Analysis and Implementation in Image Compression Technique to retain the Quality of Image with reducing the file size

[1]Ankur Gupta , [2] Pritaj Yadav, [3] Mukesh Kumar
[1]M.Tech Scholar , [2]Associate. Professor, [3] Head of Department
[1]Department of Computer Science & Engineering, [1]RNTU, BHOPAL,INDIA

Abstract

The compression is a very significant component of the solutions available for creating file sizes of controllable and transmittable dimensions. In a distributed environment, the large images file remains a major bottleneck within systems. With increasing the bandwidth by another method, the cost sometimes makes this a less attractive solution. In this paper, we have discussed some techniques based on colour image compression and implemented SPIHT algorithm and found some fruitful results.

1. Introduction

The Image compression is the process in which the size is minimized in bytes of any graphics file type without degrading the quality of an image to an unacceptable level. The decrease in file size allows more images to be stored in a given extent of disk or the system memory space. In this process, the size is also reduces the time required for images to be sent over the Internet or downloaded from Web pages.

There are a number of different ways by which image files can be compressed. The two most common compressed graphic image formats are the JPEG format and the GIF format for the use in any graphics processing task. The **JPEG** format is more often used for photographs, while the **GIF** format is normally used for line art and other images in which geometric shapes are relatively simple.

Some other techniques are available for image compression, which include the use of **fractals** and **wavelets**. These two methods has not gained widespread acceptance for use on the Internet as of this writing. However, these both methods offer promise because of they offer higher compression ratio than other file format like JPEG or GIF methods for some types of images. Another novel method that may in time replace the GIF format is the **PNG** format.

The compressing an image is significantly different than compressing the raw type binary data. The general purpose compression program can be used to compress image, nonetheless the result is less than optimal. Of course, the images have certain statistical properties which can be exploited by encoders specifically designed for them. Moreover, some of the finer details in the image can be sacrificed for the sake of saving a little more bandwidth or storage space. It also means that lossy compression techniques can be used in this area.

1.1 Process of Image Compression

There are two types of image compression; lossy and lossless. A typical lossy image compression system consists of three closely connected components namely
(a) Source Encoder
(b) Quantizer
(c) Entropy Encoder

The compression is accomplished by applying a linear transform to de-correlate the image data; than quantizing the resulting transform coefficients and after that entropy coding for quantized values.

(a) Source Encoder (or Linear Transformer):

In past few year, a variety of linear transforms have been developed which include the system like Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) etc. Each are having its own advantages and disadvantages.

(b) Quantizer:

Quantization can be performed on each individual coefficient, which is known as Scalar Quantization (SQ). The quantization can also be performed on a group of coefficients together, and this is known as Vector Quantization (VQ). The quantizer process simply reduces the number of bits which are needed to store the transformed coefficients by reducing the precision of the applied values. Since it is a many-to-one mapping which is a lossy process and the main source of compression is an encoder. Both uniform and non-uniform quantizers can be used depending on the problem at hand.

The quantization process is as given below :

- A many o one mapping that reduces the number of probable signal values at the cost of introducing errors.
- The simplest form of quantization which is also used in all the compression standards is a **scalar quantization** (SQ), in which each signal value is individually quantized.

(c) Entropy Encoder

The third process is entropy encoder which further compresses the quantized values by losslessly to give improved overall compression. This process uses a model to accurately determine the probabilities for each quantized value and produces an appropriate code based on the probabilities, so that the resulting output code stream would be smaller than the input stream. The most commonly used entropy encoders are the Huffman encoder and the arithmetic encoder, although for applications requiring fast execution, simple Run Length Encoding (RLE) has proven very effective.

It is important to understand that the properly designed quantizer and entropy encoder are absolutely necessary along with optimum signal transformation to get the best possible compression.

---

2. Literature Review

In [19] Arithmetic Coding is implemented. The author has elaborated the compression efficiency and execution time of the programs, including the effect of different arithmetic word lengths on compression efficiency. The applications of arithmetic coding are also described in the paper.

In [10] the recent control design successes in tele-operation, which is operation at a distance are discussed. In recent times, the potential of haptic communication has been recognized as being compelling to further augment human-to-human and human-to machine interaction. The tele-operation enables the human to perform manipulation tasks in distant, scaled, hazardous, or inaccessible environments. Therefore, video and audio compression is considered key enabling technologies for high-quality interaction.

In [7] reviews of different basic lossless data compression methods are explained. The algorithm used 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 Run-Length-Encoding technique.

In [20], the author has talked about a set of selected algorithms are examined and implemented to evaluate the performance in compressing text data. An experimental comparison of a number of different lossless data compression algorithms is presented. They accomplished by considering the compression times, decompression times and saving percentages of all the algorithms, the Shannon Fano algorithm can be considered as the most productive algorithm among the selected ones.

The paper [7] in 2014 explained as an Integrated Neighborhood Dependent Approach for Nonlinear Enhancement of Color Images. Proposed a new image enhancement algorithm INDANE (Integrated Neighborhood Dependent Approach for Nonlinear Enhancement of Color Images), which was applied to improve the visibility of the dark regions in digital images. INDANE was a combination of two independent processes: luminance enhancement and contrast enhancement. The luminance enhancement, also regarded as a process of dynamic range compression, is essentially an intensity transformation based on a specifically designed nonlinear transfer function.

In [8] lossless data compression methodologies are provided and their performances are compared. The Huffman and arithmetic coding are compared according to their performances. In comparison, they concluded that compression ratio of arithmetic encoding is better as compared to Huffman coding. Furthermore arithmetic encoding reduces channel bandwidth and transmission time.

At last the author implemented with nonlinear transfer function and multi-scale convolutions in image processing. The enhanced image good quality has been confirmed. The author [8] in 2014 presented a color image compression using adaptive color quantization.

---

3. Factors behind Image Compression

3.1 Basic Strategy in Image Compression

Following are the key point for the basic strategy of image compression:

- Ideally, it is often necessary to throw away both non-redundant information and relevant information to achieve the required compression.
- Practically, an image compression technique removes redundant and/or irrelevant information, and efficiently encodes what remains.
- In either case, the trick is finding methods that permit important information to be competently extracted and represented.

3.2 Some Factors Affecting Achievable Compression

During image compression, some factors can play a vital role for giving effective results which are as under;

- Sample Parameters
- Image size and viewing distance
- Sensor Characteristics
- Scene content including noise
- Post Processing (Sharpening, Dynamic Range Adjustment (DRA), Tone Transfer Curve (TTC))
- Display characteristics (noise, light level, non-linearities)
- Pre-Processing (image formation, registration)
- Required task

3.3 Different Classes of Compression Techniques

There are two ways of classifying compression techniques which are mentioned here.

(a) Lossless and Lossy compression

In lossless compression schemes, the reconstructed image, after compression, is numerically identical to the original image. However lossless compression can only achieve a modest amount of compression. However, lossy schemes are capable of achieving much higher compression. Under normal viewing conditions, no visible loss is perceived (visually lossless). An image reconstructed lossy compression contains degradation relative to the original. This is because the compression scheme completely discards redundant information.

The information loss in lossy coding comes from quantization of the data. The quantization can be explained as the process of sorting the data into different bits and representing each bit with a value. The value selected to represent a bit is called the reconstruction value. Every item in a bit has the same reconstruction value, which leads to information loss (unless the quantization is so fine that every item gets its own bit).

Lossless or Reversible Compression

Following points can be discussed for lossless compression :

- Image compression techniques such as lossless JPEG or JPEG-LS perform slightly better.
- The image after compression and decompression is identical to the original.
- Compression ratios are typically 2:1 for natural imagery but can be much larger for document images.
- Only the statistical redundancy is exploited to achieve compression.
- Data compression techniques such as LZW or LZ77 are used in GIF, PNG and TIFF file formats

Lossy (Irreversible) Compression

Following points can be discussed for lossless compression :

- Both the statistical redundancy and the perceptual irrelevancy of image data are exploited.
- Image quality can be traded for compression ratio.
- The reconstructed image contains degradations with respect to the original image.
- Much higher compression ratios compared to lossless.
- The term visually lossless is often used to characterize lossy compression schemes that result in no visible degradation under a set of designated viewing conditions.

(b) Predictive vs. Transform coding:

In the predictive coding, the information already sent or available is used to predict future values, and the difference is coded. Since this is done in the image or spatial domain, it is relatively simple to implement and is readily adapted to local image characteristics.

The Differential Pulse Code Modulation (DPCM) is one particular example of predictive coding. Transform coding, on the other hand, first transforms the image from its spatial domain representation to a different type of representation using some well-renowned transform and then codes the transformed values i.e. coefficients. This method provides greater data compression compared to predictive methods, although at the expense of greater computation.

(c) Discrete Wavelet Transform (DWT)

The discrete wavelet transform (DWT) decomposes one-dimensional (1D) sequence like line of an image into two sequences which is called sub-bands), each with half the number of samples, according to the following procedure:

- The filtered signals are downsampled by a factor of two to form the low-pass and high-pass subbands.
- The 1D sequence is separately low-pass and high-pass filtered.
- The two filters are called the analysis filter bank.

---

4. Methodology

4.1 SPIHT Method and its implementation

The SPIHT (Set Partitioning in Hierarchical Trees) algorithm is a simple extension of conventional methods for image compression and it represents an important advance in the field of space management. The SPIHT (which is based on set partitioning in hierarchical trees) is an well-organized image coding technique using the wavelet transform. In recent times, image-coding using the wavelet transform has attracted great attention. In most of the coding algorithms the surrounded zero tree wavelet coding by Shapiro and its implemented version the set of partitioning in hierarchical trees (SPIHT) by Said and Pearlman become very successful. As compared to JPEG - the current standard for still image compression, the EZW and the SPIHT are more efficient and reduce the blocking artifact.

The method provides the following qualities and deserves special attention because it:

- Good image quality, high PSNR, especially for color images;
- Efficient combination with error protection.
- Simple quantization algorithm;

It can be noticed here that different compression methods were developed specifically to achieve at least one of those objectives. The SPIHT really outstanding is that it yields all those qualities simultaneously.

The performance in 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.

The pixels on the sub-image of the highest decomposition level are the tree roots and are also grouped in 2 x 2 adjacent pixels. However, the upper-left pixel in 2 x 2 adjacent pixels has no descendant. Each of the other three pixels has four children.

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

O (i, j): set of 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, if 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 the 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).

---

5. Result & Discussion

5.1 MATLAB implementation of the algorithm

In the research study, we have proposed an implementation in SPIHT, LZW and RLE algorithm. This implementation is applied on images of square resolutions and grayscale only. 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 algorithms, it is found that the implemented algorithms outperform and shows more than 85 percentage successful results.

	Existing Algorithm			Implemented Algorithm		
	SPIHT	LZW	RLC	SPIHT	LZW	RLC
Compression Ratio	78%	72%	67%	86%	89%	76%

The following figure shows the size of original and compressed images. In all the case conditions, the images are compressed upto 70 percentages level. The compression ratio is almost same for SPIHT and LZW, but the RLE algorithm do not compress the images more than 50 percentages.

The SPIHT and LZW implementation produces the best results in image compression. The only real choice that produces comparable result are the very similar bio-rthogonal wavelet families which is called 'bior' in MATLAB.

---

6. Conclusion

The 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. After study the image compression, we have found the following observations :

– JPEG optimization can give a 35% to 65% gain in compression with proper optimization of the quantization at the same compression rate
– Parameters are optimized for the characteristics of the image and the requirements of the compression applications
– Parameters can be modified to reduce identified artifacts which may be the interaction between the compression algorithm, the image characteristics, post processing and the display process.
- Optimization process is common for a class of imagery or image characteristics
- Optimization is also common for a desired bit rate

The research study provides an analysis for the case where all the images in any format gives the compressed results. Also, described highly scalable SPIHT, LZW and Run Length Coding algorithms with its implementation to better perform in image compression. To the best of our knowledge, the work proposes a detailed implementation of algorithm to reduce the space of image. The algorithms reduce the computational complexity and it is complement to the efficient work with concern to image processing.

7. References

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

[2] Anil kumar Katharotiya, Swati Patel, Mahesh Goyani, "Comparative Analysis between DCT & DWT Techniques of Image Compression" *Journal of Information Engineering and Applications* Vol 1, No.2, 2016.

[3] Gaurav Vijayvargiya, Sanjay Silakari and Rajeev Pandey, "A Survey: Various Techniques of 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 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", ‖IEEE, pp. 999-1004, 2015.

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

[8] Vartika Singh "A Brief Introduction on Image Compression Techniques and Standards", International Journal of Technology and Research Advances, 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 NRZI 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] Fouzi Douak, Redha Benzid and Nabil Benoudjit, "Color image compression algorithm based on the DCT transform combined to an adaptive block scanning", AEU-International Journal of Electronics and Communications, vol.65, Issue 1, pp. 16-26 , Jan 2015.

[12] Basar Koc, Ziya Arnavut and Huseyin Kocak, "Lossless Compression of Dithered Images", IEEE Photonics Journal, Volume 5, Number 3, June 2015

[13] Rajinder Kaur , Mrs. Monica Goyal, "A Survey on the different text data compression techniques", International Journal of Advanced Research in Computer Engineering and Technology (IJARCET) Volume 2, Issue 2, February 2014

[14] Jayavrinda Vrindavanam, Saravanan Chandran, Gautam K. Mahanti, "A Survey of Image Compression Methods" International Journal of Computer Applications, IJCA 2016.

[15] Shrusti Porwal, Yashi Chaudhary, Jitendra Joshi, Manish Jain, "Data Compression Methodologies For Lossless Data And Comparison Between Algorithms", International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 2, March 2014

[16] M. R. Islam and S. A. Ahsan Rajon, "An enhanced scheme for lossless compression of short text for resource constrained devices", International Conference on Computer and Information Technology, 2014

[17] Sandra Hirche, Senior Member IEEE, and Martin Buss, "Human-Oriented Control for Haptic Teleoperation", Proceedings of the IEEE, Vol.100, Issue 3, 2015

[18] Firas A. Jassim and Hind E. Qassim, "Five Modulus Method for Image Compression", SIPIJ Vol.3, No.5, pp. 19-28, 2015.

[19] Mamta Sharma, "Compression Using Huffman Coding", International Journal of Computer Science and Network Security IJCSNS, VOL.10 No.5, May 2013

[20] Zheng Yunping , Chen Chuanbo and Sarem Mudar, "A NAM Representation Method for Data Compression of Binary Images", TSINGHUA SCIENCE AND TECHNOLOGY, Volume 14, Number 1, February 2013

[21] Md. Rafiqul Islam and S. A. Ahsan Rajon and Anonda Podder "Short text compression for smart devices", Computer and Information Technology, ICCIT 2012

[22] IAN H. Willen, Radford M. Neal, and John G. Cleary, "Arithmetic Coding for Data Compression", June 2011, Volume 30

[23] Data compression using encrypted text, R. Franceschini and A. Mukherjee, Digital Libraries, ADL 2012, Proceedings of the Third Forum on Research and Technology Advances

[24] Pu, I.M., 2012, Fundamental of Data Compression, Elsevier, Britain.

[25] S.R. Kodituwakku, U. S.Amarasinghe, S.R. Kodituwakku "Comparison of Lossless Data Compression Algorithms for Text Data", Indian Journal of Computer Science and Engineering, Vol 1 No 4 416-425, 2011