IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

An Implementation in Image Compression Technique and its Effect on Image

Pradeep Kumar Atulker[1*], Rajendra Gupta[2]

[1,2]Rabindranath Tagore University, Raisen, Madhya Pradesh, India

DOI: https://doi.org/10.26438/ijcse/v7i2.945948 | Available online at: www.ijcseonline.org

Accepted: 26/Feb/2019, Published: 28/Feb/2019

Abstract- One of the significant aspects of image storage is its effective compression. The compression is a very important factor of the solutions available for creating file sizes of manageable and transmittable dimensions. In a scattered environment, the big images file remains a major bottleneck within systems. With the increasing in bandwidth by another method, the cost sometimes makes a less attractive solution. The goal of image compression is to reduce the number of bits wanted to represent an image by eliminating the spatial and spectral terminations as much as possible. This paper present the algorithm working behind the image compression and its implementation to achieve maximum possible compression in image without degrading its property. After analysis the results, it is found that the proposed algorithm reduces the image file size upto 86 per cent.

***Keywords* :** Image Compression, SPIHT, Image Quality, Quantization

I. INTRODUCTION

The Image compression is the process in which the size is minimized in bytes of several graphics image file type without degrading the quality of an image to an unacceptable level. The decrease in file size permits more images to be stored in a given extent of disk or the system memory space. In this compression process, the size is also decreases the time essential for images to be sent over the Internet or while downloading from web sources.

There are a number of different ways are available from which image files can be compressed. The most commonly graphic image file formats are GIF format and JPEG format, which can be used for any graphics processing task. The JPEG format is more often used for photographs, whereas the GIF type format is generally used for line art and other images in which the geometric shape is relatively simple.

Some already proposed techniques are also available for the image compression [1-3], which comprise of the use of wavelets and fractals. These two methods has not gained widespread acceptance for use on the Internet as of this writing. However, these methods offers promise because they offer higher compression ratio than other file formats like GIF or JPEG methods for some types of images. Another innovative method that may in time replace the GIF format is the PNG format which are not discussed in this paper.

Principle behind Compression

The image compression addresses the trouble of dropping the amount of data required to represent a digital image. A common distinctive characteristic of most of the images is that the adjacent pixels are interconnected and therefore contain terminated information. The foremost task then is to find less correlated representation of the image [4].

The fundamental base point of the reduction process is to the removal of unwanted data. From a mathematical point of view, the amount to transform a 2-D pixel array into a statistically uncorrelated data set, the transformation is applied prior to storage and transmission of the images. The compressed image is decompressed at later stage to renovate the original image or an approximation to it.

Basically, there are two fundamental components of compression that are *irrelevancy reduction* and *redundancy*. The irrelevancy decreases leave out parts of the signal that is not monitored by the signal receiver, other is *redundancy reduction* which aims at removing identical data from the signal source (image or video). Generally, there are 3 types of redundancy can be identified in an image;

- **Spectral Redundancy** : correlation between altered color planes or spectral bands.
- **Temporal Redundancy** : correlation between adjacent frames in a sequence of images which are in video applications.
- **Spatial Redundancy** : correlation between neighbour pixel values.

II. LITERATURE SURVEY

In the study of image processing and it compression, several research papers are shown their perspective. In [5]

Mathematics Coding is implemented for image compression. The author has elaborated the compression effectiveness and execution time of the programs, comprising the effect of unlike arithmetic word lengths on compression efficiency. The mathematics application coding are also depict in the research paper.

In [6] the recent control design success in tele-operation, which is operation at a distance, has been discussed. In recent times, the probable of hectic announcement has been recognized as being compelling to more augment human-to-human and human-to-machine communication. The tele-operation allows the human to perform management tasks in distant, scaled, hazardous, or unapproachable environments. Therefore, video and audio compression is considered key allowing technologies for high-quality interaction.

In [7], review of different basic lossless data compression methods has been explained. The algorithm charity for lossless compression is described in-short. They concluded that in the Statistical compression techniques, Arithmetic coding technique performs better than Huffman coding, Shannon Fano coding, and RunLengthEncoding (RLE) technique.

In [8], the author has discussed about a set of designated algorithms which are inspected and implemented to appraise the performance in compressing text data. An investigation on assessment of a number of different lossless data compression algorithms is presented. They accomplished by considering the compression times, decompression times and saving proportions of all the algorithms, the Shannon Fano algorithm is considered as the most productive algorithm among the selected ones.

The paper [9] in 2014 elucidated as an Integrated Neighborhood Reliant Approach for Nonlinear Enhancement of Color Images has proposed a new image enhancement algorithm INDANE (Integrated Neighborhood Dependent Approach for Nonlinear Enhancement of Color Images), which was applied to improve the discernibility of the dark-regions in digital images. The INDANE was a grouping of two independent processes: luminance enhancement and contrast enhancement. The luminance improvement, also regarded as a process of dynamic range compression, is essentially an intensity transformation based on a precisely designed nonlinear transfer function.

In [10] lossless data compression methodology have provided and their performances has been compared. The Huffman and arithmetic coding are compared, according to their performances. In comparison they decided that compression ratio of arithmetic encoding is better as compared to Huffman coding. Additionally arithmetic encoding reduces channel bandwidth and transmission time.

III. METHODOLOGY

The SPIHT (Set Partitioning in Hierarchical Trees) algorithm is an extension of conventional methods for image compression and it represents an important advancement in the field of space management. The SPIHT algorithm which is based on set partitioning in hierarchical trees is a well-organized image coding technique that is using wavelet transform. As compared to JPEG, the current standard for still image compression, the EZW and SPIHT are more efficient and reduce the blocking artifact. The algorithm defines spatial parent child relationship in the decomposition structure. It also describes the collocation with one to four parent child relationships, where the parent is in a sub-band of the same orientation as the child but at a smaller scale [11].

The algorithm consists of two passes; first is *ordering pass* and second one is *refinement pass*. In the ordering pass, the algorithm attempts to order the coefficients according to their magnitude. In the refinement pass, the quantization of coefficients is refined. The ordering and refining is made relative to a threshold value. The threshold value is appropriately initialized and then continuously made smaller with each round of the SPIHT algorithm. The algorithm maintains three lists of coordinates of coefficients in its decomposition, which are (i) List of Insignificant Pixels (LIP), (ii) List of Significant Pixels (LSP) and (iii) List of Insignificant Sets (LIS).

The performance of SPIHT algorithm forms a hierarchical quad-tree data structure for the wavelet transformed coefficients. The set of root node and corresponding descendents are referred to as a spatial orientation tree (SOT). The tree is defined in such a way that each node has either no leaves or four offspring, which are from 2 x 2 adjacent pixels.

For the convenience of illustrating the real implementation of SPIHT, the following sets of coordinates are defined.

O (i, j): set the coordinates of all offspring of node (i, j).
D (i, j): set of coordinates of all descendants of node (i, j).
H: set of coordinates of all spatial orientation tree roots (nodes in the highest pyramid level).
LIP: list of the Insignificant Pixels
LIS: list of the Insignificant Sets
LSP: list of the Significant Pixels

$$L(i, j) = D(i, j) - O(i, j)$$

A LIS entry is of type A, it represents D (i, j), type B, if it represents L (i, j).
The steps of the algorithm are as follows:

(1) Initialization: Output $n = \log_2(\max_{(i,j)} \{|C_{i,j}|\})$;
Set the LSP as an empty list;
Add the coordinates $(i, j) \in H$ to the list LIP, and only those with descendants also to LIS.

(2) Sorting pass:

(i) for each entry (i, j) in the LIP do:

(i.1) transmit S_n (i, j);

(i.1.2) if S_n (i, j) = 1 then move (i, j) to the LSP and transmit the sign of Ci, j;

(i.2) for each entry (i, j) in the LIS do:

(i.2.1) if the entry is of type A then

• transmit S_n (D (i, j));

• If S_n (D (i, j)) = 1 then for each (i, j) ∈ O (i, j) do:

• transmit S_n (i, j);

• If S_n (i, j) = 1 then add (i, j) to the LSP and output the sign of Ci,j;

• If Sn (i, j) = 0 then add (i, j) to the end of the LIP;

If L (i, j) = ∅ then move (i, j) to the end of the LIS, as an entry of type B, and go to step (i.2.2); Otherwise, remove entry (i, j) from the LIS;

(i.2.2) if the entry is of type B then

• transmit Sn (L (i, j));

• If Sn (L (i, j)) = 1 then add each (i, j) ∈ O (i, j) to the end of the LIS as an entry of type A; Remove (i, j) from the LIS.

(3) **Refinement pass:** For each entry (i, j) in the LSP, except those included in the last sorting pass, output the *n*th most significant bit of |Ci, j |;

(4) **Quantization-step update:** Decrement *n* by 1 and go to step (2).

The SPIHT algorithm continues with the refinement pass which is the implementation suggested in this algorithm to achieve better outcomes. In the refinement pass, the very next bit in the binary representation of the coefficients in LSP gives result. The next-bit is related to the current threshold value. The processing of LSP ends one round of the SPIHT algorithm, before the next round starts the current threshold is becomes halved.

IV. RESULT & DISCUSSION

MATLAB implementation of the algorithm

For the evaluation of the performance of proposed implemented algorithm, implementations are applied on a number of images of square resolutions of power of 2 and the grayscale colour image. It produces a reconstructed output in spatial domain for comparison and also automatically Peak Signal to Noise Ratio as a measure of the difference between source and compressed images. After implementation in existing image processing compression algorithm, it is found that the implemented algorithm outperform and shows around 86 percentage successful results.

The following figure 1 shows the size of original and compressed images.

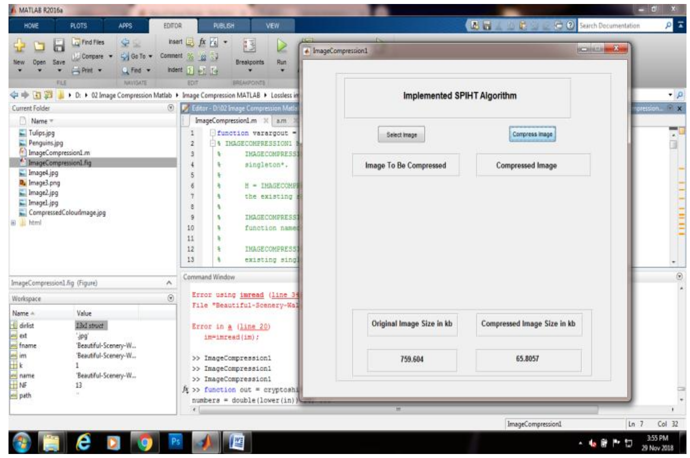

Fig. 1 : Image Compression Result using Implemented SPIHT Algorithm

The above implemented algorithm has been executed in MATLAB R2016a. While executing the Matlab code for the image, we have taken a scenery image. First we have compressed the image with existing SPIHT algorithm and then compression has been done with proposed implemented algorithm. After performing several experiments, it is found that the compression ratio is coming around 80-86 percent successful results.

V. CONCLUSION

The purpose of the research work is to analyze the performance of the different image compression and enhancement techniques and obtain the best possible outcome. Each compression algorithm has several parameters that can be modified to improve the quality, increase the compression ratio for same quality or reduce the artifacts. The existing compression techniques are further enhanced using coding techniques.

After the study of image compression, it is found that the JPEG optimization can give a 80% to 85% gain in compression with proper optimization of the quantization at the same compression rate. To the best of our work, the proposed algorithms are having detailed implementation of algorithm to reduce the size and space of an image. The algorithms reduce the computational complexity and it is complement to the efficient work with concern to image processing. The SPIHT algorithm is not a simple extension of traditional methods for image compression but also it represents an important advance in the field of image processing. This algorithm also deserve an special attention because it provides the recompense like highest image quality, progressive image transmission. After the comparison with the earlier proposed algorithm, the implemented algorithm shows better outcome.

REFERENCES

[1] Mohammed Al-laham1 and Ibrahiem M. M. El Emary, “Comparative Study Between Various Algorithms of Data Compression Techniques”, Proceedings of the World Congress on Engineering and Computer Science (WCECS 2007), 2017, San Francisco, USA.

[2] AnilKatharotiya, Swati Patel, Mahesh Goyani, “Comparative Analysis between DCT and DWT Techniques of Image Compression” *Journal of Information Engineering and Applications* Vol 1, No.2, 2016.

[3] GauravVijayvargiya, Sanjay Silakari and Rajeev Pandey, “A Survey: Various Techniques of the Image Compression", International Journal of Computer Science and Information Security, Volume 11, No. 10, October 2014

[4] Jau-JiShen and Hsiu-Chuan Huang, “An Adaptive Image Compression Method which is Based on Vector Quantization” ,IEEE, pp. 377-381, 2015.

[5] S. A. Al-Dubaee and N. Ahmad, “New Strategy of Lossy-Text Compression”,Integrated Intelligent Computing (ICIC), 2014

[6] Suresh Yerva,Smita Nair and Krishnan Kutty,“Lossless Image Compression based on Data Folding Method”,‖IEEE, pp. 999-1004, 2015.

[7] A. Alarabeyyat, S. Al-Hashemi1, T. Khdour1, M. Hjouj Btoush1,S.Bani-Ahmad1, R. Al-Hashemi “The Lossless Image Compression Technique Using Combination Methods”, Journal of Software Engineering andApplications, 2016.

[8] Vartika Singh **“**A Brief Introduction on Image CompressionTechniques and Standards”, International Journal of Technology and ResearchAdvances, Volume of 2016 issue II.

[9] Yu-Ting Pai, Fan-Chieh Cheng, Shu-Ping Lu, and Shanq-Jang Ruan, “Sub-Trees Modification of Huffman Coding for Stuffing Bits Reduction and Efficient N-R-Z-I Data Transmission”, IEEE Transactions On Broadcasting, Vol.58,No.2, June 2016

[10] Mamta Sharma, “Compression using Huffman-Coding”,International Journal of Computer Science and Network Security, Vol.10, No.5,May 2015

[11] Mohammed Al-laham1 and Ibrahiem M. M. El Emary, “Comparative Study Between Various Algorithms of Data Compression Techniques”, Proceedings of the World Congress on Engineering and Computer Science (WCECS 2015), 2015.

