Secured transmission of an Image Using Run-Legth Encoding Algorithm

Sathwika Bavikadi, Gajjela Venkata Sarath Kumar, Shashidhar Sanda
Department of Signal Processing
Blekinge Institute of Technology
Karlskrona, Sweden
{saba16,vega16,shsa16}@student.bth.se

Wlodek J. Kulesza
School of Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
wka@bth.se

Abstract— **The Increased popularity of multimedia application in image processing places a great demand on efficient data storage and transmission techniques. Mainly security troubles a lot during transmission of images and videos over internet communication. Hence Data hiding is now becoming one of the most important research areas. Our study focuses on Run-length encoding and Huffman encoding techniques. By utilizing the run-length information and an additional bit value with repeating bits in to a cover image to compress the information meanwhile it's also used to encrypt at high capacity. Comparison results show that RLE technique is better than Huffman encoding because it possesses near distortion quality of PSNR, Compression Ratio is high in case of RLE than that of Huffman.**

Keywords— Compression Ratio, Huffman encoding, Peak Signal to Noise Ratio, Run-length encoding.

I. Introduction

In the modern world technology while transmitting an image through internet there are various threats like compromising confidentiality, integrity, availability and accountability to the data of an image. Security is of utmost concern. With increase in cybercrime technology, providing secured transmission is very essential

Security is provided to images like highly confidential Defence forces images, blue prints of a company projects, secret medical images. Secured transmission of data is ensured by data hiding of the image such that the existence of data is difficult to detect by unauthorized or intruders. The main aim of data hiding of the Image is to increase the Hiding capacity and Imperceptibility. The proposed algorithm, Run-length encoding (RLE) is used to enhance the embedding capacity of the image.

In order to provide secured transmission of the data of the image efficiently we compare parameters, Peak-signal to noise ratio (PSNR), Compression ratio and Mean square error (MSE) in RLE and Huffman encoding algorithms. It can be applied to a grayscale or colored image.

II. Survey of Related Works

Research in image data hiding became very popular now-a-days. Various authors proposed various techniques, Venkata Keerthy S et al. [1] proposed an algorithm to improve the embedding capacity of the image and decrease the unauthorized usage of the data and thereby providing secured transmission of the data.

Smitha et al. [2] proposed a data hiding method which is implemented for image encryption. The Least significant bits (LSBs) of the encrypted image are compressed using a data hiding key to create space to embed an additional data and providing higher degree of security.

III. Problem Statement and Main Contribution

As one can see from the review of related works, in order to provide security, the only way would be not letting the hackers know about the presence of important information in the transaction of the image.

To overcome this problem we use a compression technique i.e., RLE [3] and Huffman encoding ensuring security for data of an image. Since the data is transmitted through a network it is always in compressed domain. This information is used in for embedding the data in compressed domain where the compressed data coefficients are manipulated to embed data. Thus providing secured transmission of image.

The main contribution of this research is to provide security of data of an image using RLE [4] and tradeoff between RLE and Huffman encoding techniques is calculated using the parameters PSNR and Compression ratio and MSE. And Implementation is done using MATLAB.

IV. Problem Solution

Initially the compression of an image is done and security of that image is provided using RLE. RLE is preferable compared to Huffman technique because it possess near distortion quality of PSNR.

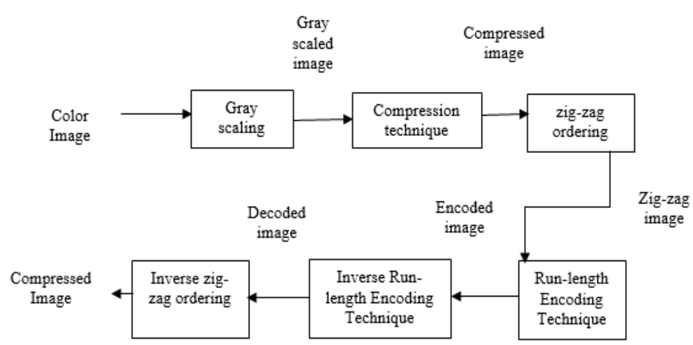

Fig. 1. Block-diagram

A. Modeling

Initially the color image is selected and it is converted into gray scale image. That image is compressed using Discrete wavelet transform (DWT) and then the zigzag scan traverse through the DWT coefficients in the increasing order of their spatial frequency. We provide security for that compressed image by applying zigzag ordering and then apply RLE for zigzag image, the encoded image is obtained. In the extraction process we apply Inverse RLE for encoded image then decoded image is obtained. To get that compressed image perfectly we apply inverse zigzag ordering so finally the compressed image is obtained.

1. Compression Technique

The compression technique used for compressing the input image signal is DWT. In DWT, an image signal can be analyzed by passing it through an analysis filter bank followed by a decimation operation. This analysis filter bank, which consists of a low pass and a high pass filter bank at each decomposition stage, is commonly used in image compression.

A two- dimensional transform can be accomplished by performing two separate one-dimensional transforms. First, the image is filtered along the x-dimension using low pass and high pass analysis filter and decimated by two. Low pass filtered coefficients are stored on left part of the matrix and high pass filtered on the right. Because of decimation, the total size of transformed image is same as the original image. Then, it is followed by filtering the sub-image along the y-dimension and decimated by two. Finally, we have split the image into four bands denoted by LL, HL, LH and HH after one level decomposition. This process of filtering the image is called pyramidal decomposition of image.

2. Zigzag Ordering Technique

Zigzag ordering technique is used to reorder the coefficients of the image sub bands. This process orders the coefficients from low frequency to high frequency.

3. Run Length Encoding

Run-length encoding is a data compression algorithm that is supported by most bitmap file formats, such as TIFF, BMP. RLE is suited for compressing any type of data regardless of its information content, but the content of the data will affect the compression ratio achieved by RLE. Initially the compression of an image is done and security of that image is provided using RLE.

B. Implementation and Application

The security of an image is provided by implementing zigzag ordering and further Run-length encoding is implemented to encode the data of an image. After the encoding, Inverse RLE is applied to the encoded image and inverse zigzag is also implemented to get the same compressed image.

This algorithm is very easy to implement. RLE compression is only efficient with files that contain lots of repetitive data. These can be text files if they contain lots of spaces for indenting but line-art images that contain large white or black areas are far more suitable. Computer-generated color images can also give fair compression ratios. The RLE technique is applied for the transmission of images for security. Implementation is done using MATLAB.

C. Validation and Verification

Mean square error (MSE), Peak signal to noise ratio (PSNR) and Compression ratio (CR) values are calculated for Run-length encoding and Huffman encoding using MATLAB. Using the equation (1) and equation (2)

TABLE I. COMPARISION TABLE OF RLE AND HUFFMAN ENCODING

Image	CR (no units)		MSE (no units)		PSNR (db)	
	RLE	***Huffman***	***RLE***	***Huffman***	***RLE***	***Huffman***
Lena	2.1	1.07	0.21	0.85	54.8	48.8
Pepper	2.49	1.31	0.20	0.8	55.0	49.1
Butterfly	2.6	1.16	0.21	0.81	54.7	49.0
Nature	3.8	1.5	0.16	0.67	55.8	49.5
Ship	1.8	1.04	0.2	0.68	54.9	49.8

The TABLE I. Represents that Mean square error is less in case of RLE and Peak signal to noise ratio is high when compared to Huffman encoding and Compression ratio of RLE is higher than Huffman encoding.

As shown in Fig.2. If PSNR is high, then the image quality is high and also the data of an image is protected.

As shown in Fig.3. and Fig.4. If Mean square error is less then the image quality is good. The compression ratio is high for RLE than for Huffman encoding so RLE is better not only for security but also for compression tenchnique

V. CONCLUSION

By the comparisons of both techniques RLE and Huffman encoding we can conclude RLE is better technique than Huffman encoding technique by using the parameters Mean square error, Peak signal to noise ratio and compression ratio. RLE possess near distortion quality of Peak signal to noise ratio. Compression ratio is also less for Huffman encoding. Hence RLE is better than Huffman encoding technique to protect the data of an image. And can be applied to any application such as Defense forces, Detective agencies, Companies, Banking and Medical images

Further additional security can be provided to data of videos. One can combine two compressed domain data hiding and image stitching algorithm to provide better security for an image.

REFERENCES

[1] Venkata Keerthy S, Rhishi Kishore T K C, Karthikeyan B, Vaithiyanathan V and Anishin Raj M M, "A hybrid technique for quadrant based data hiding using Huffman coding," Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015 International Conference on, Coimbatore, 2015, pp. 1-6.

[2] M. Smitha, V. E. Jayanthi, and A. Merlin, "Image encryption using separable reversible data hiding scheme," in *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, 2013, pp. 1–6.

[3] R. S. Lin and S. W. Hu, "A Modified Run-length Image Data Hiding for High Embedding Capacity," in *Fifth International Conference on Information Assurance and Security, 2009. IAS '09*, 2009, vol. 1, pp. 673–676.

[4] C.-C. Chang, C.-Y. Lin, and Y.-Z. Wang, "New image steganographic methods using run-length approach," *Inf. Sci.*, vol. 176, no. 22, pp. 3393–3408, Nov. 2006.

VI. APPENDIX

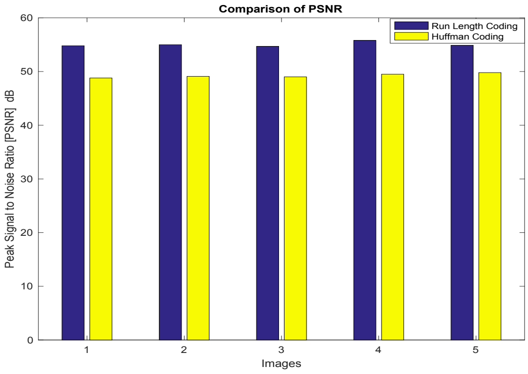

Fig.2. Comparison of PSNR between RLE and Huffman

$$PSNR = 10 * \log_e 10 \left[\frac{(255^2)}{MSE} \right] \qquad (1)$$

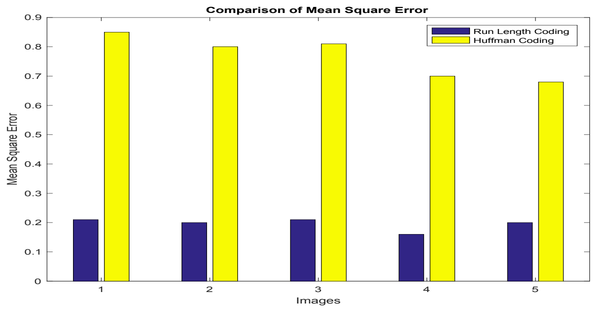

Fig.3. Comparison of Mean square error between RLE and Huffman

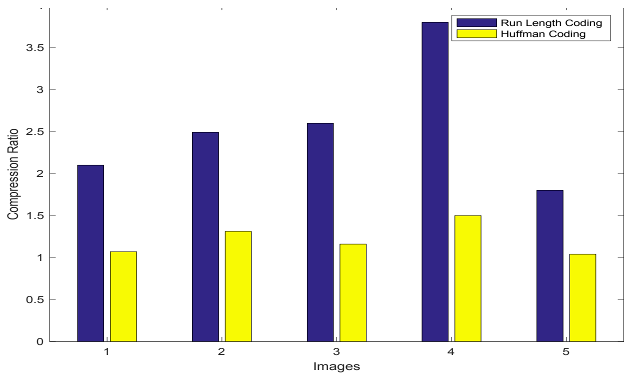

Fig.4. Comparison of Compression ratios between RLE and Huffman

$$Compression\ ratio = \frac{Original\ image}{Compressed\ image\ file\ size} \qquad (2)$$