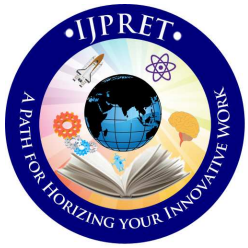
Research Article *ISSN: 2319-507X*
Nilam Shaha, IJPRET, 2013; Volume 1(8): 119-126 *IJPRET*



INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

IMAGE COMPRESSION USING HUFFMAN CODING

NILAM P. SHAHA[1], PROF. M. S. KUMBHAR [2]

1. PG Electronics student, Electronics & Telecommunication Department, RIT Sakharale Shivaji University, Kolhapur
2. Professor, Electronics & Telecommunication Department, RIT Sakharale.

Accepted Date:

27/02/2013

Publish Date:

01/04/2013

Keywords

Lossless compression; Compression ratio (CR); Peak signal to noise ratio (PSNR)

Corresponding Author

Ms. Nilam P. Shaha

Abstract

The need for an efficient technique for compression of images increasing everywhere. Image compression is widely used to reduce the data size of digital images. Images are very important documents nowadays; to work with them in some applications they need to be compressed, more or less depending on the purpose of the application. There are various algorithms that perform this compression in different ways; some are lossless which keep the same information as the original image, while some others loose information when compressing the image. In this paper we have been working with Huffman lossless compression algorithm for different grayscale images. We have seen how well the algorithm works for each of the images. The compression ratio (CR) and peak signal to noise ratio (PSNR) are obtained for different images and compared to find the influence of algorithm on images.

INTRODUCTION

Data compression has become requirement for most applications in different areas such as computer science, Information technology, communications, medicine etc. In computer science, Data compression is defined as the science or the art of representing information in a compact form [1]. Image compression is the application of data compression on digital images. In effect, the objective is to reduce redundancy of the image data in order to be able to store or transmit data in an efficient form [2].

Digital image compression techniques can be divided into two classes: lossless and lossy compression. Currently these two basic classes of data compression are applied in different areas. In lossless compression, every single pixel that was originally in the image remains same after the image is uncompressed. All of the information is completely stored. Lossy compression reduces the size of the image by permanently eliminating some redundant information. Reconstructed image contains degradation to the original image.

The main objective of this paper is to compress images by reducing number of bits per pixel required to represent it; to decrease the transmission time of images and reconstructing the image by decoding Huffman algorithm.

COMPRESSION PRINCIPLE

A common characteristic of most of the images is that the neighbouring pixels are correlated and therefore contain redundant information. The foremost task then is to find less correlated representation of the image. Two fundamental components of compression are redundancy and irrelevancy reduction.

a. Redundancy reduction aims at removing duplication forms of the signal (Image/Text).
b. Irrelevancy reduction omits parts of the signal that will not be noticed by signal receiver, namely the Human visual system.

In an image there are three types of redundancies in order to compress file size. They are:

a. Coding redundancy: Fewer bits to represent frequently occurring symbols.

b. Interpixel redundancy: Neighbouring pixels have almost same value.
c. Psycho visual redundancy: Human visual system cannot simultaneously distinguish all colours.

A. Compression advantages of variable length coding over fixed length coding

Coder assigns a code word, a binary bit stream to each symbol at the output of encoder. The coder may employ Fixed length coding (FLC) or Variable length coding (VLC).

Fixed length coding assign equal length code bits to every symbol irrespective to the number of occurrences. This method gives a constant compression to the data set. Additionally the decoding time for symbol is also uniform. This results in the slow coding system under large data processing.

Whereas variable length coding will assigns variable length code word to the symbols of set. In variable length coding compression takes place based on using shorter code words for symbols that occur more frequently and longer code word to less occurring symbols. This results in higher compression of data set with the reduction in decoding time when compared to fixed length coding.

The commonly used Fixed length coding techniques are run-length coding, gray coding etc. Huffman coding, Arithmetic coding are few examples of variable length coding. The most commonly used variable length coding is Huffman coding which is lossless.

B. Huffman coding

Huffman coding is classical data compression technique invented by David Huffman. It is optimal prefix code generated from set of probabilities and has been used in various compression applications. These codes are of variable code length using integral number of bits. This idea causes a reduction in the average code length and thus overall size of compressed data is smaller than the original. Huffman's algorithm provided the first solution to the problem of constructing minimum redundancy codes. The compression process is based on building a binary tree that holds all symbols in the source at its

leaf nodes and their corresponding probabilities at the side.

IMPLEMENTATION OF HUFF-MAN COMPRESSION ALGORITHM

Huffman code procedure is based on the two observations:

a. More frequently occurred symbols will have shorter code words than less frequently.
b. The two symbols that occur least frequently will have the same length.

The Huffman code is designed by merging the lowest probable symbols and this process is repeated until only two probabilities of two compound symbols are left. Thus a code tree is generated and Huffman codes are obtained from labelling of the code tree.

A. Working of Huffman algorithm

Following steps describes working for Huffman coding.

Step 1: Read the image on the workspace of Matlab.

Step 2: Call a function which will find the symbols (i.e. pixel value which is non-repeated).

Step 3: Call a function which will compute the probability of each symbol.

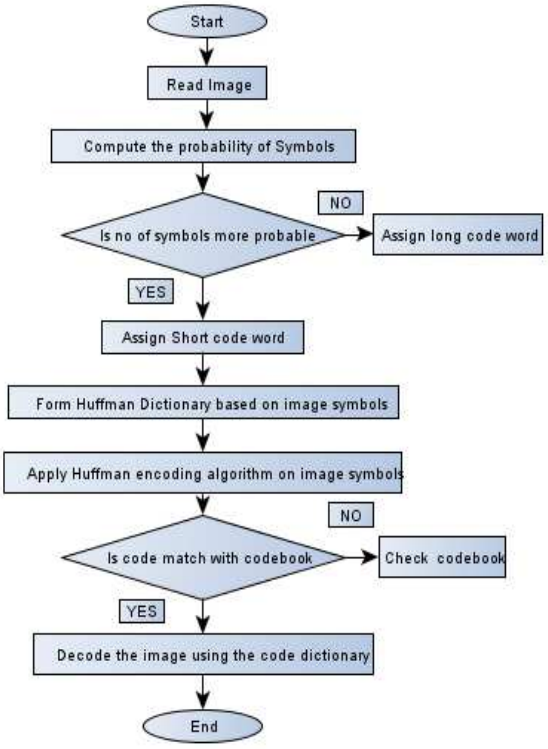
Step 4: Probability of symbols are arranged in decreasing order and lower probabilities are merged. This step is continued until only two probabilities are left and codes are assigned according to rule that, highest probable symbol will have a shorter length code.

Step 5: Further Huffman encoding is performed i.e. mapping of code words to the corresponding symbols will result in compressed data

Step 6: The original image is reconstructed i.e. decompression is done using Huffman decoding.

Step 7: Match the code words with code in dictionary to get the reconstructed image.

The flow of this code is as below



B. Performance Parameter

1. Compression Ratio (CR):-

Compression ratio (CR) is defined as ratio of number of bits required to represent the size of original image to the number of bitsrequired to represent the size of compressed image. Compression ratio shows that the number of times the image has been compressed.

$$CR = \frac{Uncompressed}{Compressed} \qquad (1)$$

To determine the distortion in the image with reference to original image, some quality measurement matrices can be applied. The following are the most commonly used measures:

2. Peak Signal to Noise Ratio (PSNR):-

The PSNR is most commonly used as a measure of quality of reconstruction of image. A higher PSNR would normally indicate that the reconstruction is of higher quality.

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

3. Mean Square Error (MSE):-

The MSE is the cumulative squared error between the compressed and the original image.

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left(f(x,y) - f^{1}(x,y)\right)^2 \qquad (3)$$

A lower value of MSE means lesser error. As seen from equation (2) there is inverse relation between MSE and PSNR.

4. Entropy (H):-

The entropy of a symbol is defined as the negative logarithm of its probability. To determine the information content of a message in bits we express the entropy.

$$H = -\sum_{i=1}^{N} (p_i . \log_2 p_i) \qquad (4)$$

Here p_i is the occurrence probability of symbols$_i$.

In compression, entropy determines how many bits of information are actually present in a message.

IV. SIMULATION RESULT

We have developed Huffman algorithm for image and analysed performance parameters PSNR, MSE, CR and Entropy.

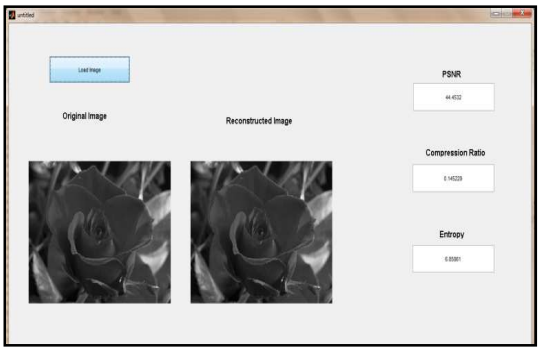
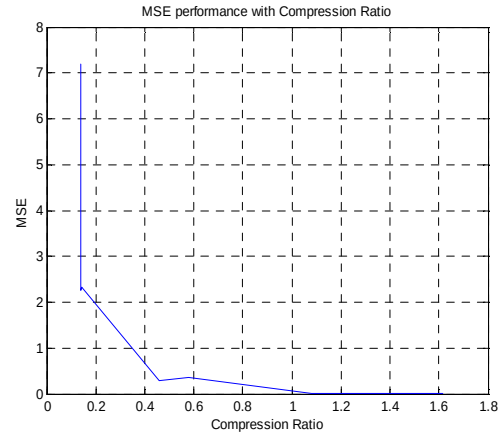


Fig.1 GUI: Huffman algorithm

Simulation results show the lossless image compression scheme. After executing algorithm on Rose1 image, it shows that the image is successfully reconstructed using Huffman coding method and the reconstructed image is exactly same as original image. It is shown in Fig.1. The PSNR and CR is 44.45dB, 0.145 respectively as shown in the Table I.

Table IPerformance measure parameter for different images

Image	CR	MSE	PSNR	Entropy
cameraman	0.140	2.582	44.011	7.084
Lena	0.138	2.248	44.611	7.200
Gaussian	0.172	0.324	53.023	5.770
vegetables	0.138	7.199	39.558	7.165
Pepper	0.143	0.770	49.261	6.995
flower	0.135	2.881	43.534	7.378
Bird	0.128	3.980	42.132	7.743
Lady	0.134	2.233	44.641	7.403
Girl	0.129	6.052	40.311	7.716
Rose	0.133	4.975	41.163	7.506
Bars	0.489	0.203	55.053	1.953
Rose1	0.145	2.332	44.453	6.858
Chex	0.577	0.355	52.631	5.373
Box	1.615	0.015	66.521	1.750
White box	1.082	0.011	67.832	0.328
Screen	1.132	0.061	60.258	0.399
Ball	0.457	0.282	53.619	6.543



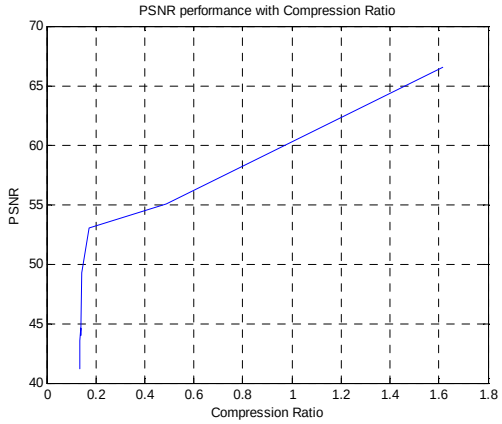


Fig.2 Compression Ratio v/s MSE

Fig.3 Relation between PSNR and CR

CONCLUSION

The proposed Huffman algorithm was applied on different images. This algorithm makes decision of what to compress based on the highest existing intensities in the image. The number of intensities selected depends on the bit rate of the images. Therefore, images with different bits per pixel can be compressed with proposed algorithm. The results show that when we increase the compression ratio then we get higher PSNR and minimum error. It means the compressed image is almost equal to the original image. As we find compression scheme is having a lower MSE and a high PSNR, we recognize that proposed algorithm is better one.

ACKNOWLEDGMENT

I would like to thank my supervisor for his knowledgably guidance throughout this work. I also like to thank all the members of department of electronics and telecommuni-cation, RIT, Rajaramnagar for their kind cooperation and help. Finally my gratitude goes to my family and friends for their constant support.

REFERENCES

1. I. M. Pu, Fundamental Data compression. Butterworth- Heinemann, 2006.

2. Sachin Dhawan, “A Review of Image Compression and Comparison of its Algori-thms”, Dept. of ECE, UIET, KurukshetraUniversity, Haryana, India. IJCET VOL.2, Issue 1, March 2011.

3. Md. RubaiyatHasan,” Data Compression using Huffman based LZW Encoding Tech-

nique", International Journal of Scientific & Engineering Research, Volume2 Issue11, November-2011,1-7.

4. Mamta Sharma, "Compression Using Huffman Coding", IJCSNS International Journal of Computer Science and Network Security,VOL.10 No.5, May 2010,133-141.

5. Structures for Data compression.http://www.gitta.info-Versionfrom:18.6.2010 [6] Dr.E.Kannan and G. Murugan, "Lossless Image Compression Algorithm for Transmitting over Low Bandwidth Line" International Journal of Advanced Research in Computer Science and Software Engi-neering Volume 2, issue 2, February 2012.

6. X. Kavousianos, E. Kalligeros and Nikolos, "Optimal Selective Huffman Coding for Test-Data Compression," IEEE Trans. Computers, vol.56, no.8, pp.1146-1152,Aug. 2007.

7. Slobodan Vucetic., "A Fast Algorithm for Lossless Compression of Data Tables by Reordering". Data Compression Conference (DCC'06). IEEE, 2006, pp. 469-469.