INFORMATION THEORY AND CODING

ECE4007

Research Paper On

"Compression using Huffman Coding for LSB Stegnography"

Submitted by

MAYANK RANJAN-17BEC0314
HRITHIK SARDA-17BEC0078
NIKHIL BHARAT RAI-17BEC0449

Submitted to

Dr.RAMYA S.
Assistant Professor Sr.Grade 1
School of Electronics Engineering
VIT

Abstract

Out of various types of Stegnography, we have chosen the most common method i.e. LSB Stegnography. Its implementation is simple but it has several limitations. One limitation is that the size of the cover image is much larger than the file concealed within it. In order to overcome this limitation, we attempt to compress the concealed file in the form of an image using Huffman coding method so as to reduce its size and hence reduce the size required to hide that image.

Keywords— LSB Stegnography, Huffman Coding, Image Compression, Data Compression

Introduction

The word steganography is obtained from the Greek words steganos (meaning hidden or covered) and the Greek root graph (meaning to write). Steganography is the technique of hiding secret data within a standard , non-secret, file or message so as to avoid detection; the key data is then extracted at its destination. the utilization of steganography are often combined with encryption as an additional step for hiding or protecting data. In simple words, it's hiding information into other information. Steganography doesn’t alter the structure of the key message, but hides it inside a cover-object. After masking, cover object and stego-object are similar. thanks to invisibility or hidden factor it's difficult to recover information without known procedure in steganography. Detecting procedure of Steganography is understood as Steganalysis. Image steganography may be a method of hiding information during a cover-image that generates a stego-image. This stego-image is then sent to the receiver by any medium, where the third party doesn't know that this stego image has hidden message. After receiving stego-image, hidden message are often extracted with or party doesn't know that this stego image has hidden message. After receiving stego-image, hidden message are often extracted with or without stego-key which is employed in embedding algorithm. during this paper, we offer an approach during which the key message is compressed using Huffman method and inserting it in to the LSB pixels of image and retrieving the message using same technique which the key message is compressed using Huffman method and inserting it in to the LSB pixels of image and retrieving the message using same technique.

Methodology

When we talk about image steganography, the idea is quite simple i.e. images comprises of pixels whose value usually indicates the color of that particular pixel. In the case of a grayscale image, the range of pixel values is 0-255, the lower value corresponds to the colour 'black' whereas the upper one corresponds to the colour 'white'.

[3] LSB stands for Least Significant bit. The idea behind LSB embedding is that if we change the last bit value of a pixel, there won't be much visible change in the color. For example, 0 is black. Changing the value to 1 won't make much of a difference since it is still black, just a lighter shade.

The required secret message is compressed using Huffman coding and then inserted in the image which is undergoing LSB Steganography.

Literature review

Year	Title	Methodology	Features
2019	An improved security and message capacity using AES and Huffman coding on image steganography[1]	A combination of AES, Huffman Coding, and Haar DWT which aims to reduce the total of message's bit in steganography. There are two main processes, which are: embedding the message process and extracting the message process. Implementing Huffman Coding in an encrypted message image and producing two different results which are a compressed encrypted message image and Huffman tree file. From a compressed encrypted message image that already got, embed into the chosen subbands (LH, HL, and HH) and produce an embedding subbands of the cover image. The final step is composing embedding subbands of the cover image by using Inverse DWT and producing a stego image. A stego image will be processed by using DWT to	In this research, a combination of AES–Huffman Coding–DWT to secure a message image and conceal into a cover image, produced a good stego image quality. Provided a higher capacity in DWT for steganography by reducing the total of message's bit up to 22.319% from the original message's bit. A good stego image quality is proven by achieving

		produce four subbands of stego image.Extract a message from the chosen subbands (LH, HL, and HH) and produce a compressed encrypted message image.Next step is by using the Huffman tree file and implementing Huffman Coding, decompress a compressed encrypted message image and producing an encrypted message image.Finally, decrypting an encrypted message image by using AES with a key and producing a message image.	the average of PSNR result is more than 40db which is 46.1788.
2019	A New Chaotic Image Steganography Technique Based on Huffman Compression of Turkish Texts and Fractal Encryption with PostQuantum Security[2]	Steganography scheme composes of four distinct phases namely natural language statistic extraction and obtaining Huffman encoded output of Turkish texts; encrypting encoded output with new Fractal encryption algorithm; determining the morphologically higher entropy regions of the selected cover image; chaotic location selection for LSB steganography. The reverse operation is used for extracting the secret message.	With the proposed method it is possible to obtain a BPP value as high as 2.93.
2019	A Comparative Study on Improvement of Image Compression Method using Hybrid DCT – DWT[3] Techniques with Huffman Encoding for Wireless Sensor Network Application[3]	It is observed that compressed image have same conclusion where the PSNR and SSIM values are high for BMP and PNG formats compared to JPG for both DCT and DWT compression. Based on the analysis done, the BMP image format can compress better than the PNG and JPG format by about 25.12% and 90.15%, respectively. This might due to the characteristics of the image format whereby the BMP is an uncompressed image (original image) which would have compression as maximum as possible until it may achieve the same result as JPG format. Subsequently, PNG is a lossless image compression (no data loss), which	This paper proposes a hybrid compression of DCT, DWT and Huffman technique for image transmission over ZigBee network. The input image is compared with three different types of image format, which are BMP, JPG and PNG. The DCT and DWT compression technique are applied

		able to compress image while allowing good image quality and followed by JPG is a lossy image compression where losses occur after compression, hence reduce a small amount of image's quality. Therefore, the higher compression ratio, the higher compression can be realized. Although PSNR is a common performance that can be analyzed in image processing, however SSIM also is a good technique to compare the image quality of an original image with the compressed image. In this result, DCT compression has good PSNR and SSIM values, while DWT compression has succeed to achieve good results in all performances tested. Thus, applying a hybrid method of both DCT and DWT are compatible to each other in supporting and providing a better performance of image compression.	for each format and the results show that the size of the compressed image is able to reduce by about 96.92% and 99.8% from its original image, thus a good quality of image have been developed. Moreover, the results obtained show better PSNR value
2018	A New Steganography Technique using JPEG Images[4]	The proposed technique consists three phases; Random Number Generator phase, Generating the Shuffling Array phase, Embedding or Extracting Secret message in cover image phase	inserted a secret message within a JPEG image without affecting the image quality and compression ratio. They utilized almost 40% to 50% out of the total number of blocks to hide the secret message.
2017	Image Steganography Based on AES Algorithm with Huffman Coding for Compressionon Grey Images[5]	Cover image and Secret message undergo AES Encryption algorithm, followed by Steganography with secret message. Then Compression is done using Huffman coding; followed by decompression and Decryption by the same respective methods to get the cover and Secret message.	The original and the reconstructed gray scale image were same but the reconstructed image consisted of 32Kb if secret data.

2017	Hybrid approach for improving data security and size reduction in image steganography.[6]	Proposed work is partition into two categories. First one is for hiding the data and second one is for extracting the hidden data from stego image. DATA EMBEDDING: a) Cover image is selected and text is entered. b) Huffman encoding technique is applied for compression so that size of data will be reduced and large amount of data will be hidden. c) DNA is applied for encrypting the data so that security level will be increased. d) State Transition is mainly used for hiding the data. e) Finally, stego image is created DATA EXTRACTION: a) At receiver"s side, stego image is selected. b) Data extraction is performed on the selected stego image. c) Decryption is done by the DNA. d) Huffman algorithm performs the decompression. e) After decompression the original text is extracted.	Steganography provides better security for data sharing. . The proposed technique will perform the similar task of steganography with two different algorithms such as DNA and Huffman. Whereas DNA algorithm is used for hiding and decryption of the text and Huffman is used for compression and decompression of the entered data. By compression and encryption of the data its security level is increased and it is not easy to recognize.
2016	Image Encryption Using Huffman Coding for Steganography, Elliptic Curve Cryptography and DWT for Compression[7]	To implement steganography Huffman coding technique is used. Encryption is implemented by using ECC (Elliptic Curve Cryptography) algorithm. To reduce the storage space used to store digital images, DWT (Discrete Wavelet Transform) technique is used along with EZW compression method.Novelty: Huffman coding is used on message symbols before it is send for encryptionthe Elliptic Curve Cryptosystem is by far the most secured. The main attraction of Elliptic Curve Cryptography is that it provides the same level of security as Diffie-Hellman or RSA but with much shorter keys. ECC can get a level of	This paper performs steganography, encryption and compression all together on the image data. For steganography It is using Huffman Coding, for encryption purpose It is using Elliptic Curve Cryptography and for compression Ii is using Discrete Wavelet Transform. After applying all

		security with a 164-bit key that other systems require a 1,024-bit key to achieve. It provides equal security with smaller key size DWT: a key advantage it has over is temporal resolution: it captures both frequency and location information (location in time)..It provides multiresolution system. Superior to fourier transform and applications range in image compression,radar and earthquake prediction.Cuts up data into different frequencies and study each component with a resolution matched to its scale.It has more basis functions than fourier transform. The embedded zerotree wavelet algorithm (EZW) is a simple, yet remarkably effective, image compression algorithm, having the property that the bits in the bit stream are generated in order of importance, yielding a fully embedded code. EZW is a lossy image compression algorithm	these techniques on image data it results in an encryption method which is highly secure. The values of PSNR, MSE and Compression Ratio for encryption of images are highly acceptable . For the implementation of this proposed work we are using Matlab software. The compression ratios are 6.4,12.5 and 42.3 which are very high.
2016	Image Steganography using LSB, LSB+Huffman Code, and LSB+Arithmetic Code[8]	Cover image and Secret message undergo AES Encryption algorithm, followed by Steganography with secret message. Then Compression is done using Huffman coding; followed by decompression and Decryption by the same respective methods to get the cover and Secret message.	The original and the reconstructed gray scale image were same but the reconstructed image consisted of 32Kb if secret data.

We can elaborate some of the above tabular content as :

The paper [2] is consists of four different phases for steganography which are: language statistic recognition and extraction of Turkish scripts and encode them using Huffman coding; encryption of the encoded output with new Fractal encryption algorithm; determining the morphologically higher entropy regions of the selected cover image; chaotic location selection for LSB steganography. The reverse operation is used for extracting the secret message. With the proposed method, there is a possibility to obtain a BPP(bits per pixel) value as high as 2.93.

The paper [4] has proposed a new steganography technique that consists of three phases sing JPEG image. They utilized almost 40-50% of the total number of blocks to hide the secret message without affecting the image quality and compression ratio.

The paper [5] uses AES algorithm along with Huffman coding for Image steganography. Both cover image as well as the secret data undergo AES encryption followed by steganography with secret message after which they are compressed using Huffman coding followed by decompression and Decryption by the same respective methods to get the cover and Secret message. They achieved in successfully reconstructing the image with 32kb of secret data while the original and final(gray scale) image were identical.

In the paper [8], application of three techniques for steganography is visualized and their results are compared. The first technique includes hiding a binary secret message into LSB of image pixels. The second one compresses a binary secret message using Huffman Code before hiding it into the Least Significant Bit of the image pixels whereas the third technique compresses a binary secret message using Arithmetic Coding before hiding.

The comparison was done using peak-signal to noise ratio by which we can infer, Huffman coding and arithmetic coding provided almost similar results which is significantly greater than the results obtained from LSB alone.

Algorithm

The encoding is done using the following steps:

- Convert the image to greyscale
- Resize the image if needed
- Convert the message to its binary format
- Initialize output image same as input image
- Traverse through each pixel of the image and do the following:
- Convert the pixel value to binary
- Get the next bit of the message to be embedded after Huffman coding
- Create a variable temp
- If the message bit and the LSB of the pixel are same, set temp = 0
- If the message bit and the LSB of the pixel are different, set temp = 1
- This setting of temp can be done by taking XOR of message bit and the LSB of the pixel
- Update the pixel of output image to input image pixel value + temp
- Keep updating the output image till all the bits in the message are embedded
- Finally, write the input as well as the output image to local system

Code

```
clear all;
close all
clc;
%ENCRYPTING
inpu = imread('mario.png'); %Image used to store our secret message
inpu=rgb2gray(inpu) ;%Convert it into Black and White
message= input('Enter: '); %Enter the secret message to be encrypted
s=unique(message);% To remove repeated characters
l=length(s);
%To calculate the probability of occurence of each character
for i=1:l
   count=0;
   for j=1:(length(message))
      if s(i)==message(j)
         count=count+1;
      end
      p(i)=count/(length(message));%Probability of occurence of ith character
   end
end
symbols=[];
for i=1:l
   symbols(i)=s(i);
end
dict = huffmandict(symbols,p);%creating huffman dictionary
comp = huffmanenco(message,dict);%Encoding the characters using the dictionary created
bin_message = comp(:);
N = length(bin_message);
bin_num_message=bin_message;
output = inpu;
%LSB Steganography
height = size(inpu, 1);
width = size(inpu, 2);
embed_counter = 1;
%Traversing through each pixel of the image
for i = 1 : height % Number of rows in the matrix of the image
   for j = 1 : width % Number of coloums in the matrix of the image
      if(embed_counter <= N)
         LSB = mod(double(inpu(i, j)), 2);%Getting the least significant bit of a given pixel
```

```
        temp = double(xor(LSB, bin_num_message(embed_counter)));%Change the bit if and
only if it is not same as that of encoded character
        output(i, j) = inpu(i, j)-temp; %Updating the pixel after modiying its least significant
bit.
        embed_counter = embed_counter+1; %Next encoded bit
      end
    end
end
imwrite(inpu, 'F:\itc\originalImage.jpg');
imwrite(output, 'F:\itc\stegoImage.jpg');
peak= psnr(output,inpu); %To calculate the peak signal to noise ratio of the new encrypted
image with respect to the original image
%DECRYPTING
in=output;
height = size(in, 1);
width = size(in, 2);
embed_counter = 1;
ii=1;
%Getting the least significant bit of every pixel of the  new encrypted
%image
for i = 1 : height
   for j = 1 : width
      if(embed_counter <= N)
         LSB(ii) = mod(double(in(i, j)), 2);
         ii=ii+1;
         embed_counter = embed_counter+1;
      end
    end
end
dcr=huffmandeco(LSB,dict); %Decoding the secret message using the Huffman dictionary
created during encrypting
msg=char(dcr); %Retrieving the secret message.
```

RESULTS

Fig: Original Coloured Image

Fig: Coloured Image converted to Grey Image

Fig: Output Image with Secret message

Using Huffman Coding we were able to achieve message compression upto 57%.

Comparison Table of Results

Year of Publication	Title of the paper	Name of the Journal	Their Results	Our results
2019	An improved security and message capacity using AES and Huffman coding on image steganography [1]	TELKOMNIKA Indonesian Journal of Electrical Engineering.	Provided a higher capacity in DWT for steganography by reducing the total of message's bit up to **22.319%** from the original message's bit. **PSNR result is more than 40db which is 46.1788.**	Peak Signal to Noise ratio of **LSB+HUFFMAN is 77.5316 db** and that of **PURE LSB is 73.797 db.**
2019	A New Chaotic Image Steganography Technique Based on Huffman Compression of Turkish Texts and Fractal Encryption with	IEEE Access	With the proposed method it is possible to obtain a **BPP value as high as 2.93**.	**BPP** in our case **is 1.**

	PostQuantum Security [2]			
2019	A Comparative Study on Improvement of Image Compression Method using Hybrid DCT - DWT Techniques with Huffman Encoding for Wireless Sensor Network Application [3]	International Journal of Integrated Engineering.	The size of the compressed image is able to **reduce by about 96.92% and 99.8%** from its original image	We used **compression in message and not in image** and were able to achieve **57% compression**.
2018	A New Steganography Technique using JPEG Images [4]	International Journal of Advanced Computer Science and Applications.	They utilized almost **40% to 50%** out of the total number of blocks to hide the secret message.	With Huffman encoding, the maximum information size that can be embedded into a cover image using LSB Steganography can be increased from **20-30%**
2017	Image Steganography Based on AES Algorithm with Huffman Coding for Compressionon Grey Images [5]	International Journal of Innovative Technology and Exploring Engineering(TM)	The original and the reconstructed gray scale image were same but the reconstructed image consisted of **32Kb of secret data.**	This depends on the base image that we take to hide secret text. In our selected image we were able to hide **20kb** of data but we never checked the maximum limit though.

2017	Hybrid approach for improving data security and size reduction in image steganography. [6]	Journal of University of Babylon	The PSNR values for secret images ranged from **52 to 60 db**	Peak Signal to Noise ratio of **LSB+HUFFMAN is 77.5316 db** and that of **PURE LSB is 73.797 db.**
2016	Image Encryption Using Huffman Coding for Steganography, Elliptic Curve Cryptography and DWT for Compression **[7]**	International Conference on Trends in Electronics and Informatics	The compression ratios are 6.4,12.5 and 42.3 which are very high.	The secret message was compressed to **57%** of its size.
2016	Image Steganography using LSB, LSB+Huffman Code, and LSB+Arithmetic Code [8]	International Journal of Computer Applications,	PSNR: (LSB+ARITH) > (LSB+HUFF) > LSB. **Arithmetic coding provides the highest PSNR ratio** out of the three	PSNR: (LSB+HUFFMAN) > Pure LSB i.e. **LSB along with Huffman compression yields a better PSNR ratio**.

CONCLUSION

Based on the test result, it can be inferred that Huffman coding is more effective and gives a bigger compression ratio if there are less color value variations within the image. With Huffman encoding, the maximum information size that can be embedded into a cover image using LSB Steganography can be increased from 20-30%. For further efforts, the author hopes to improve his huffman coding algorithm source code as his code at the time of this paper being released doesn't perform very well at compressing bitmap images, but performs really well at compressing text files. Another works in the future includes researching the actual possible maximum information after doing Huffman compression and comparing the result with the theoretical results calculated in this paper. Peak Signal to Noise ratio of LSB+HUFFMAN is 77.5316 and that of PURE LSM is 73.797 [This is calculated for the message 'Information Theory and Coding'].

From the comparison table, we infer that all our results do not exceed the boundaries of other paper's output results or ranges by a great difference. But, we consider our work as of significant value as in some cases our results were better than that of our reference papers whereas in few cases our outputs got in range or slightly below than that of the other papers.

References

[1] Sari, Atika & Ardiansyah, Giovani & Rachmawanto, Eko & Setiadi, De Rosal Ignatius Moses. (2019). An improved security and message capacity using AES and Huffman coding on image steganography. TELKOMNIKA Indonesian Journal of Electrical Engineering. 17. 2400-2409. 10.12928/TELKOMNIKA.v17i5.9570.

[2] M. C. Kasapbaşi, "A New Chaotic Image Steganography Technique Based on Huffman Compression of Turkish Texts and Fractal Encryption With Post-Quantum Security," in IEEE Access, vol. 7, pp. 148495-148510, 2019, doi: 10.1109/ACCESS.2019.2946807.

[3] Hussin, Mumtaz & Poad, Farhana & Joret, Ariffuddin. (2019). A Comparative Study on Improvement of Image Compression Method using Hybrid DCT-DWT Techniques with Huffman Encoding for Wireless Sensor Network Application. International Journal of Integrated Engineering. 11. 10.30880/ijie.2019.11.03.016.

[4] Watheq, Rand & Almasalha, Fadi & Qutqut, Mahmoud. (2018). A New Steganography Technique using JPEG Images. International Journal of Advanced Computer Science and Applications. 9. 10.14569/IJACSA.2018.0911107.

[5] Kennedy, J., Khan, T., Ahmed, J., & Rasool, M. (2017). Image Steganography Based on AES Algorithm with Huffman Coding for Compressionon Grey Images.

[6] Abod, Zaid. (2018). Hybrid Approach To Steganography System Based On Quantum Encryption And Chaos Algorithms. Journal of University of Babylon. 26. 10.29196/jub.v26i2.499.

[7] Singh, Archana & Singh, Vinay & Yadav, Shashank. (2019). Image Encryption Technique Using Huffman Coding and Spatial Transformation. 352-356. 10.1109/ICOEI.2019.8862652.

[8] Al-mazaydeh, W.I., & Sheshadri, H.S. (2016). Image Steganography using LSB, LSB+Huffman Code, and LSB+Arithmetic Code. International Journal of Computer Applications, 155, 1-7.