. Steganography has evolved over the ages and has adapted itself as per the time period of

[1]Source: Wikipedia

its use. We shall now see some of the documented cases of its use and take note of the various methods employed which can adapted to our modern challenge.

1.1 History of Steganography

1.1.1 Origins

The first documented cases of steganography being used can be traced back to *The Histories* written by Herodotus ($455-520$ BC), the first Greek historian. He recounts the story of Demaratus who sent a warning about a forthcoming attack to Greece by writing it directly on the wooden backing of a wax tablet before applying its beeswax surface. Another accounts of Histaiaeus who shaved the head of his messenger and wrote the secret message on the scalp. After the hair regrew, the messenger traveled freely to the destination without being caught and delivered the message by shaving his head again.

Pliny the Elder ($23-79$ AD) showed how the milk of Thithymallus plant dried to transparency when applied to paper making it invisible and darkened to brown when heated was the earliest forms of invisible ink used to communicate. The ancient Chinese wrote notes on pieces of silk which were then wadded into balls and coated in wax to be swallowed by the messenger who delivered the message upon retrieval at gastrointestinal convenience.

1.1.2 Modern Steganography

The more recent versions of steganography were first seen during the renaissance period where the use of steganographic ciphers was invented which utilized a particular sampling pattern of letters from verses which formed the secret message when put together.

Steganography and Cryptography gained prominence especially during war times where communication of sensitive information such as troop movement, attack targets became vital and decisive in the outcomes. The World wars saw major advancements in these fields with secret message being passed on through cover mediums like letters, newspapers, pinup posters etc. The combination of ciphers with the steganographic message was also used for the first time to derail the enemy who has found the steganographic message. The Axis were very successful in developing new powerful cipher machines and inventions such as microdots gave them the edge over the allies.

Cold war saw a revolutionary jump in the scale and techniques used for steganography with covert communication in the spy game becoming a more serious one with the life of spies constantly endangered with ever possible leak of information
.

1.2 Steganography in the Digital Age

Internet which first revolutionized information sharing, in its latest digital form has caused an explosion of information. The ease of storing the digital formats and communicating this information has increased many fold the volume being circulated. Social networking has made sharing of information very easy. This large volume of information which is difficult to monitor constantly, has been exploited by steganography. It has adapted itself to the redundancy in these digital carriers and has transformed into a viable secret communication technique.

1.2.1 Popular Cover Mediums

Steganography needs redundancy to be capable of surviving detection.Common digital formats of audio, images etc. have a lot of redundancy which can be exploited to communicate secret data.Text steganography using digital files is not used very often since text files have a very small amount of redundant data. Given the proliferation of digital images and the large amount of redundant bits present in their digital representation, images are the most popular cover objects for steganography.

To hide information in audio files techniques similar to image files are used. Unique to audio steganography is masking, which exploits the properties of the human ear to hide information. A faint but audible sound becomes inaudible in the presence of another louder audible sound.This property creates a channel in which to hide information. Although nearly equal to images in steganographic capacity, their larger size makes them less popular.

The term protocol steganography refers to the technique of embedding information within messages and network control protocols used in network transmission. In the layers of the OSI network model there exist covert channels where steganography can be used. An example of where information can be hidden is in the header of a TCP/IP packet in some fields that are either optional or are never used.

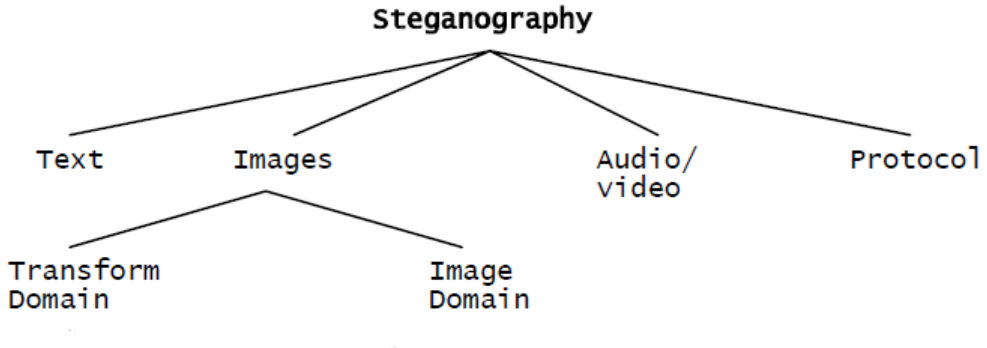


Figure 1.1: Popular Steganography Cover Mediums

1.2.2 Steganalysis

Steganalysis is the art and science of detecting messages hidden using steganography. It can also serve as an effective way to judge the security performance of steganographic techniques.The huge diversity of carrier images and the wide variation of data embedding algorithms make steganalysis a difficult task.

An original cover medium and its stego-version always differ from each other in some aspects since the cover medium is modified during the data embedding. This basic principle is used to develop methods to detect steganography. In other words, a good steganographic method should be imperceptible not only to human vision systems, but also to steganalysis techniques.

Chapter 2

Technical Introduction

This chapter gives an overview of the various aspects necessary to our understanding of image steganography. Firstly, the representation of the digital image and its common storage formats are briefly discussed. The JPEG standard for images in dealt with more in detail as it is the most common format for images especially on the internet. We shall see what advantages the standard offers and we will be using this format for the majority of the project ahead.

2.1 Digital Images

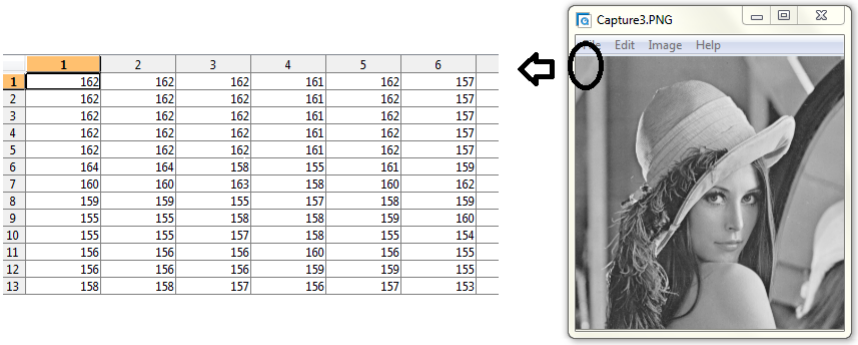
Digital images have just faded out the film images with the advancement in CCD & CMOS sensor based cameras. They offer more control to user in terms of the resolution, editing and sharing. The way that these images are captured and stored in a certain format is completely dependent on the user requirement. Most of this information stored is highly redundant which makes it ideal for steganography.

2.1.1 Image Representation

[1] *A digital image is a numeric representation of a two-dimensional image .*

Digital images in their simplest form are just a rectangular array of numeric values which represent the associated physical quantities characteristic to the image. The values stored may describe quantities such as luminous intensity, chrominance, color map association etc. As with any digital format the limited range of the values which can be expressed is the drawback for digital images as well. an example of a gray scale image and the associated image representation in the BMP format is given in the next page.

[1]Source : Wikipedia



	1	2	3	4	5	6
1	162	162	162	161	162	157
2	162	162	162	161	162	157
3	162	162	162	161	162	157
4	162	162	162	161	162	157
5	162	162	162	161	162	157
6	164	164	158	155	161	159
7	160	160	163	158	160	162
8	159	159	155	157	158	159
9	155	155	158	158	159	160
10	155	155	157	158	155	154
11	156	156	156	160	156	155
12	156	156	156	159	159	155
13	158	158	157	156	157	153

Figure 2.1: A Bit Map Image

This numeric representation forms a grid and the individual points are referred to as pixels. Most images consist of a rectangular map of the image's pixels represented as bits giving both the location of the pixel is and its intensity value for a certain property.

2.1.2 Image Formats

Image formats are standardized methods of storage for images. Each have their own advantages and trade offs. When the image is captured using a digital camera it stores the information in a particular format with JPEG, TIFF, RAW formats being most common. These can be later converted into any other format depending on the intended application. Broadly they are classified into lossless and lossy compression formats depending on the amount of original information retained in them. Lossless formats offer better image characteristics but are very large in size whereas comparatively lossy formats offer acceptable characteristics with much lower file size. Below is a table briefly describing the common formats and their characteristics.

Format	***Compression***	***Size***	***Comments***
BMP	Lossless	Large	Simple and widely accepted
GIF	Lossless	Limited	8-bit palette
PNG	Lossless	Variable bit depth	Works well with network applications
JPEG	Lossy	Can be varied	Very popular on the internet
TIFF	Lossless & Lossy	Large	A standard for printing

Table 2.1: Image formats

As mentioned previously we shall be using JPEG standard for most of the implementation part in the further sections. The following section outlines the various stages involved in the encoding process.

The JPEG Standard

The JPEG standard is the most widely used format for images and offers control over the quality and size of the image. It is a lossy compression format which exploits the properties of the human vision system (HVS) to reduce the information retained from the raw format which cannot be perceived by the naked eye.

To compress an image into JPEG format, the RGB color representation is first converted to a YCbCr representation. In this representation the Y component corresponds to the luminance (or brightness) and the Cb and Cr components stand for chrominance (or color). The human eye is more sensitive to changes in the brightness (luminance) of a pixel than to changes in its color. This fact is exploited by the JPEG compression by down sampling the color data to reduce the size of the file.

The next stage is the transformation of the image into the frequency domain. For this the Discrete Cosine Transform (DCT) is used. By grouping the pixels into 8 x 8 pixel blocks we transform each pixel blocks into 64 DCT coefficients each. A modification of a single DCT coefficient will affect all 64 image pixels in that block.

The next step is the lossy part of the compression. The human eye is fairly good at spotting small differences in brightness over a relatively large area, but not so good as to distinguish between different strengths in high frequency transitions. This means that the strength of higher frequencies can be diminished, without significantly changing the appearance of the image. JPEG does this by dividing all the values in a block by a corresponding quantization coefficient. The results are rounded to integer values and the coefficients are encoded without loss using Huffman coding.

JPEG allows for adjusting the degree of compression based on the quantization table being used. This allows for a selectable trade off between storage size and image quality. JPEG typically achieves 10:1 compression with little perceptible loss in image quality. But at higher compression the blocks become very apparent especially in the smoother regions.

The decoder does the reverse of the encoder, it decodes the lossless Huffman coding and then de-quantizes the DCT coefficients. These DCT coefficients are

now transformed back into the spatial domain using Inverse DCT. After the conversion into the RGB format, the compressed image is available in its pixel domain for viewing. The following diagram summarizes the procedure.

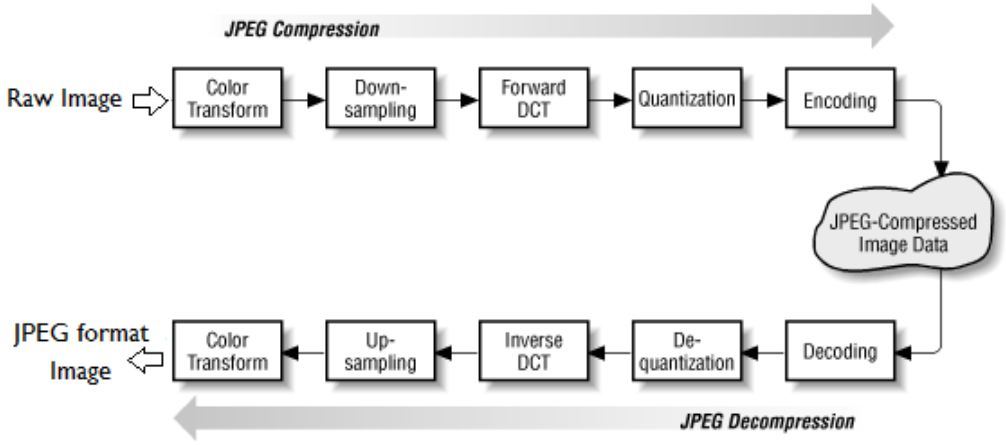


Figure 2.2: JPEG flow diagram

Chapter 3

Image Steganography

Steganography is most suited to digital formats which exhibit high degree of redundancy. Redundancy in this context can be defined as the bits of an object that provide accuracy far greater than necessary for the object's use and display. For an image, the redundant bits are those bits that can be altered without perceivable change in the image. The two components involved in image steganography are a *cover image* and the *secret message*. The cover image needs to be a very generic and common image which is unlikely to arise any suspicion. The hidden message can be anything the user wants to communicate secretly. It can be images, audio, text or anything else which can be represented as a bit stream. After the embedding the message into the cover image the resultant image is termed as the *stego-image*.

3.1 Evaluation Criteria

A good steganographic method should be evaluated based on certain criteria to ensure that the communication cannot be detected by anyone apart from the intended party. The criteria below are in the context of image steganography which will be used to analyze our implementations:

3.1.1 Visual Perception

As the first line of defense, the imperceptibility to the naked eye is crucial. To quantify this we use measures such as bit error rate (BER), mean squared error (MSE) and Peak-signal-to-noise ratio (PSNR) to compare the stego-image with the cover image.

BER

The bit error rate or bit error ratio (BER) is defined as the number of bit errors divided by the total number of transferred bits during a studied time interval. We have used it in a modified manner as the number of pixels changed divided by the total number of pixels in the image. This will help us measure the number of changes being made to the the cover image due to the information embedded.

MSE

Mean squared error is defined as :

$$\text{MSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[I_s(i,j) - I_c(i,j)]^2$$

where, $I_s$and I_c denote the stego-image and the cover image respectively for an image of M x N dimensions. This will help us measure the extent of the change made to the cover image pixels on an average due to the embedding.

PSNR

Peak Signal-to-Noise Ratio is defined as the following in the decibel scale :

$$\text{PSNR} = 10log_{10}\frac{(255)^2}{MSE}$$

where, I_{max} denotes the maximum possible value for the intensity value, for an 8-bit image it is 255 and MSE denotes the mean squared error. PSNR is a good measure for comparing restoration results for the same image, here we compare the stego and cover images. PSNR is used in addition to the MSE as it also takes the image intensity scaling into consideration.

3.1.2 Embedding capacity

The embedding capacity is defined as the maximum size of the message which can be hidden in the cover image. A method giving more embedding capacity would be preferred if compared to a method with equivalent performance in the previously mentioned metrics. We shall compare the capacities of the various approaches in the following chapters.

3.1.3 Robustness to attack

The ideal stego-image needs to be robust against attacks such as luminosity changes, re-compression, cropping effects etc.. The method implemented should be capable of withstanding some of them at the very least. Our project gives only a descriptive analysis of their effects and does not try to implement or solve for them.

3.2 Various Steganographic Approaches

Image steganography techniques can be divided into two groups based on the domain in which the message is embedded. The technique used is very dependent on the image format bring used as the cover, and the robustness expected form the stego-image.

3.2.1 Spatial Domain Approach

Spatial or Pixel domain approach involves embedding the message directly in the pixel values of the image. The techniques which apply bit insertion and noise manipulation directly on the image pixels come under this category. These methods are very simple algorithmically and easy to implement. But such an approach is possible only for image formats which are lossless like BMP,GIF etc. Brief description of the LSB substitution method is given below.

LSB in BMP images

Least significant bit (LSB) substitution is a common, simple approach to embedding information in a cover image. The least significant bit of the some or all of image pixels is changed according to the bit stream of the secret message. For example, a 8-bit gray scale image has 256 possible intensities for each pixel. Changing the LSB of the pixel results in only a +1 or -1 change in the intensity value. Such changes cannot be perceived by the human eye and the hidden message is concealed successfully. With a well chosen cover image it is possible to hide in the least two (or more) significant bits [2]with good imperceptibility.

LSB in Palette Based Images

Palette based images, for example GIF images, are a popular image file format used on the internet. By definition a GIF image cannot have a bit depth greater than 8, thus the maximum number of indexed colors it can comprise of is 256. Each pixel is represented as a single byte and the pixel data is an index to the color palette. GIF images can also be used for LSB steganography, but the problem

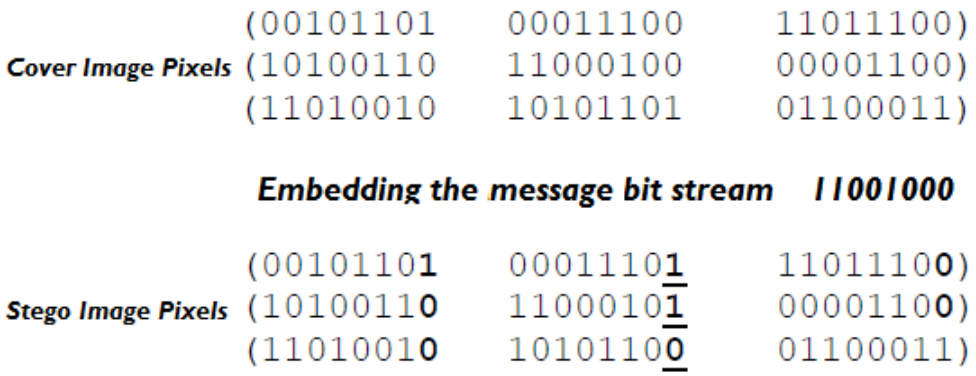


Figure 3.1: LSB on a set of 8-bit pixels

with the palette approach is that should a LSB of a pixel change, it can result in a completely different color since the index is changed. If adjacent palette entries are similar, there might be little or no noticeable change but should the adjacent palette entries be very dissimilar, the change would be evident. Unfortunately any tampering with the palette of an indexed image leaves a very clear signature making it easier to detect.

Merits

i. Method is very simple and does not involve any rigorous computation as such.

ii. Offers good embedding capacity.

Drawbacks

i. Image formats used are necessarily lossless and not commonly used anymore, hence may attract undue attention.

ii. Very sensitive to random attacks like LSB flushing, partial luminosity changes, gaussian blurring etc. and easily detectable using steganalysis.

3.2.2 Transform Domain Approach

Transform domain approach involves the transformation of the image pixel data into its frequency domain. These techniques hide the secret message in a more controlled manner inside significant areas of the cover image by utilizing the properties of the transform, making it more robust. These methods are independent

of the image format and the embedded messages are more likely to survive conversion between lossy and lossless compression.

It was presumed historically that image steganography could not be possible in lossy formats like JPEG. One of the major characteristics of steganography is the fact that information is hidden in the redundant bits of the image and since these bits are left out when using lossy compression, it was feared that the hidden message would be destroyed. However, the properties of the algorithms have been exploited in order to tackle this issue. Below the JPEG standard prescribed DCT and DWT transform based steganography are described.

Discrete Cosine Transform

During the DCT transformation phase of the JPEG compression algorithm, rounding of the DCT coefficients is done which is not much noticeable in the pixel domain. To attain the desired compression the coefficients are quantized using respective quantization levels. This is the lossy stage of the JPEG standard, after which lossless Huffman encoding is done to further compress the data.

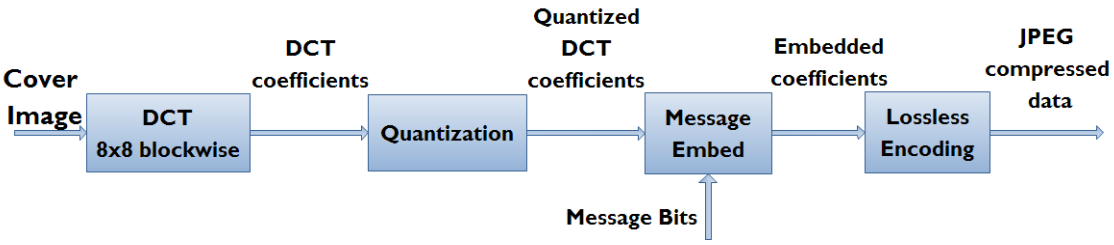


Figure 3.2: DCT based steganography

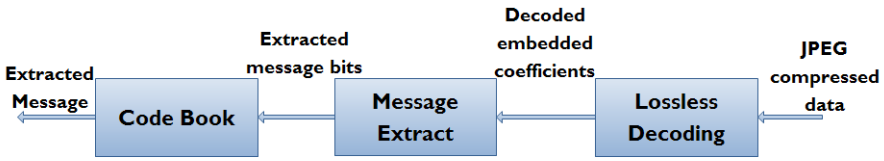


Figure 3.3: Extraction of the hidden message

Between the lossy and lossless stages, the message bit stream is embedded into the DCT coefficients. Once again LSB substitution method is used for embedding, but as it is done in the frequency domain the effect of each change is spread out across the 8x8 pixel block. This makes it very difficult to detect the embedding visually in the pixel domain.

Merits

i. The embedding in the transform domain makes it very imperceptible to visual inspection.

ii. Does not attract unwanted attention as the JPEG format is very common.

iii. Adjustable trade off between capacity and stego-image quality.

iv. Less sensitive to common random attacks due to the nature of the embed.

Drawbacks

i. A complex algorithm and more computationally tasking than previous methods.

ii. Re-compression attacks will cause loss of embedded info from the the stego-image.

iii. LSB insertion modifies the statistical properties of the stego DCT compared to the cover DCT which can be used to detect the stego-image.

Discrete Wavelet Transform

The DWT is a very powerful tool in signal processing with a relatively new application in the image formats with JPEG 2000 standard using it. For image steganography, DWT's space and frequency resolution are exploited to embed the hidden message in the sub-bands of the transform. It offers clearer frequency resolution than DCT giving us more control over the embed region.

Many wavelets are available for use in the decomposition process, here we have specifically used the Haar wavelet which is the simplest one. The single level decomposition gives the following sub-bands:

i. LL or Approximate(A)

ii. LH or Vertical(V)

ii. HL or Horizontal(H)

ii. HH or Diagonal(D)

In the project we have implemented 3 types of embedding with one being the LSB embed in each of the sub-bands to identify the most suitable one, another using a difference measure based embed in the H and V sub-bands together and finally embedding in the 'D' sub-band using the properties of the Haar wavelet.

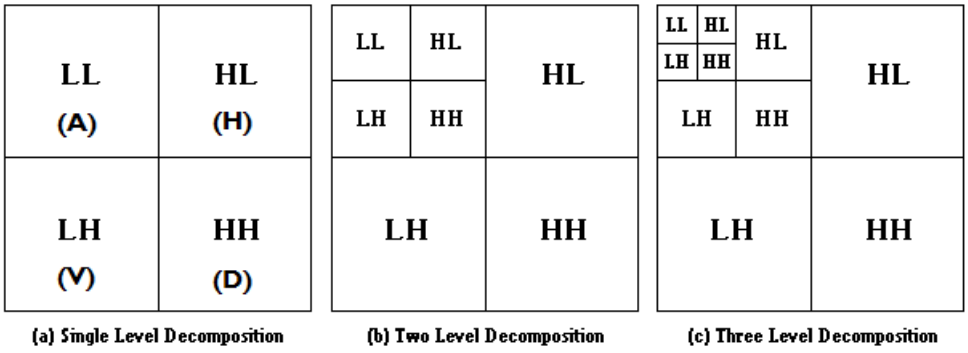


Figure 3.4: DWT transform for upto a 3 level decomposition

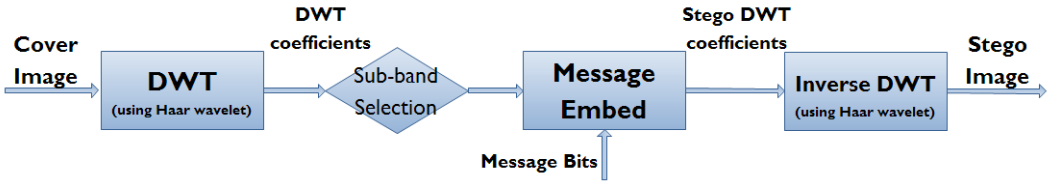


Figure 3.5: DWT based steganography

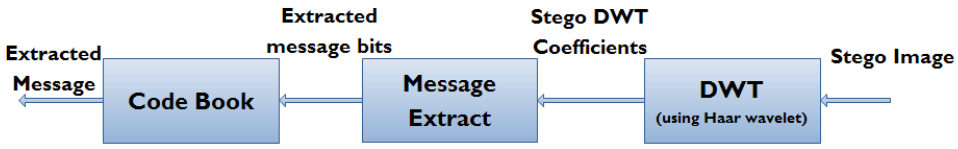


Figure 3.6: Message extraction from the stego-image

Merits

Shares all the merits of the DCT approach along with the following:

i. Gives more control over the embed than DCT.

ii. Embedding can be done in further decomposition levels giving more robustness at the cost of capacity.

Drawbacks

i. Capacity is not fully utilized as the embedding is done effectively in a single sub-band.

ii. The Haar wavelet is not well suited for good quality image reconstruction.

Chapter 4

Implementation

In this chapter, the implementation of different image steganographic techniques are outlined and the relevant discussion regarding their merits and demerits is done. The entire implementation has been carried out in MATLAB using the image processing toolbox. The derivative thought process and development of the project in incremental stages will be presented in the following sections. The various attempts made to tackle the issues faced and the necessity of each implementation will be also be justified.

We shall start with the spatial domain approach followed by the transform domain approach. The algorithms used and their understanding will give us the necessary foundation to analyze the results presented in the next chapter.

4.1 Spatial Domain Implementation

As discussed before, the simplest forms of image steganography constitute this approach. We have used a lossless BMP format image as the carrier and carried out a single bit (LSB) replacement for different message types. Each of them may be very similar in the terms of the embedding procedure, but the stego-images resulted will not share the same statistical noise properties. This can exploited by steganalysis techniques[1] to identify and categorize stego-images.

4.1.1 Message Bit Stream

The message object is converted into an embeddable bit stream. This bit stream will be embedded into the cover image to get the stego-image. The stream is generated by looking up a suitable code book which must be available at both ends of the communication.

[1]Noise Floor Consistency Analysis

Text message

A text message can be converted into a bit stream by replacing the letters with their respective ASCII values. The ASCII value (decimal) is converted into its 8-bit representation, i.e. its corresponding bit string is formed. By collating all the bit strings in the appropriate manner we can obtain a bit stream. The bit stream can be encrypted as well using a pseudo-random key generator. This will increase the protection in case the stego-image is detected.

Image message

The image message is also converted into a bit stream by converting its pixel values into corresponding bit strings. The bit strings exhibit only a steady variation for most part with significant changes only when sharp features are present. This statistical property can be exploited by the attacker to detect the presence of a certain hidden message.

Random bit stream

This message is used to replicate any other possible message patterns or emulate an encrypted message. A random generator is used to generate the bit stream.

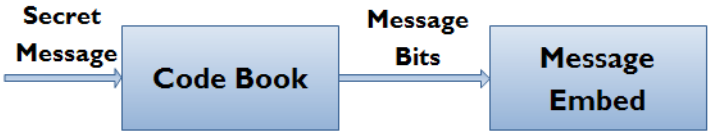


Figure 4.1: Message conversion and embedding

4.1.2 LSB Substitution

A bit replacement of the carrier image pixel's LSB is done using a simple logic. If the carrier bit matches the message bit, no replacement else modify the carrier bit appropriately by adding or subtracting 1 from the carrier image pixel. An example of the above implementation for a sample image is shown for a random bit stream. This illustrates the visual imperceptibility of the method.

But the implementation modifies the noise floor of the cover image. So simple attacks such as gaussian blurring, median filtering which essential change the noise

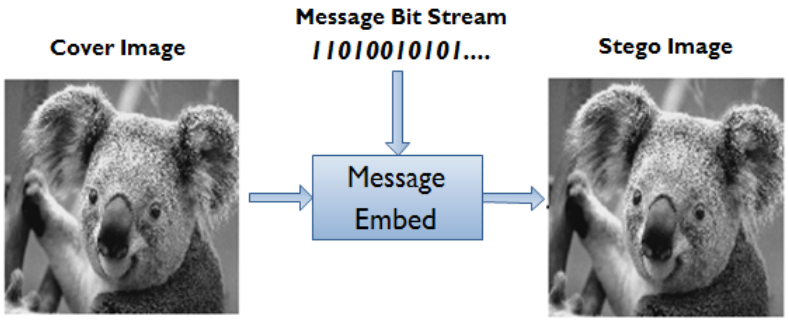


Figure 4.2: LSB example

floor will cause loss of embed information. Especially when the embedded message is pure text, changes in even a few pixels will lead to a completely different message being extracted. More complex versions of the spatial approach try to model to be embedded data in such a way to match the noise characteristics of the cover image. This will ensure that steganlaysis cannot easily detect the presence of the hidden message.

The retrieval of the hidden message with considerable accuracy is crucial to any steganographic method. The requirement of an embedding method which can survive such attacks leads us to the transform domain approach. The transform domain gives us access to the different frequency components present in the image. This implies that we can choose, in which regions of the image do we want to hide the messge data.

4.2 Transform Domain Implementation

The transform domain approach allows us to use lossy formats such as JPEG as well. Here, the cover image is transformed into its frequency domain using DCT or DWT. The embedding is done in the corresponding coefficients resulted due to the transform. As the modification is in the frequency domain the changes are spread out across all the pixels under consideration. Hence there is diffused change in the spatial domain which gives us better visual imperceptibility compared to the spatial domain approach.

4.2.1 DCT Based Methods

The cover image is converted into its transform domain as per the JPEG standard. The appropriate compression ratio is applied by using a suitable quantization table. At this stage we embed the message bit stream into the DCT coefficients

block-wise. These new modified DCT coefficients are the transform domain representation of the stego-image. We now take the inverse DCT transform and obtain the spatial domain stego-image. The common steps for methods implemented in this project are summarized below:

Cover Image 8x8 block

139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	160	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	161	163	158	158	158

Corresponding DCT coefficients

232.62	-1.03	-12.08	-5.20	2.12	-1.67	-2.70	1.32
-22.59	-17.48	-6.24	-3.15	-2.85	-0.06	0.43	-1.18
-10.94	-9.26	-1.57	1.53	0.20	-0.94	-0.56	-0.06
-7.08	-1.90	0.22	1.45	0.86	-0.07	-0.04	0.33
-0.62	-0.83	1.46	1.55	-0.12	-0.66	-0.60	1.27
1.75	-0.20	1.62	-0.34	-0.77	1.47	1.04	-0.99
-1.28	-0.36	-0.31	-1.46	-0.49	1.73	1.07	-0.76
-2.59	1.55	-3.76	-1.84	1.87	1.21	-0.56	-0.44

Figure 4.3: A block of a cover image and the corresponding DCT coefficients

Quantized DCT coefficients

15	0	-1	0	0	0	0	0
-2	-1	0	0	0	0	0	0
-1	-1	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4.4: Quantized DCT coefficients

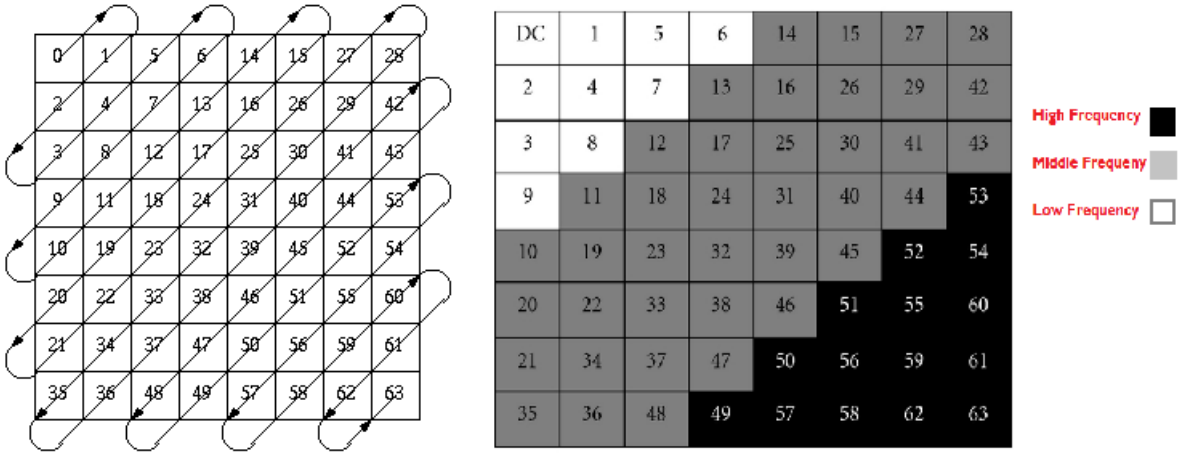


Figure 4.5: Zig-Zag scan of the DCT coefficients

Embedding in the DC coefficient

The DCT coefficients can be categorized into low, medium and high frequencies. The DC coefficient is the first coefficient we get on a zig-zag scan of the coefficients. It comprises as the name suggests the DC component of the pixel block. One of the very early implementations of DCT based steganography utilized this DC coefficient to embed the message data through LSB substitution.

DC coefficent Embed

16	0	-1	0	0	0	0	0
-2	-1	0	0	0	0	0	0
-1	-1	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Though this method gives good imperceptibility to the stego-image, since the embedding is done in only 1 out the 64 coefficients per block, this gives very limited capacity to this implementation. An example where the block in Fig. 4.4 is used for the DC coefficient embed is given here.

Uniform Embed

In this method we use all the 64 coefficients in each of the blocks for embedding. This will cause changes in all possible transitions seen in the pixel domain from slow varying backgrounds to sharp transition edges. We embed the bit stream in the order given by the zig-zag scan of the coefficients using LSB substitution.

This approach gives equivalent embedding capacity as the spatial domain approach but the stego-image is very noticeable due to the modifications made in many of the coefficients. We have implemented a low frequency (refer to Fig:4.5) coefficient and all 64 coefficient modification where the results are acceptable though still detectable in the former and drastically distorted in the later.

The reasons for why such an embedding method doesn't work is due to the quantization and de-quantization we have seen in the JPEG standard. We embed the data into the DCT coefficients between the lossy compression and lossless encoding stages. After the decoding is done we have to de-quantize the DCT

coefficients before performing the Inverse DCT to get the stego image. We use a standard quantization table to achieve this. The previously almost absent artifacts are prominently present in this case.

Uniform Embed

15	1	0	1	0	0	0	0
-2	-1	0	0	1	1	0	0
-2	-1	0	1	0	0	0	0
-1	0	0	0	0	0	0	1
0	1	1	0	1	0	1	0
1	0	0	0	1	1	0	0
1	0	1	0	1	0	0	0
0	0	0	0	1	0	0	1

Uniform Embed after De-Quantization

240	11	0	16	0	0	0	0
-24	-12	0	0	26	58	0	0
-28	-13	0	24	0	0	0	0
-14	0	0	0	0	0	0	62
0	22	37	0	68	0	103	0
24	0	0	0	81	104	0	0
49	0	78	0	103	0	0	0
0	0	0	0	112	0	0	99

Figure 4.6: De-Quantization effect

Lets suppose that a certain DCT block which has been embedded uniformly is represented on the right. Assume that the '1s' seen there are due to the LSB embedding and previously they were zero. When we de-quantize we would be multiplying each coefficient with the corresponding table value. This would make the '1s' increase to the corresponding table value whereas previously they would have remained at zero. This causes the corresponding frequency components to show up in the stego-image which cause a lot of unintended distortion. This is exactly what is being seen in the uniform embed case. To tackle this, we need a different approach to embedding.

'JSTEG'[5]

We have discussed previously that embedding in all the DCT coefficients in a block has effects across all the frequencies. This has proven to be not practical due to the large distortion in the stego-image. In JSTEG, we specify the DCT coefficients in which we avoid embedding.

JSTEG

16	0	-1	0	0	0	0	0
-3	-1	0	0	0	0	0	0
-1	-1	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Only the shaded coefficients have been used in the above example. We do not use the DCTs which are equal to '0' or '1' or '-1'. This will remove the unintended de-quantization distortion, but the trade off being severely reduced capacity. This will ensure that new frequency components are not introduced due to the embedding. Since most of the quantized coefficients tend to go to zero, we have limited choice of embedding.The '1' and '-1's are also avoided to ensure that the when embedding a zero message bit, doesn't cause an important frequency component to be lost.

Modified Quantization Table Method[6]

As the JSTEG method limits the embedding capacity very severely a different approach to tackle the de-quantization problem is needed. To avoid the large change in the embedded DCT coefficients upon de-quantization, we now modify the quantization table itself. The modification is done such that the coefficients in which embedding are done are not changed upon de-quantization. This is achieved by changing those range of corresponding table values to '1'.

Modified Quantization Table

16	11	1	1	1	1	1	1
12	12	1	1	1	1	1	1
14	1	1	1	1	1	1	1
1	1	1	1	1	1	1	62
1	1	1	1	1	1	103	77
1	1	1	1	1	104	113	92
1	1	1	1	103	121	120	101
1	1	1	98	112	100	103	99

DCT coefficients with modified table

15	0	-12	-5	2	-2	-3	1
-2	-1	-6	-3	-3	0	0	-1
-1	-9	-2	2	0	-1	-1	0
-7	-2	0	1	1	0	0	0
-1	-1	1	2	0	-1	0	0
2	0	2	0	-1	0	0	0
-1	0	0	-1	0	0	0	0
-3	2	-4	0	0	0	0	0

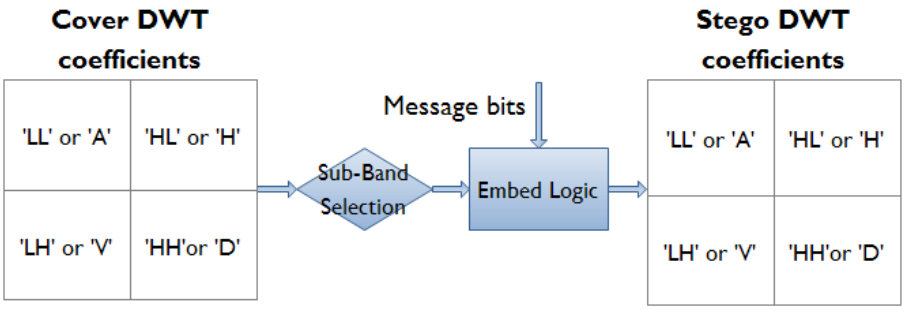
Figure 4.7: Modified Quantization Table

We have used from the 5^{th} to the 48^{th} DCT coefficients for embedding. As we are using more coefficients to embed the data per block we get more capacity than JSTEG. In this way the hidden message is preserved and the attacker trying to read the message after detection finds it very difficult as the quantization table required to safely extract the info is not at his/her access.

This advantage also is a disadvantage as the quantization table being used to decipher needs to be available at both ends or at least the range of frequencies being used to embed should be known. The bit rate is also increased as no quantization is done in certain frequencies which leads to more data to be sent for transmitting them. The stego-image formed with this table retains more of the original frequency components than the standard quantization making it much nearer to the original cover.

4.2.2 DWT Based Methods

The cover image is transformed using the Haar wavelet into its wavelet domain. It is the simplest of the wavelets which can be used for the DWT. We perform a 1^{st} level wavelet decomposition on the cover image which results in a 4 sub-band representation of the cover image. The message is embedded using LSB substitution in one or more of the sub-bands, in addition two other methods of embedding have also been implemented.



LSB substitution

Here we use LSB substitution in one or more sub-bands to embed the data. We compare among the different sub-bands to obtain the most suited sub-band. The approximate 'LL' or 'A' sub-band coefficients contain most of the data from the original image. It is normally avoided for embedding as it contains most of the image energy. Embedding in this sub-band, however, could increase robustness significantly. The 'HH' or 'D' sub-band is seen to be most optimal for this as it is difficult to perceive changes made in it. The edges and textures of the image are contained here and the human eye is not generally sensitive to changes in the high frequency regions.

Using only one sub-band is just utilizing a quarter of the total capacity. But more modification will result in more degradation of the stego-image. This also has been verified and illustrated in the results section.

'V-H' Sub-Band Embed[7]

We have implemented another method to embed the message stream into the wavelet domain. The embedding is done through a difference based criterion between the corresponding coefficients of 'H' and 'V' sub-bands. Below we describe

the logic used to modify the coefficients. As 2 bands are used giving effectively one band equivalent embed, the capacity is reduced in this approach.

For a message bit b , the corresponding horizontal and vertical wavelet coefficients are denoted by $H(x, y)$ and $V(x, y)$, respectively. The bit b is embedded by increasing the difference between $H(x, y)$ and $V(x, y)$. A threshold T is defined to take into account the visibility of the embed. The threshold also gives us the flexibility of addressing the DWT compression within certain limits. But using a higher threshold implies more modification made to the stego-image and consequently the quality suffers.

if $b = 1$, and $D_1 = H(x, y) - V(x, y) < T$ then

$$\begin{aligned} H^{`} &= H(x, y) + \frac{T - D_1}{2} \\ V^{`} &= V(x, y) - \frac{T - D_1}{2} \end{aligned} \tag{4.1}$$

else do nothing.
if $b = 0$, and $D_0 = V(x, y) - H(x, y) < T$ then

$$\begin{aligned} H^{`} &= H(x, y) - \frac{T - D_0}{2} \\ V^{`} &= V(x, y) + \frac{T - D_0}{2} \end{aligned} \tag{4.2}$$

else do nothing.

The extraction is simplified where the comparison of $H(x, y)$ and $V(x, y)$ gives the embedded bit b. If $H(x, y)$ greater than $V(x, y)$ then $b = 1$ else $b = 0$.

Sub-Band Embed using Haar transform properties[8]

The detail sub-band is the most preferred for embedding as it difficult to perceive changes made in it. The edges and textures of the image are contained here and the human eye is not generally sensitive to changes in the high frequency regions. We have implemented an embedding technique which utilizes the properties of the Haar wavelet to hide secret data. We embed the data into the presence or absence of a fractional part of a coefficient.

The fractional parts of the coefficient are dependent on the pixel values for the haar transform. The sub-band pairs 'HH-HL'and 'LL-LH' are connected in a sense where their corresponding coefficients are of the same parity, i.e. either both have a fractional part or both do not have any, due to the nature of the haar

transform. This requires us to embed the same data in both the connected sub-bands to maintain parity which is essential for good reconstruction of the extracted message. Embedding only in one sub-band, say 'HH' alone will give less than 50 percent reconstruction. Whereas, redundant embedding in both connected sub-bands 'HH' and 'HL' will give almost 70 percent extraction and in all 4 sub-bands to maintain the parity gives almost complete extraction.

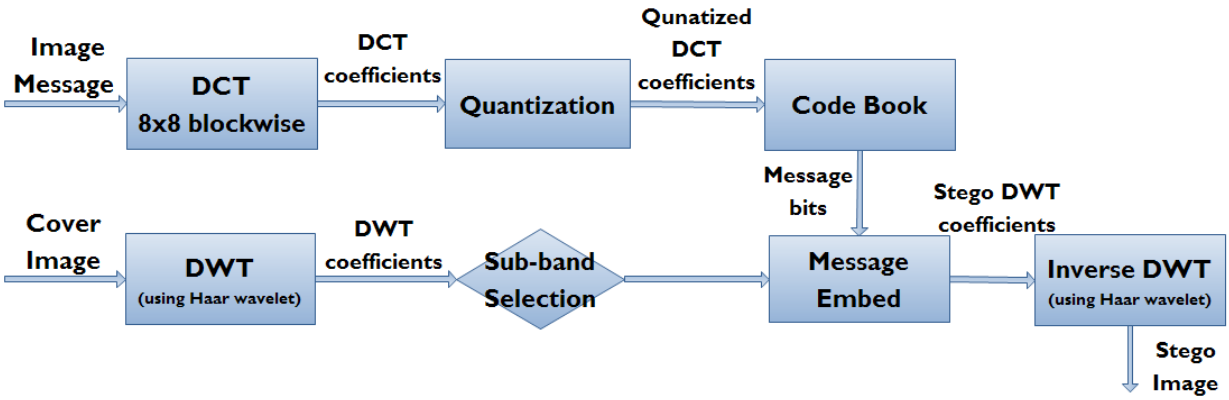
The embedding logic is given below. $< p >$ denotes the fractional part of p where $D(x, y)$ and $D_s(x, y)$ denote the detail coefficients('HH' and 'HL') of the cover and stego images.

$$\text{if b} == 0 \ \& < D(x,y) > != 0, \text{ then } D_s(x,y) = D(x,y)$$
$$\text{if b} == 0 \ \& < D(x,y) > = 0, \text{ then } D_s(x,y) = sgn(D(x,y)) * (D(x,y) - \tfrac{1}{2})$$
$$\text{if b} == 1 \ \& < D(x,y) > != 0, \text{ then } D_s(x,y) = sgn(D(x,y)) * (D(x,y) - \tfrac{1}{2})$$
$$\text{if b} == 1 \ \& < D(x,y) > = 0, \text{ then } D_s(x,y) = D(x,y)$$

The extraction of the embedded bit is done by checking the fractional part of the detail coefficient. If fractional part of the stego coefficient is equal to zero then message bit is '1' else '0'.

4.2.3 DCT & DWT Combination[9]

For purely an image type message steganography we have implemented an approach which transforms the message as well. This transformed representation is converted into the corresponding bit stream which is embedded into the cover image, also in the transformed domain. The rationale behind this method is that even if some of the embedded bits are lost due to random attacks, we can reconstruct the image message with relatively better accuracy than the previously implemented methods.



As the transform domain coefficients are lossless representation and modifying

some of them will change the image properties, but the redundancy present in an image can be exploited to overcome bit info loss during attacks. The combination of DWT for the cover and DCT for the image has been implemented.

The message image is blockwise transformed using DCT. The coefficients are converted into a bit stream which is embedded into the DWT coefficients of the cover image. The results are presented in the next chapter.

Chapter 5

Results & Analysis

5.1 Spatial Domain Approach

Below a successful extraction of the image message is shown and further along the performance metrics for the various message types are tabulated. The cover images used are BMP and stego-image is also stored in the same lossless format. The following lists the associated components of the steganography technique:

Object	**Comment**
Cover Image	BMP format, 512 x 512, luminance plane only
Maximum Capacity	32768 bytes
Text Message	178 characters, each has a 8-bit ASCII code
Image Message	BMP format, 128 x 128 (Bit Depth: 8)
Random Sequence	1024 bytes

Cover	**Message**	**BER**	**MSE**	**$\mathbf{PSNR_{dB}}$**
Lighthouse	Text	0.0026	0.0026	73.17
	Image	0.2511	0.2511	53.27
	Random	0.0156	0.0156	65.33
Koala	Text	0.0027	0.0027	73.06
	Image	0.2494	0.2494	53.45
	Random	0.0156	0.0156	65.49

Table 5.1: Results : Spatial Domain Approach

In this approach, LSB substitution is a modification in the spatial domain which directly translates in to the number of pixel modified given by the BER. As the modification is also in the LSB, the BER and MSE are numerically equal in this case. From the Table 5.1, we can infer the following :

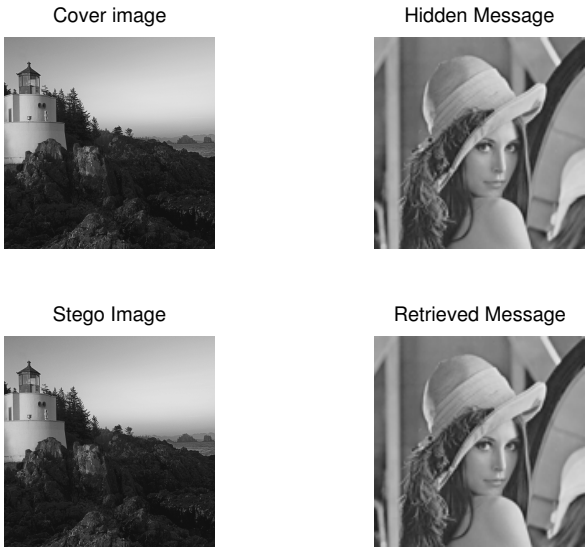


Figure 5.1: LSB substitution for an image message

i. More the embedded data, more is the distortion between the cover and stego i.e. adjustable trade off between capacity and stego-image quality

ii. Suitable for embedding text directly as the change is very difficult to perceive at those values of the metrics.

iii. The random sequence embedded while much larger in size compared to the text embedded gives comparable results. This hints towards encryption of data with noise floor emulation may give better results than plain embed.

5.2 Transform Domain Approach

The Transform domain approach is meant to increase the imperceptibility of change due to embedded data in the cover image. We shall discuss how the results reflect this and any other inferences which can be made.

5.2.1 DCT Based Methods

The DCT based approach allows embedding in the frequency domain of the image. A LSB modification of the DCT coefficients in a block results in the change of pixels values across the block, hence the change is spread out and more difficult to detect.

Here we have used JPEG format cover images for the implementation which are converted from RGB to YCbCr plane as discussed in chapter 3.

Embedding in the DC coefficient

The maximum capacity for a 512 x 512 cover image is just 4096 bits of data, one bit for each 8 x 8 DCT block. The embedding which can be done is very minimum i.e. severely restricted capacity. Very little change is visible in the stego-image as a result.

Cover	Message	BER	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	Random	0.0076	0.03	63.47
Koala	Random	0.0076	0.03	63.34

Table 5.2: Results of Embedding only in the DC coefficient

Uniform Embed

In this method, the embedding is done across all the 64 coefficients in a block. This includes all the frequency ranges and the effects of LSB modification are seen throughout the image. This gives a max capacity as previously stated but the stego-image undergoes severe distortion. The results for both, all 64 coefficients embed and a low frequency coefficients '0' to '9' are tabulated.

Figure 5.2: Distortion is evident even in low frequency(0-9) range embedding

Cover	Message	BER	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	Image	0.500	1108.9	17.83
Koala	Image	0.500	1110.2	17.70

Table 5.3: Results of embedding in all 64 coefficients

Cover	Message	BER	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	Image	0.8000	11.02	37.85
Koala	Image	0.7993	11.07	37.70

Table 5.4: Results of embedding in low frequency coefficients

'JSTEG'

The 'JSTEG' avoids '1's, '0's and '-1's to embed the data. For a cover image like *Koala* we can expect more number of non-zero coefficients as it contains many rough and sharp regions consisting of high transition components. This is clearly seen through the max capacity available for a smoother image of *Lighthouse* and much higher capacity available for *Koala*.

As expected, with more embedding the changes in the stego-image are more apparent. Comparing JSTEG with previous methods, it can be seen that it gives much better performance in terms of imperceptibility at the cost of capacity. Choosing a suitable cover image is very important for good results with this method.

Cover	Max.Capcity	Message size	BER	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	17616 bits	32 x32	0.3875	5.72	40.70
		40 x40	0.5977	7.74	39.39
Koala	31396 bits	32 x32	0.2231	5.58	40.68
		40 x40	0.3252	9.30	38.46

Table 5.5: JSTEG results

Modified Quantization Table Method

The maximum capacity for a cover image of size 512 x 512 is given as 180224 bits for the embedding range of 5^{th} and 48^{th} coefficients. The results reflect the fact that this method gives better performance as well as much higher capacity. The trade off of a modified quantization table is certainly justified. Larger size of the embedded message results in more distortion of the stego-image as reflected in the results.

Cover	Message size	BER	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	64 x64	0.1780	0.06	59.54
	128 x128	0.7112	0.24	53.53
Koala	64 x64	0.1780	0.06	60.62
	128 x128	0.7112	0.24	54.60

Table 5.6: Modified quantization table results

5.2.2 DWT Based Appraoch

The DWT approach gives us more control over the embedding region. The resolution properties of the transform allow us to specifically embed the data as per the requirement. The capacity offered depends on the effective number of sub-bands being used for the embed with each giving one-quarter of that offered by spatial domain approach.

Object	Comment
Cover Image	JPEG format, 512 x 512, RGB
Maximum Capacity	8192 bytes (for single sub-band)
Image Message	BMP format, 64 x 64 (Bit Depth: 8)

LSB substitution

Each of the sub-bands is individually embedded and the results compared. We have also used multiple bands to embed data and compare the trade off between capacity and stego-image quality.

Cover	Message	A	H	V	D
Lighthouse	Image	56.26	56.48	56.43	56.59
Koala	Image	56.59	56.46	56.50	56.47

Table 5.7: $PSNR_{dB}$ values for LSB substitution in different sub-bands

The table 5.7 shows that the LSB modification in the individual sub-bands gives better results than the previous approaches but does not vary much comparing across the sub-bands. Usually 'A' contains most of the features of the cover image and is avoided for embedding. The 'D' sub-band is however gives very good imperceptibility as changes made in it cannot be perceived by human vision.

We now embed in multiple bands and see the trade off between capacity and stego-image quality. The result with 4 sub-bands may be due to the overlap of changes made to the pixels canceling each other.

Cover	No.of bands	Capacity	MSE	$\mathbf{PSNR_{dB}}$
Lighthouse	1	$\frac{1}{4}^{th}$	0.13	56.26
	2	$\frac{2}{4}^{th}$	0.15	55.81
	3	$\frac{3}{4}^{th}$	0.28	52.82
	4	max	0.25	53.29

Table 5.8: Comparing capacity and stego-image quality

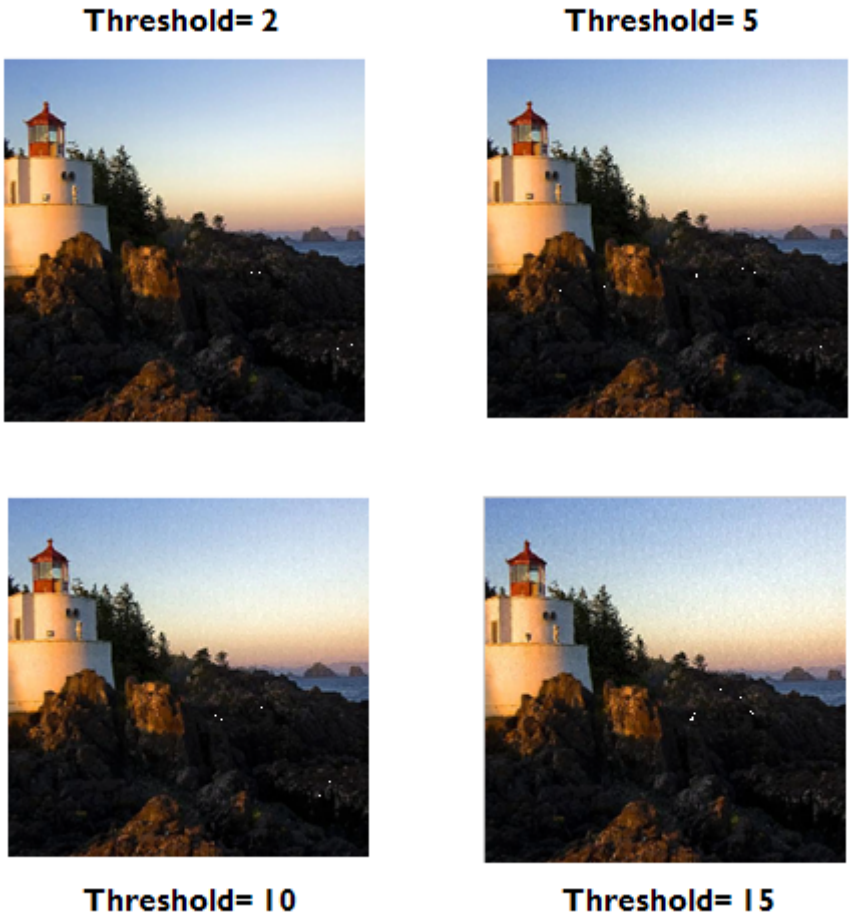
'H-V' Sub-Band Embed

We use a difference based embedding technique here to modify the coefficient data in the sub-bands. The results for different thresholds with the corresponding stego images are shown below. Better performance can be obtained using more complex wavelets, as to the usage of the Haar wavelet, though simple in nature does not give good image reconstruction results. Implementations using other wavelets for a similar approach have given better results[7]

Cover	Threshold	MSE	$PSNR_{dB}$
Lighthouse	2	23.87	33.60
	5	25.82	33.26
	10	31.38	32.41
	15	39.94	31.37

Table 5.9: Difference based embed in 'H-V'sub-bands

Though the performance metrics do not change very significantly, the stego-images below clearly exhibit distortion as the threshold increases.



Sub-Band Embed using Haar transform properties

Embedding in a single sub-band will give less than 50 percent extraction of the message as discussed previously in Section 4.2.2. We shall see the effect of redundant embedding of the message incrementally in all the sub-bands and the amount of extraction possible.

Cover	Sub − Bands	MSE	$PSNR_{dB}$
Lighthouse	D	0.01	67.17
	D&H	0.02	64.53
	D,H&V	0.16	55.25
	D,H,V&A	0.16	55.42

Figure 5.3: Extent of extraction possible for different number of sub-bands used

5.2.3 DWT + DCT Based Approach

This approach has the hidden message in its DCT domain. This will help reconstruct the image with decent accuracy even if some of the DCT coefficients's bits are lost. The effect would be similar to quantization for that particular coefficient. The trade off is that we need to additionally convert the message into its DCT domain.

Cover	Embed Logic	MSE	$PSNR_{dB}$
Koala	LSB	0.11	56.82
	H-V	27.82	32.97
	Haar properties	0.16	55.34

Table 5.10: DWT of cover embedded with DCT of message

Plots summarizing the results for DCT & DWT

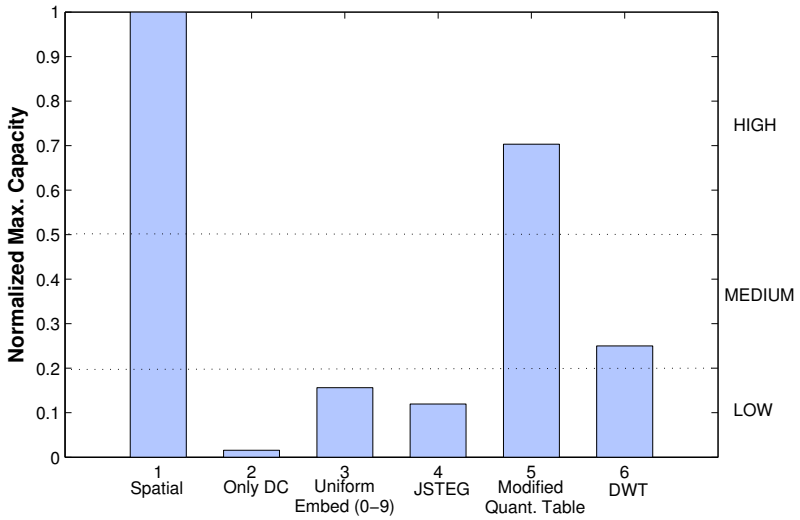


Figure 5.4: Capacity bar graph

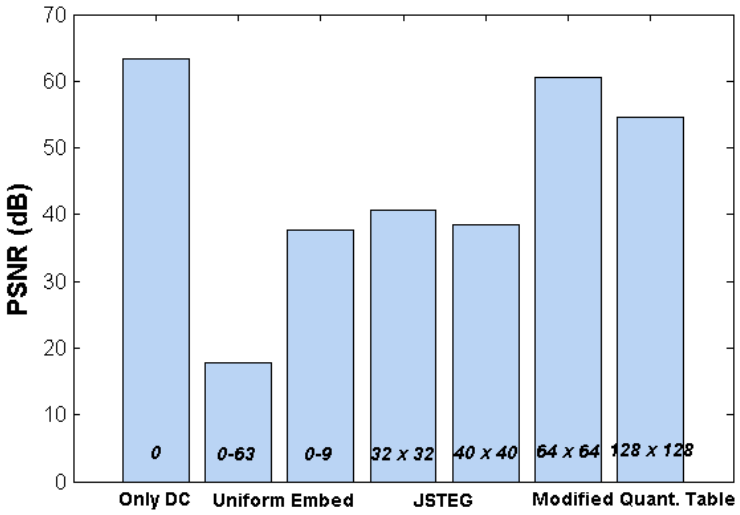


Figure 5.5: PSNR plot for DCT methods

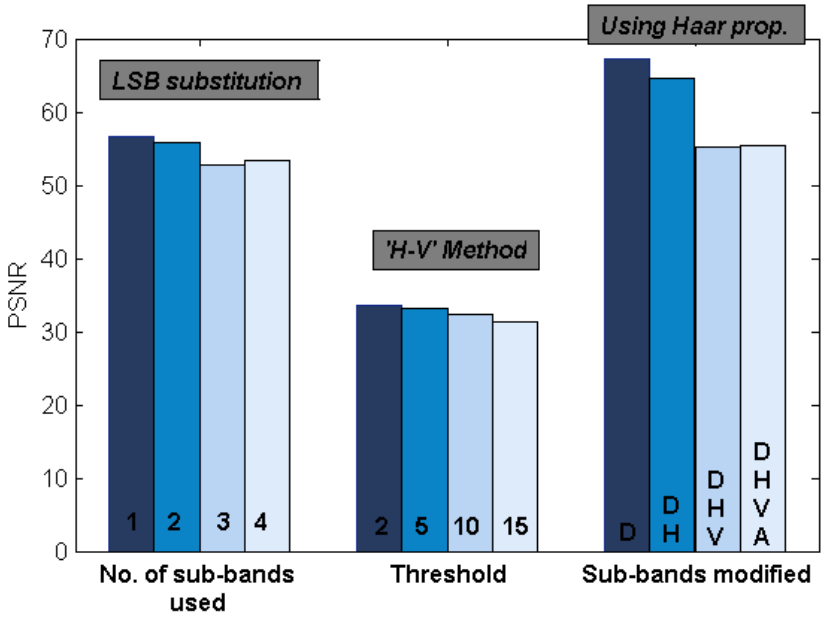


Figure 5.6: PSNR plot for DWT methods

Chapter 6

Conclusions

Summary of Results

From the Fig. 5.4 we can see that the spatial domain approach gave high capacity, the DCT approach gave a range of capacities specific to the method used with the modified quantization table giving the highest. The DWT approach gave medium to high capacity for the 3 methods depending on the effectively number of sub-bands used.

The spatial domain approach done on lossless image formats, is useful for pure text or encrypted message bits, where even a few errors in the extracted bits is not acceptable as it changes the message significantly. It gives good performance but is very sensitive to attacks and steganalysis.

In the transform domain approach, we conclude that they are more suited to image type messages where slight changes in the extracted bits is acceptable. The DCT methods, though computationally tasking have given good performances with the modified quantization table being the best. The DWT based methods gave more control and flexibility over the embed based on the decomposition levels used. The 'H-V' implementation gave control over extraction possible even under compression attacks. Utilizing the properties of the Haar wavelet, good performance was obtained though redundant embedding was required.

The combination approach of DWT and DCT gave similar performance to DWT based methods additionally giving a more reliable way for reconstructing the message after an attack.

Concluding Remarks

Implementing the various image steganography techniques and comparing their results leads us to conclude that the technique used needs to be very specific to the user requirement. There can never be a single solution to many steganography needs as each method has its own benefits and associated drawbacks.

Bibliography

[1] Dr. Ekta Walia, Payal Jain and Navdeep, *An Analysis of LSB & DCT based Steganography*, Global Journal of Computer Science and Technology, Vol. 10 Issue 1, April 2010

[2] Shailender Gupta, Ankur Goyal and Bharat Bhushan, *Information Hiding Using Least Significant Bit Steganography and Cryptography*, I.J.Modern Education and Computer Science, June 2012

[3] V. Lokeswara Reddy , Dr. A. Subramanyam and Dr.P. Chenna Reddy, *Implementation of LSB Steganography and its Evaluation for Various File Formats*, Int. J. Advanced Networking and Applications Vol. 2, No. 5, 2011

[4] Hsien-Wen Tseng and Chin-Chen Chang , *Steganography Using JPEG-Compressed Images* , Proceedings of the Fourth International Conference on Computer and Information Technology, 2004

[5] Niels Provos and Peter Honeyman, *Hide and Seek: An Introduction to Steganography*, The IEEE Computer Society, 2003

[6] Chin-Chen Chang, Tung-Shou Chen and Lou-Zo Chung *A steganographic method based upon JPEG and quantization table modification*, Information Sciences 141, 2002

[7] Nagham Hamid, Abid Yahya, R. Badlishah Ahmad and Osamah Al-Qershi *Characteristic Region Based Image Steganography Using Speeded-Up Robust Features Technique* , 2012 International Conference on Future Communication Networks

[8] Vladimr BNOCI, Gabriel BUGR, Duan LEVICK *A Novel Method of Image Steganography in DWT Domain* , 2011 IEEE

[9] A.A.Al-Saffar *Proposed Steganography Method Based on DCT Coefficients* , Jouranl for Pure & Applied Sciences Vol. 24(3) 2011

[10] Robi Polikar *The Wavelet Tutorial*, Rowan University.