A JPEG-Discrete Cosine Transformation Based Coding Technique for Image Data

A L Choodarathnakara

Abstract— **Image compression plays a major role in modern multimedia communications and internet traffic. JPEG is one of the standard algorithms which are being widely used. The purpose of this paper is to develop a compression algorithm which satisfies JPEG standard. This algorithm is being applied on Gray level images; with few additions it will be applicable for RGB images also. A useful property of JPEG is that the compression ratio can be varied by varying quality factor. Quality Factor of 25 can be achieved by maintaining a good quality of image.**

Index Terms—**Image Compression, Joint Photographic Experts Group, Discrete Cosine Transform. For list of image file formats and compression standards, visit the web site at http://www.jpeg.org**

NOMENCLATURE

DCT	Discrete Cosine Transform
GIF	Graphical Interchange Format
JFIF	JPEG File Interchange Format
JPEG	Joint Photographic Experts Group
MSE	Mean Square Error
PSNR	Peak Signal to Noise Ratio
RMSE	Root Mean Square Error
TIFF	Tagged File Format

I. INTRODUCTION

The basic Multimedia data types involve Text, Audio, Full Motion Video and Images. Amongst text is the simplest of all data types and requires the least amount of storage space. Hypertext is an application of indexing text to provide a rapid search of specific text strings in one or more documents. From the perspective of multimedia applications, a hypermedia document is the basic complex object of which text is a sub-object. A hypermedia document almost always has text and may in addition; it can have one or more sub-objects such as Image, Sound and Full Motion Video.

Images include Document images, Facsimile systems, Bitmaps, Metafiles and Still pictures. Visible images include Drawings (Blue prints, Engineering drawings, Town layouts and so on), Documents (scanned as images), Paintings (scanned, computer based paint applications), Photographs (scanned or entered directly by electronic camera) and Still frames captured from video camera. In all these cases, the images include every pixel captured by the input device. Non-visible images are those that are not stored as images but are displayed as images. Examples include Pressure gauges, Temperature gauges and other metering displays. Abstract images are really not images that ever existed as real-world objects or representations. Rather they are computer-generated images based on some arithmetic calculations. The discrete functions result in still images and continuous functions are used to show animated images.

When Multimedia objects like binary document images, gray scale images, color images, photographic or video images, audio data or video data and full motion video are digitized, large amounts of digital data are generated. There has been considerable debate on whether compression and decompression should be performed in specialized hardware or in software. The two driving parameters for this discussion are performance and cost. Higher the performance more will be the cost. Therefore a tradeoff is necessary between performance and the cost. Now a day, it is possible to use CPU for most applications, very high resolution imaging applications and documents using JPEG or MPEG standard components will continue to require Digital Signal Processors to provide acceptable performance.

II. LITERATURE SURVEY

Since, the work belongs to the area of Image Processing and Compression, it is necessary to understand the concept of compression techniques and Image processing. Hence, an extensive reference has been made to various textbooks and IEEE papers.

The basic of image processing is better understood by referring to R C Gonzalez and Woods [22]. Kenneth R Castleman [16] book discuss the computer hardware and image processing system. He explains the fundamentals with good examples. However he doesn't give any source code.

The concept of computer vision and image processing are explained by Martin D L. He discusses many of the biological vision and perception systems as well as computer algorithm. Craig A Lindley [7] book contains source code written exclusively for personal computers. The book explains the various building blocks using an image digitizer. He covers TIFF specifications as well as PCX file formats. The source code is written on Turbo C compiler.

John C Russ [14] book covers many of the image processing algorithms along with their applications. Arthur R Weeks Jr [2], not only discuss basic image processing along with applications. The software accompanying the book can read images in TIFF, GIF, BMP and PCX file formats. He blends theory and practice very well.

Various Transforms are used in image processing also. Some of the image transforms are used in the area of image compression. They are DCT, Hadamard, Haar, Walsh, KLT and Wavelet transform. Wavelet and Haar transforms are widely used in fractal image processing. Anil K Jain [1],

Manuscript received April, 2015.
A L Choodarathnakara*, Department of Electronics & Communication Engineering, Government Engineering College, Kushalnagar-571234, Kodagu District, Karnataka State, INDIA,*

Milan Sonaka [19], Weidong kou [27] gives performance comparison of different image transforms. They also discuss the best image transform for a particular application.

Image compression is based on the removal of any redundant data present in the image. The fundamentals of image compression techniques are described with special reference to multimedia applications by Anil K Jain [1], Gonzalez [22] and K R Rao, Y C Yip [18]. Various coding techniques are used for compression purpose. Coding techniques are excellently explained by Sam Shanmugam [24]. The theoretical and practical aspects of image compression are excellently described by Ze-Nian Li and Mark S Drew [29]. They also explained the compression formats and standards for data, images, audio and video. They also considered both loss and lossless compression.

Compression techniques fall into two categories, Information preserving and Lossy. If the image is a medical image and satellite image then we prefer error free compression technique. However, for other applications we prefer lossy compression. In modern day computing data compression is one of the important fields.

David Salomon [8] provides necessary consideration for image compression. An image compression method is specified for a specific type of images. New data compression methods that are developed and implemented have to be tested. Testing different methods on the same data makes it possible to compare their performance both in compression efficiency and speed. Hence, we have standard test data namely, ‘Lena’, ‘Mandril’ and ‘Peppers’. They are continuous tone images although ‘Lena’ image is mostly pure continuous tone, especially the wall and the bare skin areas. The hat is good continuous tone, where the hair and the plume on the hat are bad continuous tone. The straight lines on the wall & curved parts are features of discrete tone image. There are various approaches to compress the image. Various authors have discussed the different approaches with their relative advantage and disadvantage [1], [4], [5], [8], [9], [11], [15], [17], [21], [25], [27].

The implementation work is carried out on a TMS processor. For this Code Composer Studio (CCS) and C language is used. The processor fundamentals and the Code Composer Studio are discussed by Bhaskar and Venkatramani [5], Rulph Chassing [23], Avtarsingh and Srinivasan [3]. To implement this work a good knowledge of C programming and hands on experience on TMS processor is required. Reference has been made to Yashwanth kanitkar [28] and Code Composer Studio [6].

III. IMAGE COMPRESSION

Every day, an enormous amount of information is stored, processed and transmitted digitally. Because much is online/offline information is graphical or pictorial in nature, the storage and communication requirements are immense.

The Table 1 indicates, the bandwidth required and time required for transmission on a particular mode. Since the size of the images being different and the transmission time required also varies. It is necessary to select proper compression ratio so that bandwidth can be utilized effectively. In this implementation work we explore these quantities and we have considered Gray level images for compression along with RGB color images and a medical image. Based on the results obtained a comparative study has been made.

A. Image Compression Model

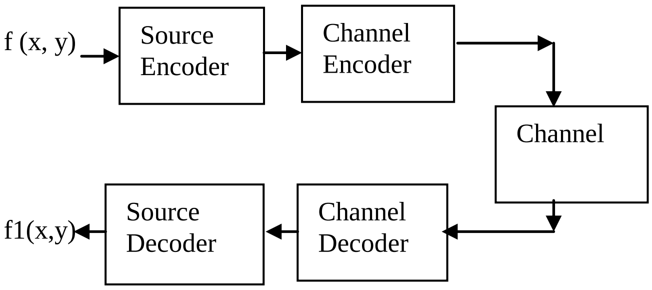


Fig 1: A General Compression System Model

A compression system consists of two distinct structural blocks: an encoder and a decoder. An input image f(x, y) is fed into the encoder, which creates a set of symbols from the input data. After transmission over the channel, the encoded representation is fed to the decoder, where a reconstructed output image f1(x, y) is generated. In general f1(x, y) may or may not be an exact replica of f(x, y). If it is, the system is error free or information preserving; if not some level of distortion is present in the reconstructed image.

Both the encoder and decoder shown in Fig. 1 consist of two relatively independent functions or sub blocks. The encoder is made up of a source encoder, which removes input redundancies, and a channel encoder, which increases the noise immunity of the source encoder’s output. As would be

Multimedia	Size/ Duration	Bits/pixel Bits/sample	Un-compressed size (B=Bytes)	Transmission Bandwidth (b-Bits)	Tx time using 28.8kbps modem
A page of Text*	11”x 8.5”	Varying resolution	4 – 8KB	32 – 64kb/page	1.1-2.2sec
Telephone Speech*	10sec	8bps	80KB	64Kb/sec	22.2sec
Gray scale Image	512x 512	8bpp	262KB	2.1Mb/image	1min13sec
Color Image	512x512	24bpp	876kB	6.29Mb/image	3min39sec
Medical Image*	2048x1680	12bpp	5.16MB	41.3Mb/image	23min54sec
Full Motion Video*	640x480 1min (30 frames/sec)	24bpp	1.66GB	221Mb/sec	5days8hrs

*Compression/details of these are out of scope of this paper

Table 1: Comparisons of Requirements of Different Multimedia Data

expected the decoder includes a channel decoder followed by a source decoder. If the channel between the encoder and the decoder is noise free, the channel encoder and decoder are omitted.

Two types of models are available for image compression:

1. The Source Encoder and Decoder
2. The Channel Encoder and Decoder

Basically there are three types of redundancies. Data compression is achieved when one or more of these are reduced or eliminated.

- Coding Redundancy
- Interpixel Redundancy
- Psychovisual Redundancy

Based on the requirements of reconstruction, data compression schemes can be broadly classified into:

- Lossless compression
- Lossy compression

B. Lossless Compression

Lossless compression techniques as the name implies, involve no loss of information. If data have been losslessly compressed, the original data can be recovered exactly from the compressed data. Lossless compression is generally used for applications that cannot tolerate any difference between original and reconstructed data.

C. Lossy Compression

Lossy compression has algorithm based on compromising the accuracy of reconstructed image in exchange for increased compression. If the resulting distortion can be tolerated, the increase in compression can be significant. The term lossy compression and irreversible information loss might sound horrible at first. But the amount of information lost in most lossy compression techniques is nothing compared to the information lost in the original scanning of the image, due to the hardware limitation.

Lossy coding is based on Discrete Cosine Transform (DCT). It allows substantial compression to be achieved in reconstructed image with respect to original image. The simplest DCT based coding process is Baseline sequential process. It provides a capability, which is sufficient for different kinds of applications.

The primary purpose of lossy image compression is:

- Reduce the memory required for their storage.
- Reduce the effective data access time when reading the storage devices.
- Reduce the bandwidth.

D. Image Fidelity Criteria

The fidelity of an image after compression is an important aspect of image compression. Compression methods that are lossless produce images that are an exact replica of the original uncompressed image. Lossy compression methods that remove visual redundancies produce uncompressed images that have lost visual information during the compression process.

An image fidelity criterion is useful for measuring image quality and for rating the performance of a processing technique or a vision system. There are two types of criteria that are used for evaluation of image quality, subjective and quantitative. The subjective criteria use rating scales such as goodness scales and impairment scales. A goodness scale may be a global scale or a group scale. The overall goodness criterion rates image quality on a scale ranging from excellent to unsatisfactory.

E. JPEG Compression Standard

JPEG is the acronym for Joint Photographers Experts Group. JPEG is compression method for color and grayscale still images. It is not applicable for bi-level images very well. JPEG is designed as a compression method for continuous tone images. Its main goals are;

- High compression ratios, especially in cases where quality is judged as very good to excellent.
- The use of many parameters, allowing sophisticated users to experiment and achieve the desired compression/quality tradeoff.
- Obtaining good results with continuous tone image, regardless of image features (size, color space etc).
- A sophisticated but not a complex compression method, allowing software and hardware implementations on many platforms.

JPEG has four distinct modes of operations. Namely,

1. Sequential DCT based process
   - Decoder scans with above1, 2, 3 and 4.
   - DCT based process.
   - Source image: 8-Bit samples with in each component.
   - Sequential.
   - Huffman coding: 2AC and 2DC tables.
2. Progressive DCT based process
   - Decoder scans with above 1, 2, 3 and 4.
   - DCT based process.
   - Source image: 8-Bit/ 12-Bit.
   - Sequential/progressive.
   - Huffman/Arithmetic coding: 4AC and 4DC tables.
3. Lossless process
   - Decoder scans with above 1, 2, 3 and 4.
   - Predictive process, not DCT based.
   - Source image: p-Bits ($8 \le p \le 16$).
   - Sequential.
   - Huffman/Arithmetic coding: 4DC tables.
4. Hierarchical process
   - Decoders scans with above 1 and 2
   - Multiple frames.
   - Uses extended DCT based or lossless process.

Advantages:

- Applicable for both gray level (8 bit) as well as Color images (24 bit).
- Customizable Quantization tables and Huffman tables provide flexibility to users for developing the algorithm.
- The resultant image is capable of representing 8 bit as well as 24 bit images. This provides a plus point for resultant image over GIF images.
- JPEG files are supported by all imaging softwares.

Dis-advantages:

- JPEG is a lossy compression. Therefore a tradeoff between quality of image and quality factor is necessary.
- Not well applicable for Binary images.
- Since it is lossy compression, it is not very well suitable for Medical and Satellite images.

F. Run Length Coding

RLE is a natural candidate for compressing graphical data. A digital image is made up of small dots called pixels. Each pixel can be either one bit indicating a black or a white dot, or several bits, indicating shades of gray. We assume that the pixels are stored in an array called a bitmap in memory, so the bitmap is the input stream for the image. Pixels are normally arranged in the bitmap in scan lines, so the first bitmap pixel is the dot at the top left corner of the image, and the last pixel is the one at the bottom right corner.

Each run of pixels of the same intensity (gray-level) is encoded as a pair (run length, pixel value). The run length usually occupies one byte following for runs upto 255 pixels. The pixel value occupies several bits, depending on the number of gray levels (typically between 4 and 8 bits). Example: an 8 bit deep gray scale bitmap that starts with 12,12,12,12,12,12,12,12,12,35,76,112,67,87,87,87,5,5,5,5,5 ,5,1….. is compressed into 9, 12, 35, 76, 112, 67, 3, 87, 6, 5, 1….Where underlined numbers indicates counts. The problem is to distinguish between a byte containing a grayscale value (such as 12) and one containing a count (such as 9). Here are some solutions:

- If the image is limited to just 128 gray scales, we can devote one bit in each byte to indicate whether the byte contains a gray scale value or a count.
- If the number of gray scale is 256, bit can be reduced to 255 with one value reserved as a flag to precede every byte with a count. If the flag is, say, 255, then the sequence above becomes 255,9,12,35,76,112,67,255,3,87,255,6,5,1….

In this type of coding there is always a danger of actually increasing the size of data, instead of decreasing, so usually some other pre-filtering methods are employed to ensure that the data to be compressed contains large sequences of the same values. One such method is the utilization of color quantization methods in lossy compression.

In this work we have considered Run Length Coding in detail. It is possible to implement the work using RLC as well as Huffman coding.

Advantages:

- Better image quality at the same file size.
- To make the image files smaller and having them appear virtually the same as the original. Making image files smaller is a plus for transmitting files across networks and for archiving libraries of images.
- If we compare GIF and DCT based compression, the size ratio is usually more like 4:1.
- DCT based compression is far more useful than GIF for exchanging images among people with widely varying display hardware i.e., Low complexity option for devices with limited resources
- Scalable image files—no decompression needed for reformatting. With DCT compression standard, the image that best matches the target device can be extracted from a single compressed file on a server. Options include:
  - Image sizes from thumbnail to full size.
  - Grayscale to full 3 channel color.
  - Low quality image to lossless.
  - Progressive rendering and transmission through a layered image file structure. Example: from a single 100Kbytes image file of a 512 * 512 original image, low resolution 32 * 32 pixel thumbnail images can be transmitted by sending only 10 Kbytes. Sending an additional 15 Kbytes increases the resolution to 64 * 64 pixels, and so on. Other layering provide for progressive transmission and rendering based on quality, color component and spatial location in the image.

Dis-advantages:

- A lot of people are scared off by the term "lossy compression". But when it comes to real-world scenes,*no* digital image format can retain all the information that our eyeball can see.
- The real disadvantage of lossy compression is that if we repeatedly compress and decompress an image, we lose a little more quality each time.

The performance of these coders generally degrades at low bit rates mainly because of the underlying block-based DCT scheme. Because the block wise frequencies do not bear a simple relation to the frequencies achieved by just transforming the image into the Fourier (or frequency) domain.

IV. SOFTWARE IMPLEMENTATION

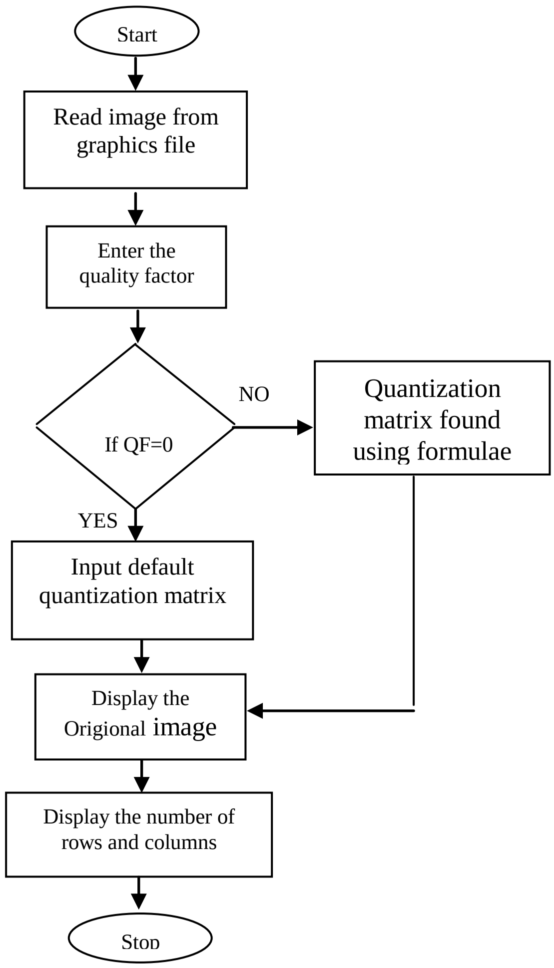


Fig 2: Flowchart for Start of Coding

The basic JPEG implementation involves following steps,

1. Color space conversion
2. Zero padding & Depadding
3. Level shifting & Inverse level shifting
4. Forward DCT & Inverse DCT
5. Quantization & Dequantization
6. Zig zag Coding & Inverse zig zag coding
7. DC difference coding & Inverse DC difference coding

8. Runlength coding & Runlength decoding
9. Entropy encoding & Entropy decoding

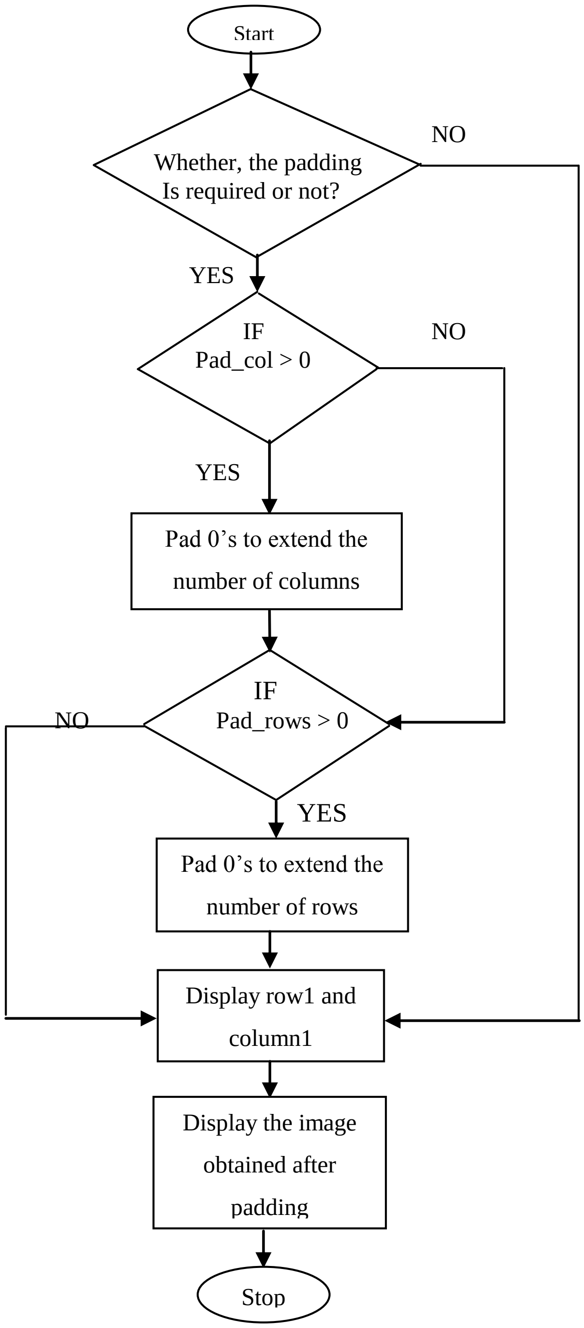


Fig 3: Flowchart for Padding

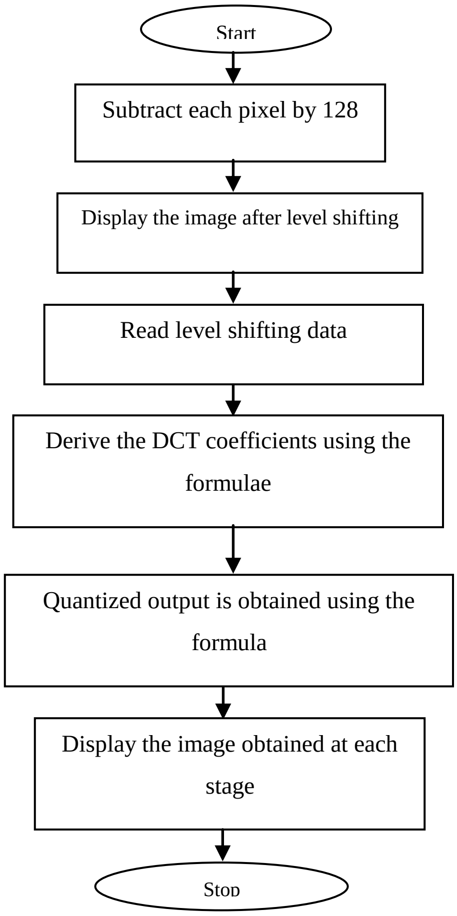


Fig 4: Flowchart Level Shifting, DCT & Quantization

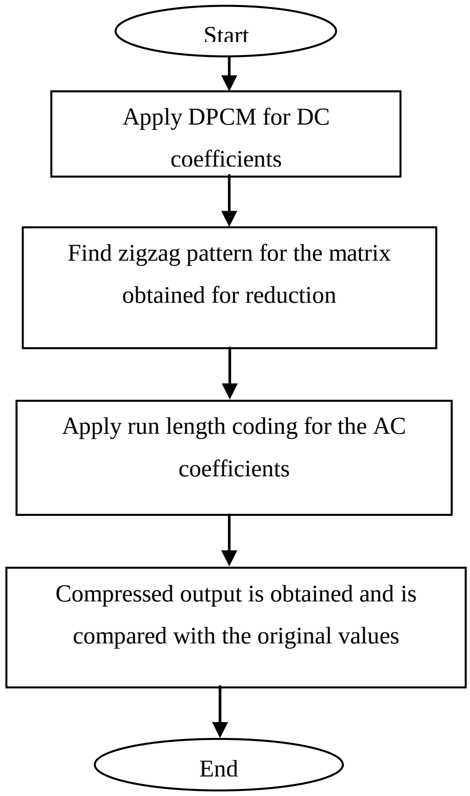


Fig5: Flowchart for DC Level Shifting, Zig-Zag and Run Length Encoding

The methodology involves design and implementation of both compression and decompression systems using JPEG algorithms. The algorithm consists of mainly two blocks, the

Encoder and the Decoder. The input to the encoder is an image in the BMP format. Initially the image is stored in the buffer and passed to encoder block. The simulation is done by using Code Composer Studio. The encoded data is the input for the decoder. The decoder decodes and the image is reconstructed. The compression performance achieved is dependent on quality factor.

The algorithm for decompression can be obtained by the inverse of the compression part i.e., just the reverse of the procedure followed for compression.

V. RESULTS & DISCUSSION

The results are taken by changing the Quality Factor. Quality Factor qf = 0 refers to the standard quantization matrix. The input images are of Windows BMP and TIFF format. The output images are of JPG format. The Error image presented here are the actual error multiplied by 8; this is done to get a proper view of the amount of error present.

Mathematical parameters such as Compression ratio, Compression performance, Square error, Mean Square Error (MSE), Root Mean Square Error (RMSE), and Peak Signal to Noise Ratio (PSNR) are determined for each of the tested image.

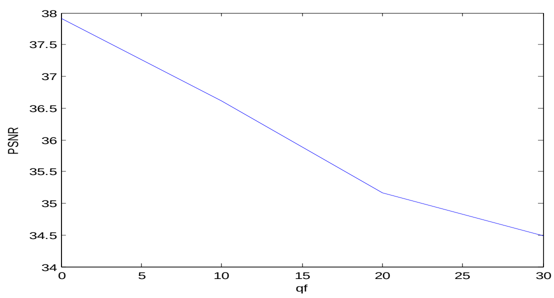


Fig. 6: qf vs PSNR for MRI image

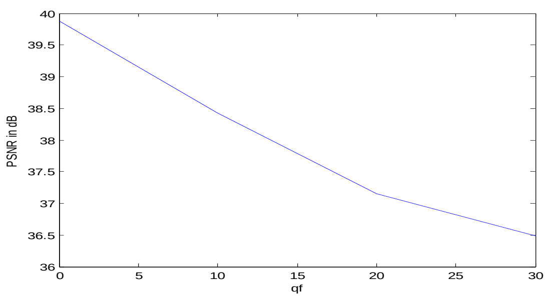


Fig. 7: qf vs PSNR for Cameraman image

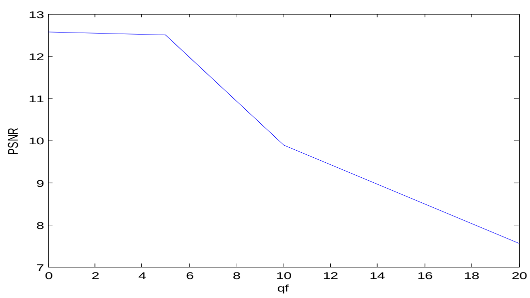


Fig. 8: qf vs PSNR for Lena image

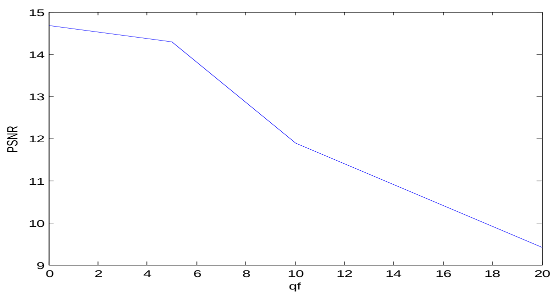
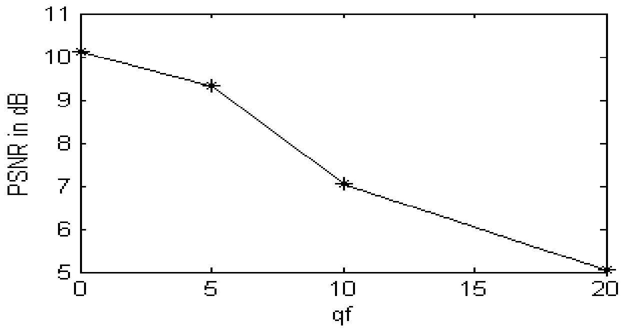


Fig. 9: qf vs PSNR for Einstein image

Table 2: Tabulated Results for Gray Level Images

Image (Gray level)	**Original Size (KB)**	**Quality Factor (qf)**	**Size of o/p (KB)**	**Compression Performance (%)**	**PSNR (dB)**
MRI 128 x 128	17.46	0	3.52	79.84	37.91
		10	3.12	82.14	36.61
		20	2.51	85.64	35.16
		30	2.18	87.5	34.49
Cameraman 256 x 256	65.24	0	9.67	85.18	39.89
		10	6.88	89.45	38.43
		20	4.82	92.6	37.15
		30	3.98	93.9	36.38
Einstein 256 x 256	65.1	0	9.30	85.71	14.6833
		5	8.40	87.09	14.2948
		10	6.23	90.43	11.8885
		20	4.40	93.24	9.4062
Lena 265 x 276	73.2	0	12.30	83.19	12.5733
		5	11.40	84.42	12.4953
		10	8.72	88.08	9.8896



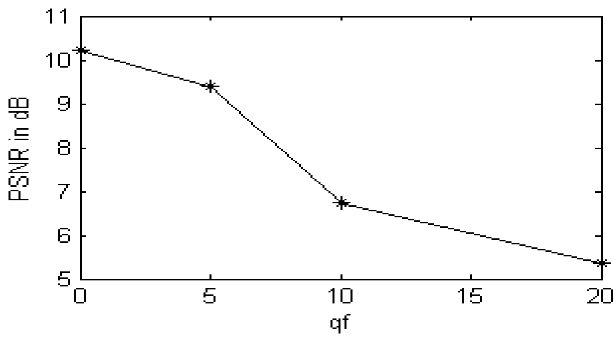


Fig. 10: qf vs PSNR for Aerial image

Fig. 11: qf vs PSNR for Stefigraph image

Table 3: Tabulated Results for Color Images

Image (Color, 24-bit)	Original Size (KB)	Quality Factor (qf)	Size of o/p (KB)	Compression Performance (%)	PSNR (dB)
Aerial 192 x 160	90.1	0	5.63	93.75	10.1288
		5	4.74	94.73	9.3281
		10	3.34	96.29	7.0564
		20	2.23	97.52	5.0629
Stefigraph 240 x 232	163.2	0	8.63	94.71	10.2389
		5	7.46	95.42	9.3900
		10	5.54	96.60	7.2892
		20	4.06	97.51	5.3598

From the above results it can be determined that:

- Color images get compressed to more extent than gray level images.
- The amount of redundancy varies from image to image. As a result the images of same input size may have different output size.

VI. CONCLUSION

- JPEG algorithm provides a better solution in reducing the volume of an image by keeping same number of rows and columns as that of original image. JPEG provides an opportunity for the user to customize the quality factor.
- The compression ratio achieved by RLC technique is in the range of 20 to 40 %. However, there are other compression algorithms that are recently developed which provide still further better compression ratios.
- Higher the quality factor more will be the compression but less will be the quality of output image and vice versa. Hence a tradeoff is necessary between image quality and compression. When we select a quality factor of 25 for all the images considered, compression provided is good and the image quality is also good.

REFERENCES

[1] Anil K Jain, *Fundamentals of Digital Image Processing*, PHI Eastern Economy Edition, 2003.
[2] Arthur R Weeks Jr, *Fundamentals of Electronic Image Processing,* PHI Eastern Economy Edition, 2005.
[3] Avtarsingh and S Srinivasan, *Digital Signal Processing Implementations using DSP Processors with Examples from TMS320C54xx*, Thomson Brooks/Cole, 2004.
[4] Bhaskaran V and Konstantinos K, *Image and Video Compression Standards: Algorithms and Architectures,* Kluwer, 1997.
[5] B Venkataramani and M Bhaskar, *Digital Signal Processors Architecture, Programming and Applications*, Tata McGraw-Hill, 2003.
[6] Code Composer Studio Getting Started Guide, Texas Instruments, 2004.
[7] Craig A Lindley, *Practical Image Processing in C*, John Wiley & Sons, 1991.
10. David Salomon, *DATA COMPRESSION: The complete reference,* Third edition, Springer International Edition, 2005.
[8] Feig E and E Linzer, *Discrete Cosine Transform Algorithms for Image Data Compression*, in Proceedings Electronic Imaging, 1990, pages 84-87.
[9] Golomb, Solomon W, *Run-Length Encodings*, IEEE Transactions on Information Theory, pp. 399-401.
[10] Held, Gilbert and Thomas R Marshall, *Data and Image Compression: Tools and Techniques*, 4th Edition, John Wiley, 1996.
[11] Huffman and David, *A Method for the Construction of Minimum Redundancy Codes*, Procedings of IEEE, pp.854 – 867, 1952.
[12] Iain E and G Richardson, *VIDEO CODEC DESIGN Developing Image and Video Compression Systems*, John Wiley & Sons Ltd, 2003.
[13] John C Russ, *The Image Processing Handbook*, CRC Press, Florida, 1992.
[14] Leger, Omachi and Wallace, *JPEG Still Picture Compression Algorithm*, Optical Engineering, Vol. 30, No. 7, pp. 947-954, 1991.
[15] Kenneth R. Castleman, *Digital Image Procesing*, PHI, Englewood Cliffs, 1979.
[16] Khalid Sayood, *Introduction to Data Compression,* Second Edition, Morgan Kaufmann, 2000.
[17] K R Rao and P C Yip, *The Transform and Data Compression Hand Book*, CRC press, 2001.
[18] Milan Sonaka, Vaclav Hlavac and Roger Boyle, *Image Processing Analysis and Machine Vision*, Second edition, PWS publishing, 1999.
[19] Netravali and J O Limb, *Picture Coding: A Preview*, Proceedings of the IEEE, pp 366-406, 1980.
[20] Rabbani, Majid, and Paul W Jones, *Digital Image Compression Techniques*, Spie Optical Engineering Press, 1991.
[21] Rafael C Gonzalez and Richard E Woods, *Digital Image Processing*, Second edition, Pearson Education, 2004.
[22] Rulph Chassaing, *DSP Applications Using C and the TMS320C6x DSK*, Wiley Interscience Publication, 2002.
[23] Sam Shanmugam, *Digital and Analog Communication Systems,* John Wiley & Sons Ltd, 1996.
[24] Stephen P Yanek, Quentin E Dolecek, Robert L Holland and Joan E Ferrer, *Fundamentals and Standards of Compression and Communication*, Academic Press© pp.759 – 770, 2000.
[25] TMS320C6711 Floating point Digital Signal Processor, SPRS088M, Texas Instruments, 2005.
11. Weidong Kou, *Digital Image Compression Algorithms and Standards*, First edition, Kluwer Academic Publishers, 1995.
[26] Yashavant Kanetkar, *Let Us C*, Third edition, BPB Publications, 1999.
[27] Ze-Nian Li and Mark S Drew, *Fundamentals of Multimedia,* Pearson Education, 2005.
[28] http://www.jpeg.org.

A. L. Choodarathnakara was born in Hassan District of Karnataka State in INDIA. He received the B.E. degree in Electronics & Communication Engineering in the year 2002 and M.Tech. degree in Digital Electronics & Communication Systems in the year 2008, both from the Malnad College of Engineering, Hassan affiliated to Visvesvaraya Technological University, Belgaum, INDIA.
He is an Assistant Professor of Electronics & Communication Engineering Department at Government Engineering College, Kushalnagar of South Kodagu District in Karnataka. His subjects of interest include Natural Language Processing, Language Learning Technologies, Text, Speech, Image, Signal & Remote Sensed Data Processing. Asst. Prof. Choodarathnakara A. L. is a Life Member of the Indian Society of Technical Education (ISTE-43168), Indian Society of Remote Sensing (ISRS-3533) and Indian Society of Geomatics (ISG-1258).

