ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

Lossless Image Compression Using Huffman Coding Algorithm

Jyoti A. Patil, Dipali M. Lohar, Prachi M. Badgujar, Dipmala V. Patil

B. E Student's, Department of Computer Engineering, Shram Sadhana Bombay Trust's College of Engineering and Technology, Bhambhori, Jalgaon, India

ABSTRACT: Nowadays, the volume of the image is increasing with time, transmission of large amount of image in less time, less bandwidth with better quality. For many years, people debated about problem in storage and transfer. Many people think is only one solution to overcome problem is lossy image compression. People have failed to notice, lossy compression degrades quality of image it compressed. Rethinking approach to lossless image compression using Huffman Coding algorithm to fix quality degradation problem. Replacing lossy compression technique gives better compression ratio without degradation of quality. For solving problem, Huffman coding algorithm approach is use lossless image compression. Using lossless Huffman coding compression algorithm image is compressed up to 40.08%. Image compression plays vital role in saving memory and time while transmission Huffman algorithm is comparatively better in overall compression technique.

KEYWORDS: Image Compression, Huffman Coding, lossless, lossy.

I. INTRODUCTION

In recent days, use of image is now rapidly increasing. Simply, one of the person having multiple images and he is trying to store more and more image within minimum storage space. If anyone want to send image on internet he try to send that images in minimum transmission bandwidth and fast also. So option is to compress image. Image contains redundant information because correlated pixel having the same color and information. Field of Image compression is rapidly growing and it is active research area. The aim of image compression is to remove noisy and repeated pixel to compress image. If image is compress then the storage space and transmission bandwidth problem is solve for transferring the large amount of image data without degrading the quality of image.

II. LITERATURE SURVEY

In 2010, Jau-Ji Shen et al presents vector quantization based image compression technique. Adjust the encoding of the difference map in original image and after it restored in VQ compressed version. It experimental results show although scheme needs to provide extra data, it can substantially improve quality of VQ compressed images and further adjusted depending on difference map from lossy compression to lossless compression.

In 2011, Suresh Yerva, presents approach of lossless image compression using novel concept of image folding. Proposed method uses property of adjacent neighbor redundancy for prediction. In lossless method, column folding followed by row folding is applied iteratively on image till image size reduces to a smaller predefined value. Proposed method is compared with existing standard lossless image compression algorithms and results show comparative performance. Data folding technique is a simple approach for compression provides good compression efficiency and lower computational complexity compared to the standard SPIHT technique for lossless compression

III. IMAGE COMPRESSION

Image is combination of pixel. Pixel is containing the information about that point which he occupies. In that it contains RGB colours also. Pixels are redundant which are correlated to each other some space. Redundancy is

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

duplication of data pixel in image. Image compression is representation of image which is similar to original image but it requires minimum space as compare to original image space.

- Types of image compression.

1. Lossy Image Compression: Lossy compression degrades the quality of image when it is compressed.
2. Lossless Image Compression: Lossless compression doesn't degrade the quality of image when it is compressed.

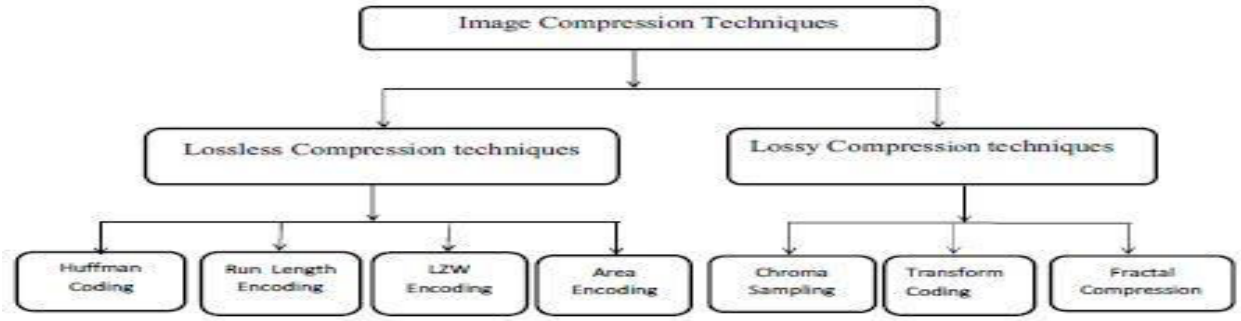

Figure 1: Classification of Image Compression

In Figure 1 all classification is shown of image compression technique. In that first lossless compression techniques contains the 4 techniques. From this Huffman Coding is better for compressing image as compare to other techniques. So we use this technique for developing project. In that we use Java language and also use NetBean for easy management of code. Huffman coding is easy to implement as compare to others. It has standard algorithm for data compression also. On that concept we prefer to use Huffman coding Algorithm for image compression.

IV. HUFFMAN CODING ALGORITHM

It is very popular technique for compression of data. The inventor of Huffman coding is David A. Huffman (1925-1999). On the name of inventor this techniques called as Huffman Coding Algorithm. It compresses the data into small form or in binary form to use it by simple way. Huffman is an MIT student belongs to Electrical Engineering and PHD also. He invents this technique in 1950 and inception of this technique in 1952. This technique gives short length code for compressing the data as compare to other techniques is prove after the long discussion of Gilbert and Moore 59. Algebraic approach which is used for constructing Huffman code is by Karf 61. Huffman code is construction of minimum redundancy code to compress image.

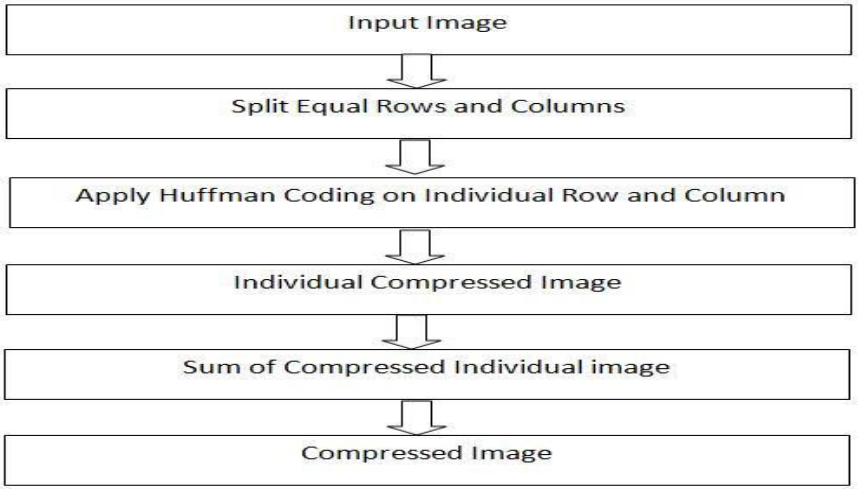

Figure 2: Block Diagram of Huffman Coding Compression

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

In the Figure 2, stepwise execution of Huffman coding is shown. For compression of image, give input as an image, split image into equal rows and column. Next apply Huffman coding on individual rows and column. Individual rows and column get compressed then compressed the whole image. Output is produced as compressed image.

- **ALGORITHM**

The algorithm which is use for developing to compress image is in the following order:
Step 1: Using image box control read image which want to be compressed.
Step 2: Call function which will sort the pixel in Rows and Column. Function prioritizes the pixel based on frequency count of each pixel in image.
Step 3: Call function which will create initial heap. Reheap tree according to occurrence of each node in tree, lower occurrence is attached in heap. Create new node where left child is lowest in sorted list and right is the second lowest in sorted list.
Step 4: Build Huffman tree based on prioritized list. Chop-off those two elements in sorted list as they are now part of one node and add probabilities. The result is the probability for the new node.
Step 5: Perform insertion sort on list with new node.
Step 6: Repeat STEPS 3, 4, 5 UNTIL you only have1 node left.
Step 7: Perform a traver saloftree to generate code table. This will determine code for each element of tree in the following way.

Code for each symbol obtained by tracing path of symbol from root of the tree. A 1 is assigned to a branch of one direction and 0 is assigned for a branch to other direction. For example, symbol which is reached to branch at twice, then left once represented by pattern '110'. The figure depicts codes for nodes of sample tree.

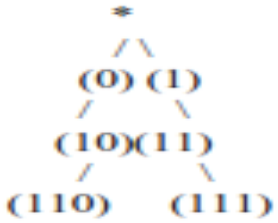

Figure 3: Huffman Tree

Step 8: First Huffman tree is built, Huffman codes, which require minimum information to rebuild, may be generated by steps:
Step 1: Length of soft codes resulting from a Huffman tree generated as shown in Figure 3.
Step 2: Sort the symbols to be encoded by the lengths of their codes (use symbol value to break ties).
Step 3: Initialize current code as all zeros and assign code values to symbols from longest to shortest code as follows:
A. If current code length is greater than length of code for current symbol, right shift off extra bits. Assign code to current symbol.
B. Increment the code value.
C. Get the symbol with the next longest code.
D. Repeat from until all symbols are assigned codes.

Step 9: Encoding pixel of Huffman code has been generated, data is encoded simply by replacing symbol with its code.
Step 10: Original image is reconstructed i.e. decompression by using Huffman Decoding.
Step 11: Generate Huffman tree equivalent to encoding tree. Huffman code for some encoded data, decoding can accomplished by reading encoded data one bit at a time.
Step 12: Read input character wise and left to tree until last element is reached in tree.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

Step 13: Output the character encodes in the leaf and returns to the root, and continues the step 12 until all the codes of corresponding symbols is known.

V. IMPLEMENTATION

For developing the Huffman Coding for Image Compression we use the Java programming language and NetBean. NetBean for easy code management. It simply manages files for java development.

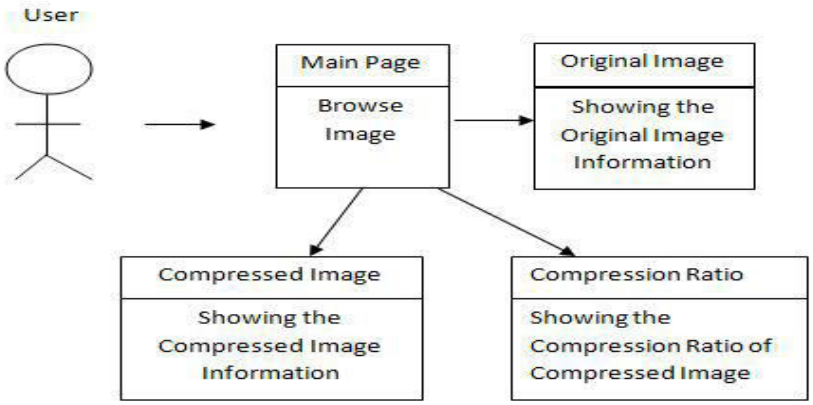

Figure 4: Flow of System

The Figure 4 describes flow of system. Firstly, user visit main page. It select browse button for choosing image from desktop to main page. It show image path on main page. User select original image detail button for showing original image information, show original image. User come to main page select compression image details button it show compression image details compressed image. User come to main page select compression ratio button it show original image size, compressed image size and compression ratio.

VI. RESULT

Image compression requires quality in form of frequency. Frequency explicitly set from developer for particular image types is required for image compression.In proposed system, it takes compression quality in frequency which is inversely proportional to image compression ratio. Quality is required for less compression ratio and vice versa. Quality of images as per frequency is varying.

In our system we take JPG and PNG are 2 image types. In result firstly it takes JPG image in 0.52f frequency it gives 40.08 percentage compression ratio and it is feasible for project which is shown in following Table A.

Image Name	Original Image Size (KB)	Compressed Image Size (KB)	Frequency (f)	Compression Ratio (Percentage)
sun	365.62	208.62	0.64	42.94
sun	365.62	219.34	0.6781	40.01
tree	200.62	115.35	0.99	42.50
tree	200.62	119.35	0.9916	40.51
Logo	12.41	7.44	0.2687	40.04

Table A: For JPG Images

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

In result, second take PNG image with 0.6781f frequency and it gives 40.1 percentage compression ratios. It is feasible for project which is shown in Table B.

Image Name	Original Image Size (KB)	Compressed Image Size (KB)	Frequency (f)	Compression Ratio (Percentage)
sun	365.62	208.62	0.64	42.94
sun	365.62	219.34	0.6781	40.01
tree	200.62	115.35	0.99	42.50
tree	200.62	119.35	0.9916	40.51
Logo	12.41	7.44	0.2687	40.04

Table B: For PNG Images

It is not compressed size of image at high level, images are not properly visible to user it give the 40 percentage ratio.

VII. ANALYSIS

In the Analysis section firstly take JPG image. User gives compression quality in frequency form with frequent manner and analyse compression ratio. Table C shows multiple frequency use for compression ratio.

Attribute	Existing System	Proposed System
Compression Ratio (percent)	40.00	40.08

Table C: For JPG Images

In Table C comparison of existing system compression ratio and proposed system compression ratio is analyse. Analysis from Table C is existing system gives 40 percentage compression ratio and proposed system gives 40.08 percentage compression ratio. Proposed system is gives better compression ratio as compare to existing system.

VIII. CONCLUSION AND FUTURE SCOPE

Comparing performance of the compression technique is difficult unless identical data sets and performance measures is used. After analysis of all techniques it is found lossless image compression techniques are most effective over lossy compression techniques, but lossy provides a higher compression ratio as compare to lossless. Image compression plays vital role in saving memory storage space and saving time while transmission images over network. Compression technique increase storage capability and transmission speed. Using compression coding techniques shares size in image secret sharing is reduced. By using Huffman coding image is compressed by 40 percent. Huffman algorithm is comparatively easier because of its simpler mathematical calculation in order to find various parameters. Lossy gives better compression but degrade quality. In feature one of lossy technique gives quality result for image compression.

REFERENCES

[1] Prof. A. A. Shaikh, Prof. P. P. Gadekar, " Huffman Coding Technique for Image Compression", ISSN:2320-0790, 2015.
[2] Rachit patel, virendra kumar, vaibhav tyagi, vishal asthana, "A Fast and Improved Image Compression Technique Using Huffman Coding", IEEE, PP. 2283-2286, 2016.
[3] Jagdish H. Pujar and Lohit M. K. Kadlaskar, "A New Lossless Method of Image compression and decompression using Huffman coding technique", Journal of Theoretical and Applied Information Technology, 2010.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

[4] Nehal Markandeya, Prof. Dr. Sonali Patil, "Image Compression Using Huffman Coding", ISSN: 2319-7242, Page No. 19999-20002, 1 Jan. 2017.
[5] Anitha. S, Assistant Professor, GAC for women, Tiruchendur, Tamil Nadu, India, " LOSSLESS IMAGE COMPRESSION AND DECOMPRESSION USING HUFFMAN CODING",e-ISSN: 2395-0056, p-ISSN: 2395-0072, Apr-2015.
[6] Lakhani, G, "Modified JPEG Huffman coding", IEEE Transactions Image Processing, 12(2), 2003 pp. 159 – 169.
[7] School of Science RK University, Rajkot, Gujarat, India, "Comparative Analysis of Lossless Image Compression Techniques for Color Image", January 2017.
[8] Anitha. S, Assistant Professor, GAC for women, Tiruchendur, TamilNadu,India, "Lossless Image Compression and Decompression using Huffman Coding", "www.irjet.net", April 2015.
[9] Ritu Antil, Sandeep Gupta "Analysis and Comparison of Various Lossless Compression Techniques","www.ijraset.com", March 2014.
[10] Nehal Markandeya, Prof. Dr. Sonali Patil, "Image Compression Using Hu_man Coding", International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 6, "www.ijecs.com", Januvary 2017.
[11] Ridhi Jindal, "A Review on Recent Development of Image Compression Techniques", International Journal of Advance Research and Development, "www.ijrand.com", March 2013.