

Securing Confidential Data using Dual Image Steganography by maintaining Data Integrity using Huffman Parity Algorithm

MSc Academic Internship
Cyber Security

Ashish Shetty
Student ID: x18183077

School of Computing
National College of Ireland

Supervisor: Prof. Imran Khan

**National College of Ireland
Project Submission Sheet School
of Computing**



Student Name:	Ashish Shetty
Student ID:	X18183077
Programme:	MSc in Cybersecurity
Year:	2020
Module:	MSc Internship
Supervisor:	Mr. Imran Khan
Submission Due Date:	17/08/2020
Project Title:	Securing Confidential Data using Dual Image Steganography by maintaining Data Integrity using Huffman Parity Algorithm.
Word Count:	6278 Words
Page Count:	32

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature:	Ashish Shetty
Date:	17-08-2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	Q
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	Q
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	Q

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Securing Confidential Data using Dual Image Steganography by maintaining Data Integrity using Huffman Parity Algorithm.

Ashish Shetty

X18183077

Abstract

With the advancement in technology, over the years, the amount of data has increased exponentially, and with growing data, security has become a significant concern. Steganography plays a vital role in hiding the data within an image, audio, and video file. It has been seen that the mere use of Steganography with sensitive data can be easily decrypted, or degrading methods or techniques can be used to degrade the pixels in an image. So, in order to maintain the confidentiality and integrity of the data, an enhanced version of Steganography and cryptography comes in the picture. In this paper, we have proposed a technique of hiding and securing sensitive data using Dual Image Steganography, which plays a significant role in securing secret/sensitive data within two wrapped images, which uses 256-bit AES Encryption for ciphertext using Haar Discrete Wavelet Transform (HDWT) for hiding the data in one of the high-frequency sub-data which comprise of four bands and has excellent space-frequency localization property, which increases the impalpability and the hiding capacity, while implementing the Huffman Parity Coding Algorithm which acts as a lossless compression method so that the data should not get lost and lose its quality. The proposed method can help in achieving a secure transmission of sensitive data between the end-users.

Keywords: Dual Image Steganography, 256-bit AES Encryption, Haar Discrete Wavelet Transform, Huffman Parity Algorithm.

1. Introduction

With the growing number of users in the 21st century, the protection of user's data has become a significant concern. The organization faces severe challenges regarding data protection. With the fast-paced, growing technology, improving and developing security measurements has become a significant challenge to protect it from attackers. End to End transmission of sensitive data, storing and securing sensitive data, achieving confidentiality and integrity are some of the measurements that can be achieved with Steganography. Strategies that are stable, such as the combination of Steganography and Cryptography, provides an excellent way to secure user data. With enough research on Steganography, researchers have found that Steganography and other combinations of security approaches have such as 256-bit AES Encryption, Haar Discrete Wavelet Transform, Huffman Parity Coding is an excellent way to achieve data security.

Cryptography is a secure technique that is used for secure and reliable communication of data that is being encrypted and sent over various communication protocols so that an attacker cannot interpret what data is being sent even if the data gets leaked. Through cryptography, users can achieve goals known as [1]:

- Confidentiality
- Integrity
- Authentication
- Non – Repudiation
- Access Control

Use of Advanced Encryption System (AES) is a symmetric encryption algorithm that was designed to work both in Hardware and Software systems efficiently. Advanced Encryption System (AES) supports block length of 128 bits, 192 bits, and 256 bits. Advanced Encryption System (AES) is used to encrypt plain text to ciphertext. Advanced Encryption System (AES) uses a single key, also known as the secret key that is used for encryption and decryption. The Advanced Encryption System (AES) consists of 14 rounds of processes that involve mixing, substitution, and transposition of the input plaintext to convert into a final output.

With 256-bit AES Encryption, Haar Discrete Wavelet Transform (HDWT), and Huffman Parity Coding, there is a need for additional security methodology with Steganography. Steganography consists of a variety of different techniques, which include Textual Steganography, Audio Steganography, Video Steganography each and every Steganography methodology, provides some security that varies from one to another.

Even though Steganography has its advantages, but there are multiple approaches, such as the Least Significant Bit (LSB), Dual Key Approach, etc. Among all of them, Haar Discrete Wavelet Transform is efficient in hiding data because Haar Discrete Wavelet Transform has an excellent space-frequency sub-bands known as LL1 (Approximation Coefficients), LH1 (Vertical Details), HL1 (Horizontal Details), and HH1 (Diagonal Details) mainly used for 2D filter image processing. Haar Discrete Wavelet Transform is a frequency domain technique that provides an absolute secure image that is hard to detect the secret data. [2].

Coming to Huffman Parity Coding, it is vital that the secret data must not be lost its integrity, so Huffman Parity Coding is applied to the stego image in order to create a secure image where the bits are not lost during transmission of the image.

How can we secure confidential data using Dual Image Steganography using 256-bit AES Encryption, and (HDWT (Haar Discrete Wavelet Transform Algorithm)) by maintaining the integrity of the original encrypted data?

The rest of the thesis is presented according to the following section, in Section 2, this section involves a literature review, where a comparison of previous works needs to be studied and analyzed based on Steganography. Section 3 involves Research Methodology, where a thorough evaluation needs to be made about the research procedure that we have made. Section 4 involves Design Specification, where a clear representation of the implementation of the project needs to be discussed, such as architectural aspects of proposed Steganography implementation; it involves an explanation of how different algorithms will work, such as Haar Discrete Wavelet Transform (HDWT), and Huffman Parity Coding. Section 5 includes the implementation of the proposed Steganography project. Section 6 involves Evaluation, and Section 7 involves Conclusion and Future Work.

2. Related Work

An extensive comparison is made in this literature review on Steganography and other proposed Algorithms such as 256-bit AES Encryption, Haar Discrete Wavelet Transform, Huffman Parity Algorithm. We have analyzed and compared the previous methodologies made by researchers, which is outlined in their work.

2.1. Least Significant Bit (LSB)

According to *A. K. Singh and J. Singh* [3], Least Significant Bit is a novel methodology to hide secret data inside an image, video, or audio file. According to the authors, using the LSB technique, sensitive data is hidden in the spatial data domain. The Payloads (data) are embedded into the cover image using the Least Significant Method (LSB) to produce a Stego Image. The LSB technique's methodology uses grayscale consisting of value one, but LSB has the advantage of storing a pixel value of 3 within an image file. LSB embedding may even be applied in particular data domains, for example – embedding a hidden message into the color values of RGB bitmap data or the frequency coefficient of a JPEG image. The LSB embedding technique on the stego image can be applied using a variety of data types, but due to this very reason, the image can get distorted, and the image file may get corrupt. According to the authors [4], After using several grayscale images to perform LSB, the technique follows some steps such as the images were converted into a matrix value of the following bits. After performing the LSB technique on those images, the results showed that the values, also known as the average intensity value, got changed

when 100 characters were embedded in the image but compared to the DWT technique, **DWT performed on the images provided better average brightness more than LSB technique.** According to the authors K. B. Raja, C. R. Chowdary, K. R. Venugopal, and L. M. Patnaik, [5] they performed a Least Significant Bit (LSB) embedding on two images which consisted 24 bits and 8 bits of images. The LSB technique was first embedded in the 24-bit image where information of 3 bits were stored in each of the pixels, and then the LSB technique was performed on the 8-bit image, where only information of one bit could hide. **Following the result, various types of testing were made, such as compression of the image, which required a more significant bandwidth. It affects the image and may become noticeable.**

2.2. Haar Discrete Wavelet Transform (HDWT)

According to the authors [6], A Wavelet transform provides better capability on frequency time. In a 2D-Haar Wavelet Transform, the information is passed through low and high pass filters, which consists of low and high frequencies, respectively. The DWT Analysis consists of two functions consisting of two filters, namely high and low pass. The manner that decomposition follows consists of separability. Haar Wavelet basically works by differences and elements. According to the authors [7], A comparison has been made between the Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) and DWT has been more secure and suitable to implement because it follows the Multi-Resolution Analysis (MRA). In the Wavelet transform, there are two types Wavelet Transform of known as Continuous Wavelet Transform (CWT) and Discrete Wavelet Transform (DWT). In DWT, it uses a filter bank to analyze the given pixels, and then it rebuilds the signal. In Haar Discrete Wavelet Transform if the 2D wavelet transform is only applied only on one image, it will decompose the image file into non-overlapping four sub-bands which as follows, A = Approximate, H = Horizontal, V = Vertical, and D = Diagonal or also known as the LL1 (A), LH1 (V), HL1(H), and HH1 (D). The 2DHDWT provides four sub-bands known as perpendicular, calculation, transverse, and parallel on an image.



Fig 1. Decomposition using Haar Discrete Wavelet Transform [8]

Haar Discrete Wavelet Transform the A and H are the first two sub-bands, and V and D are the high pass filters. **The Haar Wavelet is an excellent technique to embed data in an image because of the decomposition and reconstruction algorithm. In this paperwork, the authors [8] performed a multi-resolution decomposition on an image. After performing the HDWT on the image, it was found that the HDWT has its space-frequency, which has excellent space-frequency property.**

Proposed Method:

1. In the first step, the decomposition of the colored cover image is decomposed within 3 color planes known as Red (R), Green (G), and Blue (B).
2. Using DWT, the decomposition of each color plane is embedded into the four sub-bands.
3. Here, in each plane of the HH band, DCT is applied.
4. The dispersion of secret data is embedded into binary images, using components that are high in frequency, which is using a 2D sequence.
5. The inverse transformation is applied once the bits are embedded into secret images so in order to get the spatial form.
6. The stego image is generated using the three-color planes.

2.3. Dual Image Steganography

According to the authors, [9] has proposed a Dual Image Steganography for hiding data inside an image for securing the communication in order to protect it from insecure channels. The basic model which is proposed consists of a secret data that is converted using various cryptographic algorithms like Data Encryption Standard (DES), Rivest Shamir Adelman (RSA), and Advanced Encryption System (AES). Once the secret data is embedded in the image, it forms a stego image. Using another cover image which fused with the stego image gives rise to Dual Image Steganography. According to the authors, if an attack occurs on the stego image, the hidden data remains scrambled within the image, and the secret data cannot be recovered or understood by the attackers or eavesdropper. In this research paper, the authors have proposed a Dual Image Steganography using Least Significant Bit [10]. The proposed method uses two images one is reference images, and the other is the cover image. Mainly LSB method is used to secure the dual image steganography where it works by the reference image gets divided into a set of blocks with a different set of block codes. Using the embed key, all the embedding parameters, and the total number of blocks are stored in that embedding key. The secret message's bit pairs are encoded in different blocks of bit pairs. Authors also include different goals of image steganography to achieve goals such as Capacity, Imperceptibility, and Robustness. In this research paper,[11], the authors have implemented a Dual Image Steganography, which provides a high level of payload capacity and better transparency. In this proposed method, two cover images are chosen, and two stego keys are that are different from each other. A stego key length of 10 bits is chosen, which is a mixture of alphabets and numerical.

Here the methodology involves the use of LSB technique where 4 bits of LSB embedding algorithm is embedded with stego key 1 out of which stego image 1 is generated. The stego image which was generated is now embedded inside the cover image 2. Here, the LSB Algorithm of 4-bits using stego key 2 due to which a final stego image is achieved. Authors have achieved quality images because two planes were used instead of three planes to maintain the quality of the image. The image quality metrics such as Mean Square Method (MSE) and Peak Signal to Noise Ratio (PSNR) were used to check the quality parameters of the images. **High PSNR value provides a better quality of an image, and MSE shows dissimilarity between compared images.**

2.4. Huffman Parity Coding in Steganography

In this paper [12], the authors have proposed a method of securing secret data using Steganography and Huffman Parity Coding. In this paper, the authors state that using Huffman Parity Coding is a lossless compression technique used in order to achieve a lossless image without losing a single bit. Huffman Parity Coding is a compression technique is based on the frequent occurrence of all the symbols consisting of a file. Huffman Parity Coding works by constructing a tree that follows the bottom-up approach.

Working:

1. Every character value consists of a frequency within a message.
2. The frequency sorts the list by making two elements that are the lowest into leaves; due to this, a parent node is created.
3. Two of the elements get removed from the list, and then a new parent node gets inserted in the list because of the frequency value.
4. The two elements that are lowest by repeating the loop.
5. Repetition continues until only one element is left on the list.

Symbol	Frequency	Probability
A	5	$5 / 11 = 0.45$
B	2	$2 / 11 = 0.18$
C	1	$1 / 11 = 0.09$
D	1	$1 / 11 = 0.09$
E	2	$2 / 11 = 0.18$

Fig 2. Frequency Table

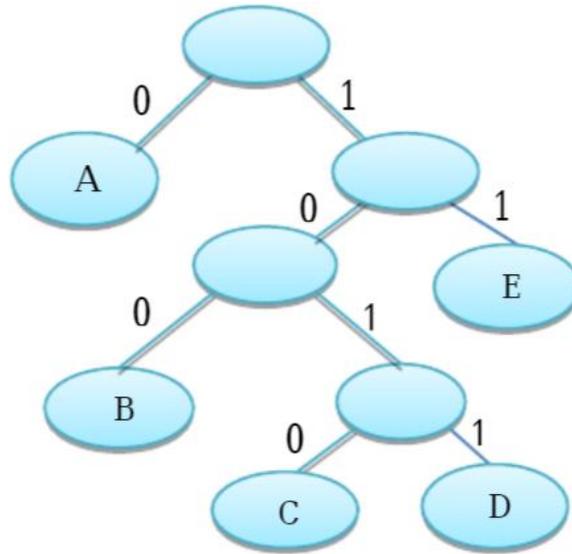


Fig 3. Representation of Huffman Tree

Huffman Parity Code is generated by traversing the tree to the value the user requires, In the left-hand branch, the tree can be traversed to the value of 0 output whenever the user takes a left branch and 1 in the right branch. Once the Huffman tree finishes binding the text, a codeword is generated, which is 23 bits. The message consists of ASCII values in 88 bits (11 characters x 8 bits). **Hence, Huffman Parity Algorithm saves more than 25% of the size of the message, but this paper fails to show the Entropy value, which is required to compress a JPEG and PNG Value. The entropy coding scheme is applied for image compression to check how good the Huffman Parity Coding has worked on the image. The Entropy Value and Efficiency Value should be checked on the Stego Image to check how good the compression ratio of Huffman Parity Code has worked, and it consists of bits rate value, which defines the calculated number which is transmitted through Noise Channel.**

3. Research Methodology

The research methodology outlines a secure method to secure sensitive data in order to achieve a secure transmission and confidentiality of data. The whole prototype is built to ensure that the image file consisting of secret data using 256-bit AES Encryption reaches the endpoint without losing its original data using Huffman Parity Coding. In order to add an extra layer of security, Dual Image Steganography is applied to enhance the security and to protect the image from various stego attacks. In this research paper, [13], the author has demonstrated various types of steganographic attacks such as visual attacks and statistical attacks in order to show how steganographic systems and files are affected by those attacks. Lossy Compression is used to embed data in an image file, which results in unwanted noise ratio, which can lead to degradation of an image. Due to this, we have implemented Huffman Parity Coding, which is a lossless compression method that does not lose data once the image is decompressed. In this research paper [14], the author has stated that various methods have been demonstrated to ensure the quality of the image should be preserved, which carries the embedded data to cover the marginal statistics such as “F5”. To encode the message, instead of using LSB to flip to secure or encode the message bits, “F5” can be used to increase or decrease coefficient values to maintain the histogram but according to the author “F5” is unreliable to use as it changes the histogram’s coefficient value when cropped from the image. So, in order to avoid different attacks and to avoid detection, Haar Discrete Wavelet Transform (HDWT) is used. D. P. Mahajan and A. Sachdeva [15] have compared and analyzed the security variation of Advanced Encryption Standard (AES) in terms of data security and how efficient and safe it is. According to the authors, AES consumes fewer resources when compared to DES and RSA encryption algorithms. AES encryption has proven to be the safest and secure algorithm.

3.1. Comparison between Steganography and Cryptography

Steganography	Cryptography
Steganography is a method of hiding secret data inside a digital file such as Image, Audio, and Video file.	Cryptography is a technique of converting plain text into a ciphertext so that an eavesdropper should not understand the encrypted message.
Steganography results in securing data without getting noticed.	Cryptography results in a combination of various gibberish values that cannot be understood by reading it.

The primary goal of Steganography is to prevent attackers from exploiting the secret data.	The primary goal of Cryptography is to prevent attackers from obtaining information that was transmitted via medium and which is confidential.
Steganography, in conjunction with Cryptography, becomes imperceptible to the attackers.	Cryptography provides robustness, confidentiality, and integrity. The encrypted message cannot be perceived or broken easily.

Table 1: Comparison between Cryptography and Steganography

3.2. Comparison and Evaluation of Image Steganography using Various Algorithm

Steganography Methods	Cover Media	Embedding Techniques	Advantages
Hiding the Image	Image		
Haar Discrete Wavelet Transform (HDWT)		HDWT works by embedding the secret data in one of the planes as it provides excellent space properties.	The wavelet's coefficient can be altered accordingly with the noise only when the level is tolerable.
Huffman Parity Coding		Huffman Parity Coding is applied to the image using the priority queue.	While decoding the image, the embedded data gets back into the original image without losing a single bit because of the Entropy Value.

Table 2

3.3. Advanced Encryption System (AES)

With the growing number of private data, securing those private data has become an important task for organizations. According to the authors, Encryption plays an essential role in securing private information. Technically, computation is an enormous tool to scramble all the plain text to ciphertext. Cryptography has several encryption systems such as Rivest-Shamir-Adleman (RSA), Data Encryption System (DES), Blowfish Encryption System, and Twofish Encryption System, etc. Over the years, these encryption methods were found to be prone to attacks due to which NIST chose Advanced Encryption System (AES) in replacing previous encryption methods. Before Advanced Encryption System (AES), DES was widely used cryptographic methodology and was designed by IBM, but in the early 1990's it was discovered that DES encryption was prone to attacks. Advanced Encryption System (AES) was developed in order to produce a reliable and secure encryption system for the U.S government. Advanced Encryption System (AES) has 3 bits of key sizes, such as 128-bits, 192-bits, and 256-bits.

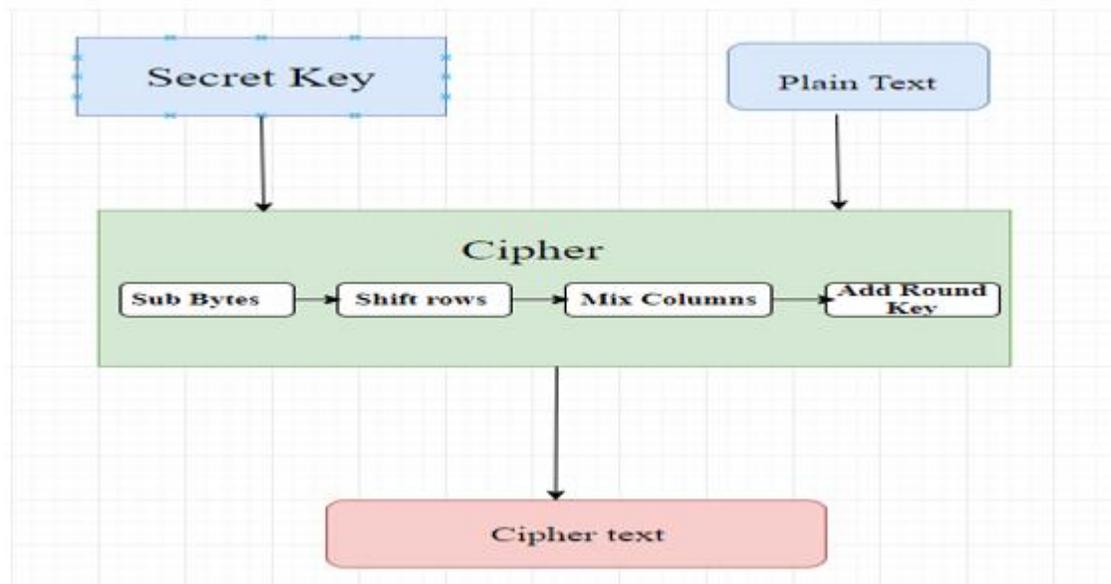


Fig 4. Block Diagram of the AES Algorithm

Advanced Encryption System (AES) is a symmetric-key algorithm, which means it uses the same key or single key to encrypt and decrypt data. **The design principle of the Advanced Encryption System (AES) is based on the substitution-permutation network.**

Advanced Encryption System (AES) consists of four rounds:[16]

1. Key Expansion – For each round to add more round, 128-bits of a round key of AES is required. Using the AES Key schedule, round keys are based on the cipher key.
2. Initial Round Keys – Using bitwise XOR, every byte is combined with a combination of the round key.
3. 9,11, or 13 rounds:
 - SubByte
 - ShiftRows
 - MixColumns
 - Add Round Key
4. Final Round
 - SubBytes
 - ShiftRows
 - Add Round Key

Depending on the length of the keys, it decides the total number of rounds that will be applied for each bit, such as 128 bits will use 10 rounds, 192 bits will use 12 rounds, and 256 bits will use 14 rounds.

3.4. Haar Discrete Wavelet Transform (HDWT)

Daubechies proposed the Haar Discrete Wavelet Transform (HDWT) in 1988, due to which the research development was broadly encouraged. To analyze the time series, a majority of the researchers started focusing on time series. The Haar Discrete Wavelet Transform (HDWT) is mainly used for signal processing applications, for watermarking. The advantage that Haar Discrete Wavelet Transform (HDWT) provides is that when a structure of data matches the resolution of the image and can be decomposed at any level. Haar Discrete Wavelet Transform (HDWT) has an excellent space-frequency sub-bands known as LL1 (Approximation Coefficients), LH1 (Vertical Details), HL1 (Horizontal Details), and HH1 (Diagonal Details) mainly used for 2D filter image processing. Haar Discrete Wavelet Transform is a frequency domain technique that provides an absolute secure image that is hard to detect the secret data. **The main advantage of Haar Discrete Wavelet Transform (HDWT) is that when the sensitive data is embedded in the lower part of the wavelet, i.e., the LL subbands it cannot be seen by the human eyes as it is sensitive to the low-frequency part of the decomposition and the image quality remains better.** [17].

The proposed algorithm for hiding the secret image is:

1. First, the disintegration of the cover image is done into the 3 colour planes known as Red (R), Green (G), and Blue (B).
2. Using Haar Discrete Wavelet Transform (HDWT), the cover images' colour plane is decomposed into the four non-overlapping sub-bands. The four sub-bands are LL1 (Approximation Coefficients), LH1 (Vertical Details), HL1 (Horizontal Details), and HH1 (Diagonal Details).
3. The plane is divided using Haar Wavelet filters.
4. Once the disintegration of the secret image is done into four sub-bands known as LL1 (Approximation Coefficients), LH1 (Vertical Details), HL1 (Horizontal Details), and HH1 (Diagonal Details) and the procedure of LL1 (Approximation Coefficients) gets processed in order to obtain the wavelet coefficients.
5. The LL1 (Approximation Coefficients) sub-band embeds the secret message separately into cover images' different bands.

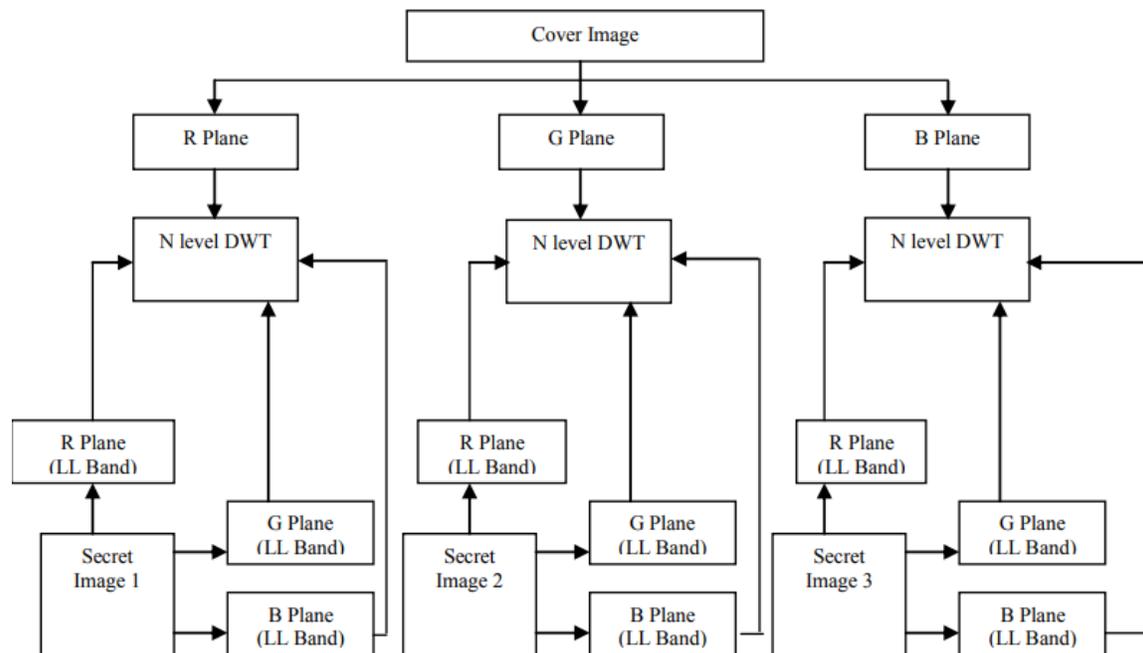


Fig 5. Embedding process of the secret image using Haar Discrete Wavelet Transform (HDWT)

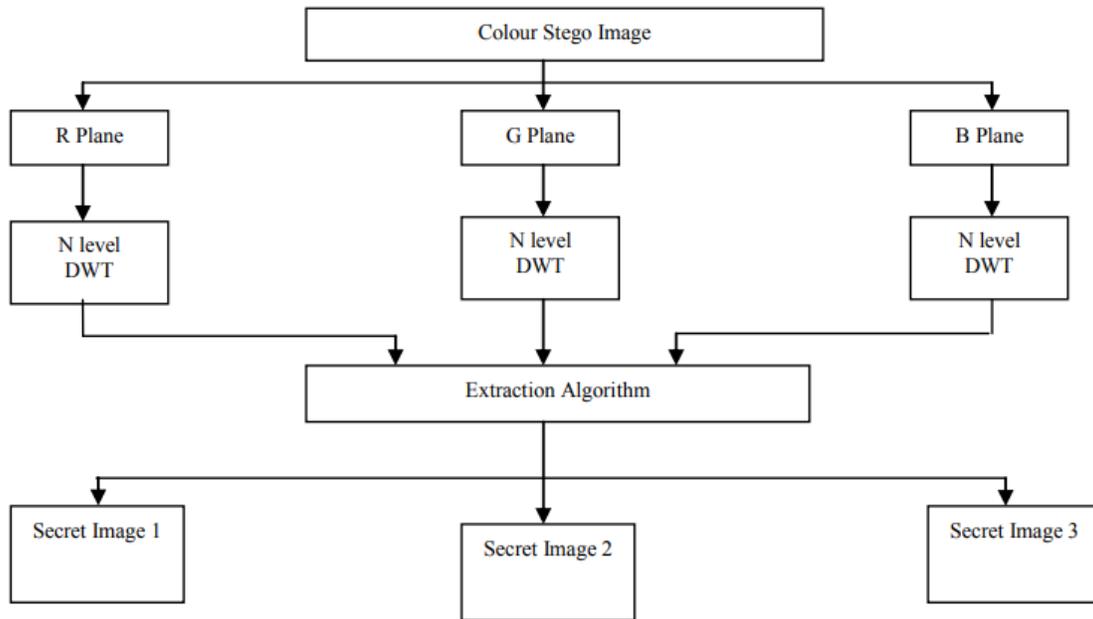


Fig 6. The extraction process of the secret image using Haar Discrete Wavelet Transform (HDWT)

4. Design Specification

The proposed methodology is introduced in order to enhance the steganography techniques, which is implemented in order to safeguard the data embedded in the digital file. To protect the stego image from steganalysis attacks, various algorithms are implemented, such as 256-bit AES Encryption in order to protect the sensitive data from Brute force attack, Embedding Haar Discrete Wavelet Transform (HDWT) for hiding the data in one of the subs-bands which provides an excellent frequency property and in order to achieve a lossless transmission without losing a single bit while transmitting the image. This design specification includes two phases – Embedding Phase and Extraction Phase.

4.1. Embedding Phase

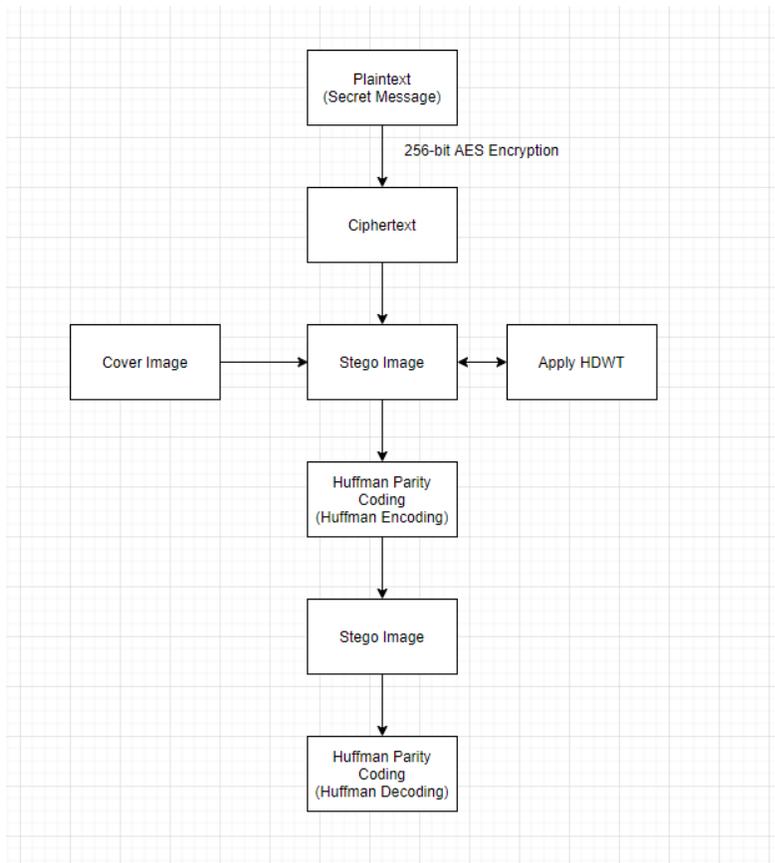


Fig 7. Embedding Process

The prototype of the embedding process that is shown in the below image defines the working module of Image Steganography by maintaining Data Integrity using Huffman Parity Algorithm. A plaintext that is a secret message is chosen in order to convert it into a ciphertext using 256 bits Advanced Encryption System (AES) once the plain text is converted into ciphertext. Then the ciphertext is embedded into the stego image. Once the ciphertext is embedded into the image file, another cover image is inserted into the stego image. Then the Haar Discrete Wavelet

Transformation (HDWT) is applied to the given image to enhance the security of the embedded data in one of the sub-bands, which provides excellent space-frequency after that Huffman Parity Coding is applied on the stego image in order to create a lossless image which will not lose the bits of data consisting inside the data.

4.2. Extraction Phase

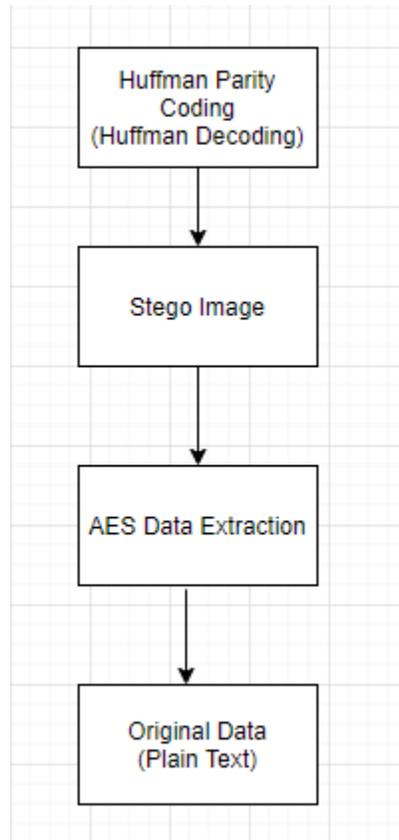


Fig 8. Extraction Process

The extraction process of the prototype works by first decoding the Huffman Parity Coding, where the Entropy and the Efficiency are calculated. Then the same stego image is chosen in order to extract the AES data, and then finally, the AES data is converted into extract the AES data. Then finally, the AES data is converted into the original data, i.e., the plain text.

5. Implementation

The entire prototype is designed and build on the MATLAB R2020a application. The encryption and decryption code of 256-bits Advanced Encryption System (AES) is implemented on MATLAB as well. For implementing Dual Image Steganography, we will use Ubuntu Linux and Python version 2.7 in order to merge two images.

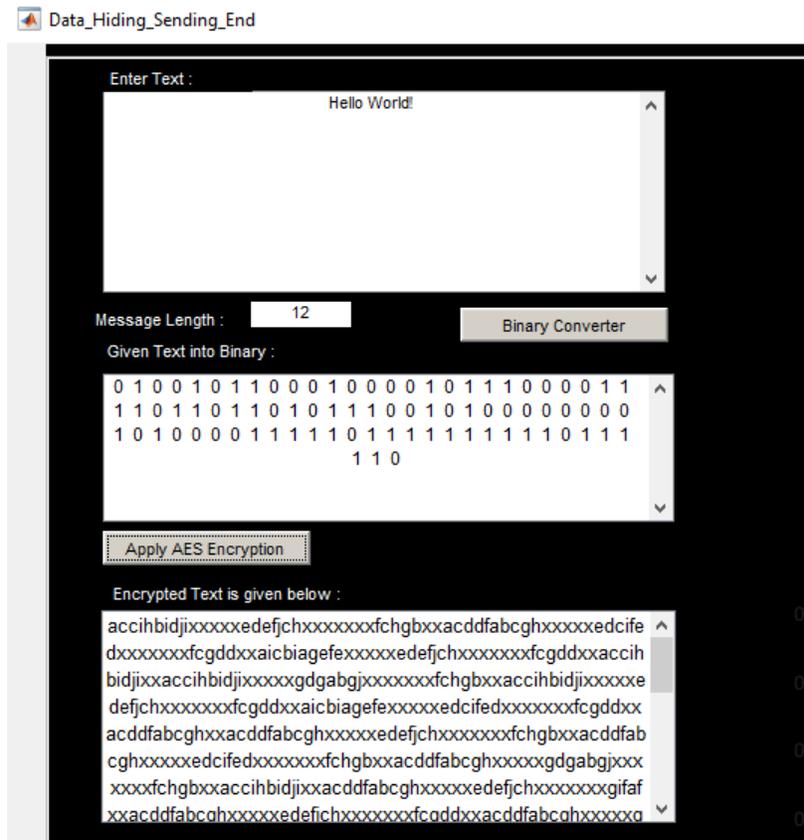


Fig 9. The encryption process of plain text using Advanced Encryption System Encryption



Fig 10. Represents the embedding of the ciphertext in the plain image. Once the ciphertext is embedded in the plain image, it is converted into a stego image. In order to calculate the quality parameters of the stego image here, the Peak Signal to Noise Ratio (PSNR) and Mean-Square Error (MSE) value is calculated.

- **Peak Signal to Noise Ratio (PSNR)**

Peak Signal to Noise Ratio (PSNR) is used to measure the decibels of two images. The ratio comparison is made between two images to check the quality parametric of the two images. If the Peak Signal to Noise Ratio (PSNR) is higher, the better the quality of the recovered image will be. **PSNR value of Lena is 49.9467.**

- **Mean-Square Error (MSE)**

Mean-Square Error (MSE) means the computation and comparison performance byte by byte of the image. If the Mean-Square Error (MSE) represents a lower error rate, it means that the image is quite clean. **MSE value of Lena is 0.421416.**

- **Hiding Capacity**

Hiding Capacity here represents the capacity of an image to hide data inside it. The higher the capacity of an image to hold the embedded data inside it the more secure the data becomes. **Hiding Capacity is 1534 bytes.**

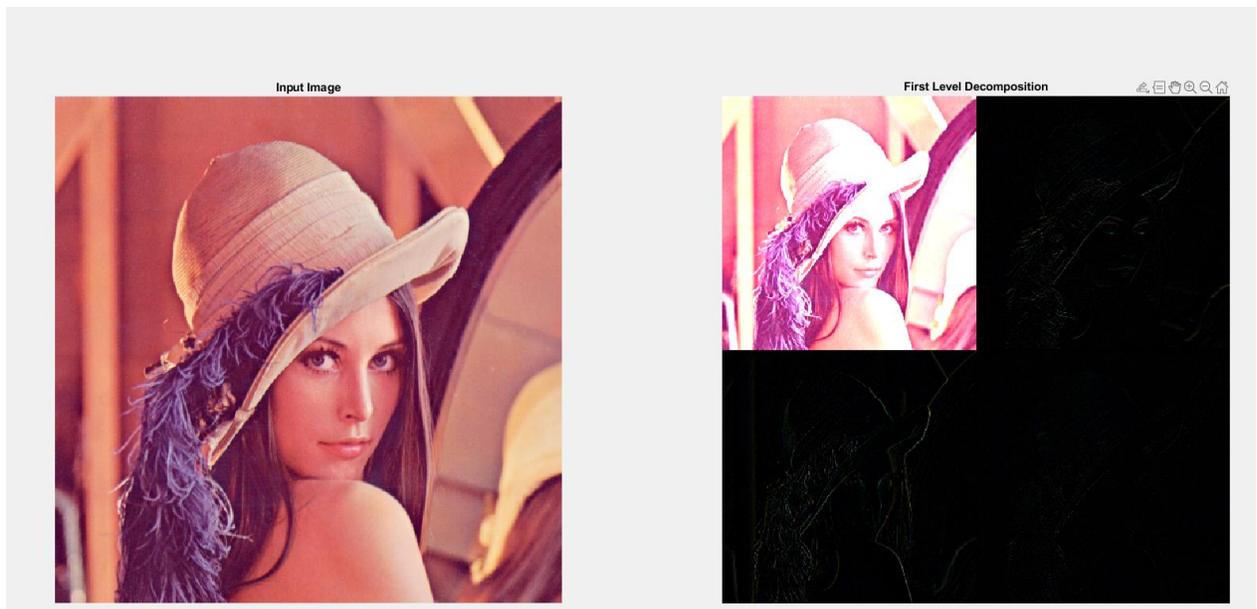


Fig 11. Here the Haar Discrete Wavelet Transform is applied on the stego image in order to provide an excellent space frequency on one of the four sub-bands.

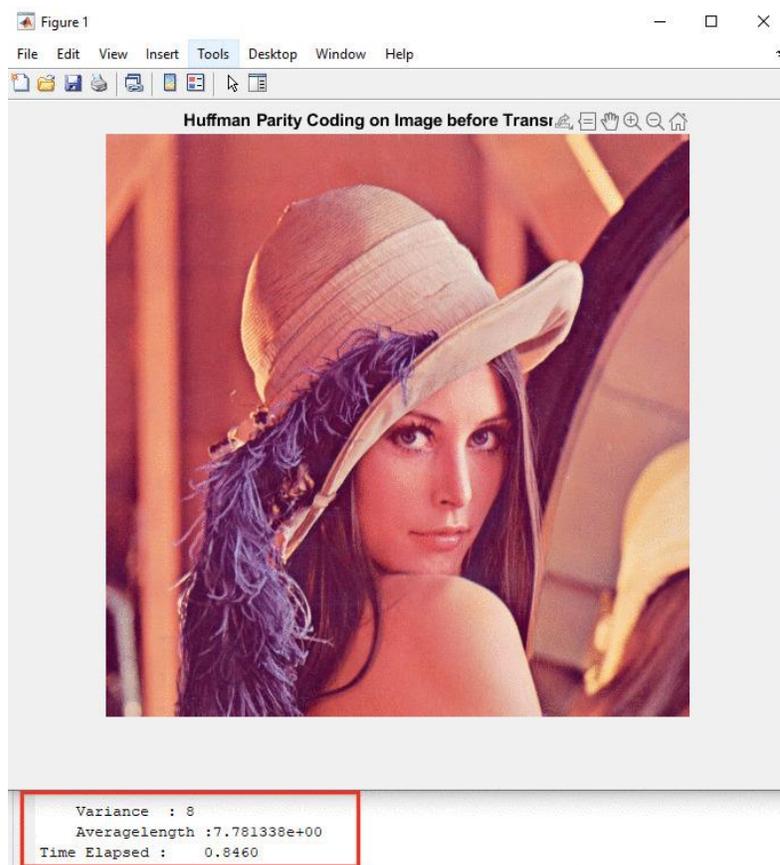


Fig 12. Here the Huffman Parity Coding is applied to the Stego image. Once the Huffman Parity Coding is applied on the Stego Image, the variance and the average length is calculated.

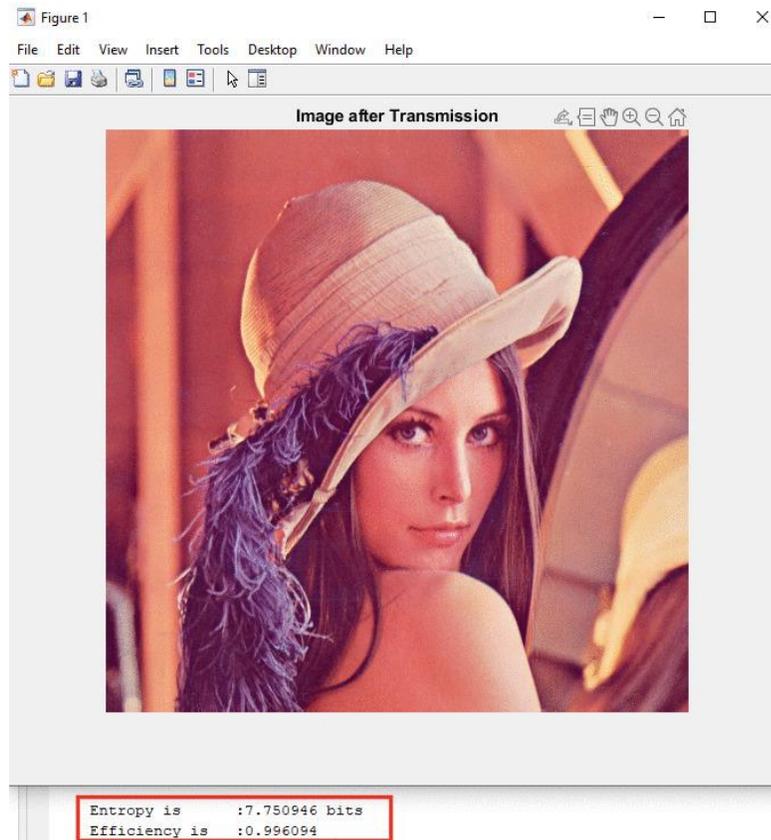


Fig 13. Here once the Huffman Parity Coding is decoded on the stego image, the Entropy Value and Efficiency Value is calculated. The Entropy Value and Efficiency Value here represents how good the compression ratio on the stego image of Huffman Parity Coding (Compression) has worked. Here the Entropy Value and Efficiency Value is transmitted via Noise Channel. The higher the value of Entropy (Compression Ratio) is, the better, the better the shape of the image and better secure and intact the embedded data will be.

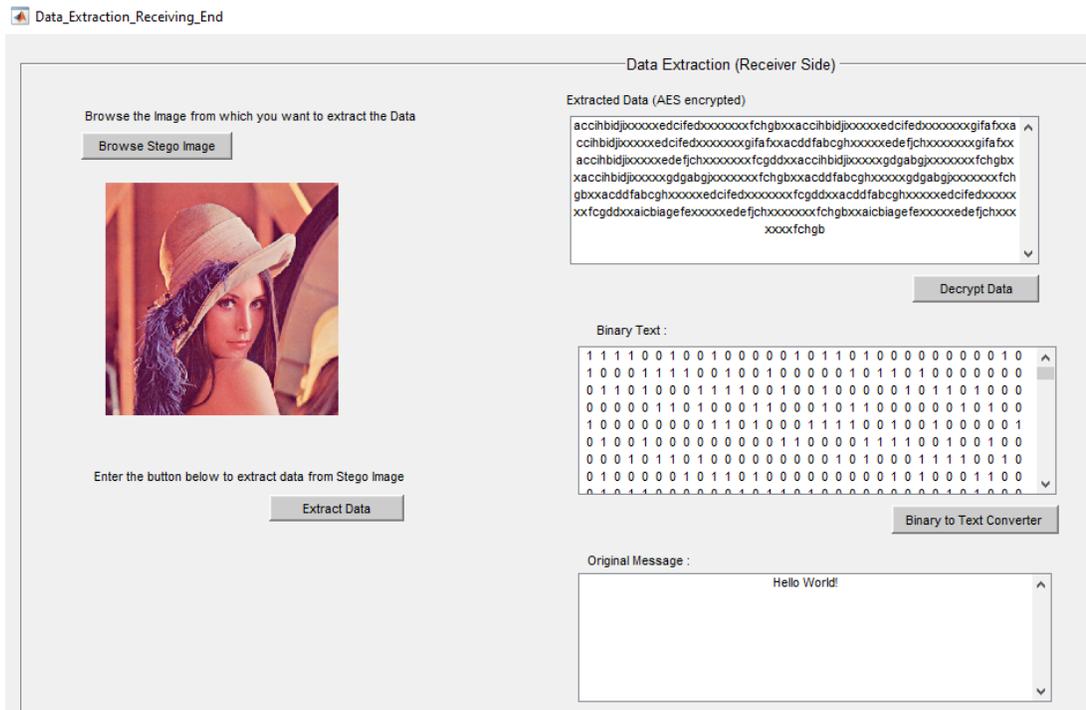


Fig 14. The extraction process here represents the extraction of secret data, which was embedded in the stego image.

```

elliot@ubuntu: ~/Documents/steganography-master
File Edit View Search Terminal Help
elliot@ubuntu:~/Documents/steganography-master$ python steganography.py merge --img1=res/img1.jpg --img2=res/img2.jpg --output=res/output.png

```

```

steganography.py
# Steganography: Hiding an image inside another
## Usage

Create a `virtualenv` and install the requirements:
...
virtualenv venv
source venv/bin/activate
pip install -r requirements.txt

Then, merge and unmerge your files with:
...
python steganography.py merge --img1=res/img1.jpg --img2=res/img2.jpg --output=res/output.png
python steganography.py unmerge --img=res/output.png --output=res/output2.png
...

```

Open ▾



steganography.py

~/Documents/steganography-master

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import click
from PIL import Image

class Steganography(object):

    @staticmethod
    def __int_to_bin(rgb):
        """Convert an integer tuple to a binary (string) tuple.

        :param rgb: An integer tuple (e.g. (220, 110, 96))
        :return: A string tuple (e.g. ("00101010", "11101011", "00010110"))
        """
        r, g, b = rgb
        return ('{0:08b}'.format(r),
                '{0:08b}'.format(g),
                '{0:08b}'.format(b))

    @staticmethod
    def __bin_to_int(rgb):
        """Convert a binary (string) tuple to an integer tuple.

        :param rgb: A string tuple (e.g. ("00101010", "11101011", "00010110"))
        :return: Return an int tuple (e.g. (220, 110, 96))
        """
        r, g, b = rgb
        return (int(r, 2),
                int(g, 2),
                int(b, 2))

    @staticmethod
    def __merge_rgb(rgb1, rgb2):
        """Merge two RGB tuples.

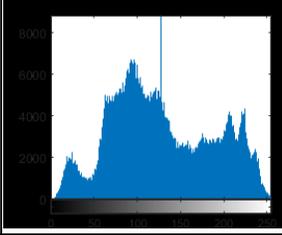
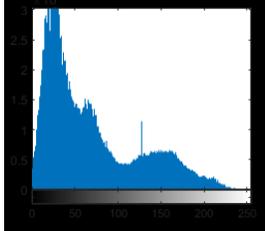
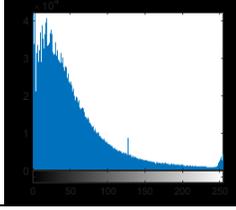
        :param rgb1: A string tuple (e.g. ("00101010", "11101011", "00010110"))
        :param rgb2: Another string tuple
        (e.g. ("00101010", "11101011", "00010110"))
        :return: An integer tuple with the two RGB values merged.
        """
        r1, g1, b1 = rgb1
        r2, g2, b2 = rgb2
        rgb = (r1[:4] + r2[:4],
                g1[:4] + g2[:4],
                b1[:4] + b2[:4])
```

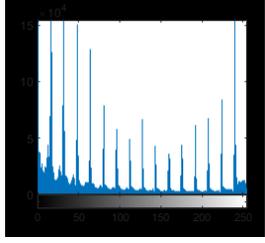


Fig 15. The above code python code was used in order to merge two images (Cover image) (images provided above). We used the “click and image” python packages in order to make the dual image steganography work.

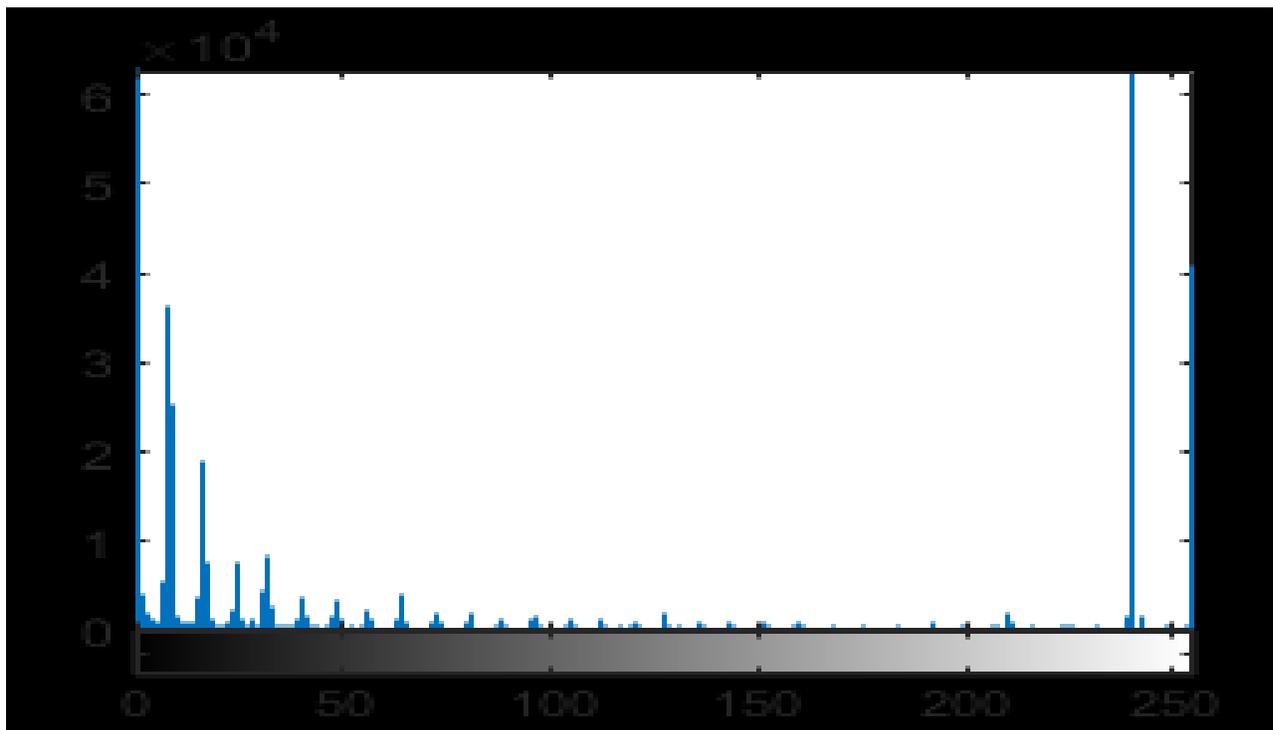
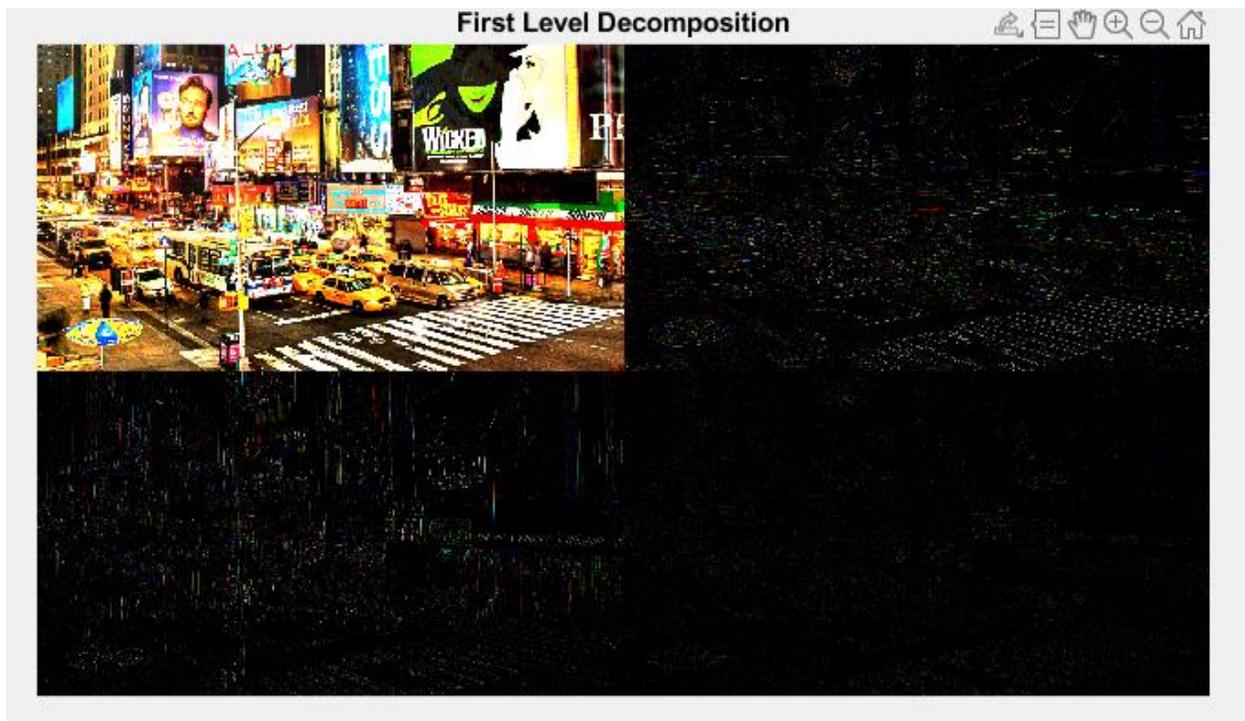
6. Evaluation

A performance analysis was performed on the 3 types of different images, and the PSNR value and MSE value was calculated on each of the following images, and then another performance evaluation was performed on Dual Image Steganography.

No.	Plain Image	Stego Image	Hiding Capacity	PSNR Value	MSE Value
1 - Lena			Hiding Capacity - 5998 	52.1119	0.176849
2- Cat			Hiding Capacity – 3598 	51.9299	0.186628
3- Space			Hiding Capacity – 3838 	52.2404	0.166326

4- City and Skyscrapers (Dual Image)	 		Hiding Capacity- 6000 	52.4092	0.1843306
--------------------------------------	--	---	---	---------	-----------

Evaluation of Haar Discrete Wavelet Transformation on Dual Image Steganography



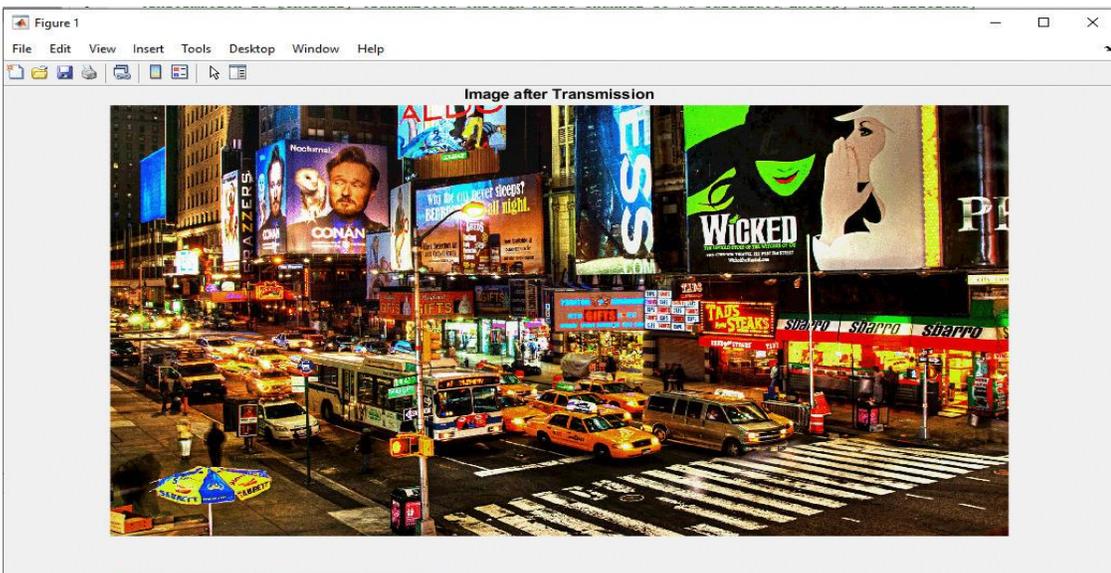
Here, we can see that the decomposition of Dual Image within the four sub-bands of the Haar Discrete Wavelet Transformation and the histogram plots, as discussed in the related work of the Haar Discrete Wavelet Transformation (HDWT). We can see that the stego image which was embedded inside the cover image which contains the secret data of the city image is not visible, and the decomposition of HDWT on each band produces a secure image.

Evaluation of Huffman Parity Coding on Dual Image Steganography

1. Image Before Transmission



```
>> run_this_first
Variance : 8
Averagelength : 6.450415e+00
Time Elapsed : 5.0766
```



```
Entropy is : 6.432479 bits
Efficiency is : 0.995984
```

6.1. Discussion

Considering the above evaluation, we can see a clear comparison between the single image and dual image steganography. The Peak Signal to Noise Ratio, Mean-Square Error, Variance, Entropy, and Efficiency value has been calculated. The Hiding Capacity, PSNR and MSE value of Dual Image Steganography Image against Single Image Steganography have better hiding capacity value, PSNR value, and MSE value. Huffman Parity Coding on Dual Image gave better Variance, Entropy, and Efficiency value. For example, the Hiding Capacity of Single Image Steganography ranged between 3500 to 5998, the Peak Signal to Noise Ratio (PSNR) value ranged from 51-52, and the Mean-Square Error (MSE) value ranged from 0.166 to 0.186 but compared to Dual Image Steganography (City and Skyscrapers (Dual Image)) had a Hiding Capacity value of 6000, PSNR value of 52.4092, and MSE value of 0.1843306 which was better and efficient than Single Image Steganography. The capacity of Huffman Parity on Dual Image Steganography had a better Variance length of 8, Entropy value (Compression Ratio) of 6.432479 bits, and Efficiency value of 0.995984. After the decoding of bits of Dual Image, we found that some bits were lost in the process, but the clarity was much better the secret image (ciphertext) was still secure without any trace of exposure. The efficiency value was good enough to transmit the image over the internet. This proves that the Dual Image Steganography is far better and secure in terms of hiding data. The goal of maintaining the data inside the dual image using Huffman parity coding was achieved. Another method for evaluation is the Structural SIMilarity (SSIM) index is a method for measuring the similarity between two images. The Structural SIMilarity (SSIM) index is a method for measuring the similarity between two images It is an improved version of the universal image quality index proposed before. SSIM is another way of evaluating two images among each other. Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," has made a good comparison of an image which shows a compression of JPEG image how it produces annoying pseudo-contouring effects. [18].

6.2. Limitations

As of now, the Stego Images may not hold a huge amount of secret data, which may lead to disruption of images, which can lead to pixelated images. Due to this reason, attackers may perform a stego analysis attack on such images, which carries a more amount of embedded data. If an image is compressed to its limits, it might be impossible to add extra data within it even if Huffman Parity Coding is applied to it. Dual Image Steganography can be more prone to a Resampling attack, which can lead to loss of bits, but at least the image will stay intact. More improvements like the combination of the Least Significant Bit and Haar Discrete Wavelet Transformation (HDWT) can be implemented on Dual Image Steganography in order to enhance the security of the image and can be less prone to attacks. We have observed a combination of different types of files such as ".png," ".jpg," ".bmp," ".jpeg" have experimented for Dual Image Steganography, but we faced many complications where the images got distorted.

7. Conclusion and Future Work

In this paper, we have successfully implemented and demonstrated the working of Securing Confidential Data using Dual Image Steganography by maintaining Data Integrity using the Huffman Parity Algorithm. The entire demonstration was implemented in MATLAB and Ubuntu Linux, and Python 2.7 was used. We have successfully shown that how the data integrity was achieved through Huffman Parity Coding by showing the Entropy and the Efficiency value of the Dual Image and how confidential data was secured using Haar Discrete Wavelet Transformation (HDWT) because of the hiding capacity that resulted better than single image steganography.

In future work, a combination of different embedding algorithms can be implemented, such as the combination of Least Significant Bit (LSB) and Haar Discrete Wavelet Transformation (HDWT) on Dual Image Steganography. Dual Image Steganography can be tried with other steganographic techniques such as Audio and Video Steganography in order to achieve a more secure version of steganography.

References

- [1] A. C. Gautam, "Secure End to End transmission using Audio Steganography and AES encryption," p. 19.
- [2] Tanmay Bhattacharya, Nilanjan Dey, and S. R. Bhadra Chaudhuri "DWT based Dual Steganographic Technique" *International Journal of Computer Applications* (0975 – 8887) Volume 38– No.5, January 2012, Reference IEEE.
- [3] A. K. Singh and J. Singh, "Steganography in Images Using LSB Technique," *Int. J. Latest Trends Eng. Technol.*, vol. 5, no. 1, p. 5, 2015.
- [4] "(PDF) COMPARISON OF LSB AND DWT STEGANOGRAPHY TECHNIQUES." https://www.researchgate.net/publication/328901349_COMPARISON_OF_LSB_AND_DWT_STEGANOGRAPHY_TECHNIQUES (accessed Aug. 08, 2020).
- [5] K. B. Raja, C. R. Chowdary, K. R. Venugopal, and L. M. Patnaik, "A Secure Image Steganography using LSB, DCT and Compression Techniques on Raw Images," in *2005 3rd International Conference on Intelligent Sensing and Information Processing, Bangalore, India, 2005*, pp. 170–176, doi: 10.1109/ICISIP.2005.1619431.
- [6] E. H. Houssein, M. A. S. Ali, and A. E. Hassanien, "An image steganography algorithm using Haar Discrete Wavelet Transform with Advanced Encryption System," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS), Sep. 2016*, pp. 641–644.
- [7] A. Alharbi and M.-T. Kechadi, "A Steganography Technique for Images Based on Wavelet Transform," in *Future Data and Security Engineering*, vol. 10646, T. K. Dang, R. Wagner, J. Küng, N. Thoai, M. Takizawa, and E. J. Neuhold, Eds. Cham: Springer International Publishing, 2017, pp. 273–281.
- [8] T. Bhattacharya, N. Dey, and S. Bhadra Chaudhuri, "A Session based Multiple Image Hiding Technique using DWT and DCT," *Int. J. Comput. Appl.*, vol. 38, Aug. 2012, doi: 10.5120/4684-6808.
- [9] H. A. Prajapati, D. N. G. Chitaliya, and A. Professor, *Secured and Robust Dual Image Steganography: A Survey*.
- [10] K. Thakre and N. Chitaliya, "Dual Image Steganography for Communicating High Security Information," vol. 4, no. 3, p. 6, 2014.
- [11] S. Gupta, G. Gujral, and N. Aggarwal, "Enhanced Least Significant Bit algorithm For Image Steganography," vol. 15, no. 4, p. 3, 2012.

[12] A. Ali, A.-H. Seddik, and N. Hussien, "A Combined Approach of Steganography and Cryptography Technique based on Parity Checker and Huffman Encoding," *Int. J. Comput. Appl.*, vol. 148, pp. 26–32, Aug. 2016, doi: 10.5120/ijca2016911031.

[13] A. Westfeld and A. Pfitzmann, "Attacks on Steganographic Systems," in *Information Hiding*, vol. 1768, A. Pfitzmann, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 61–76.

[14] P. Sallee, "Model-Based Steganography," in *Digital Watermarking*, vol. 2939, T. Kalker, I. Cox, and Y. M. Ro, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 154–167.

[15] D. P. Mahajan and A. Sachdeva, "A Study of Encryption Algorithms AES, DES and RSA for Security," p. 9, 2013.

[16] "Advanced Encryption Standard," *Wikipedia*. Aug. 12, 2020, Accessed: Aug. 13, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Advanced_Encryption_Standard&oldid=972590376.

[17] C.-C. Lai and C.-C. Tsai, "Digital Image Watermarking Using Discrete Wavelet Transform and Singular Value Decomposition," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 11, pp. 3060–3063, Nov. 2010, doi: 10.1109/TIM.2010.2066770.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004. <https://www.cns.nyu.edu/~lcv/ssim/> (Accessed Sep. 27, 2020).

All the source code was found from GitHub, and I made some changes accordingly as per my proposed prototype. The links are given below.

- <https://github.com/bishwasmandal246/Encryption-and-Image-Steganography-using-DWT>
- <https://github.com/rajyash1904/Image-Compression-and-Transmission>
- <https://github.com/kelvins/steganography>