JPEG IMAGE COMPRESSION USING VERILOG

B.Raghu Kanth[1], S R Sastry Kalavakolanu[2], M.Aravind Kumar[3], D.N.Bhushan Babu[4]

[1]*Asst.Professor, E.C.E Department, Andhra Loyola Institute of Engineering and Technology, Andhra Pradesh, India,* ***raghukanth8@hotmail.com***
[2] *Asst.Professor, E.C.E Department, Andhra Loyola Institute of Engineering and Technology, Andhra Pradesh, India,,* ***sastryece52@gmail.com***
[3] *Asst.Professor, E.C.E Department, Andhra Loyola Institute of Engineering and Technology, Andhra Pradesh, India,,* ***aravindkumarmtech@gmail.com***
[4] *Asst.Professor, E.C.E Department, Andhra Loyola Institute of Engineering and Technology, Andhra Pradesh, India,,* ***bhushan4me@gmail.com***

Abstract

Data compression is the reduction or elimination of redundancy in data representation in order to achieve savings in storage and communication costs. Data compression techniques can be broadly classified into two categories: Lossless, Lossy schemes. Digital images require an enormous amount of space for storage. The architecture exploits the principles of pipelining and parallelism to the maximum extent in order to obtain high speed the architecture for discrete cosine transforms and the entropy encoder are based on efficient algorithms designed for high speed VLSI. For example, a colour image with a resolution of 1024 x 1024 picture elements (pixels) with 24 bits per pixel would require 3.15 Mbytes in uncompressed form. Very high-speed design of efficient compression techniques will significantly help in meeting that challenge. The JPEG baseline algorithm consists mainly of two parts: (i) Discrete Cosine Transform (DCT) computation and (ii) Entropy encoding. The architecture for entropy encoding is based on a hardware algorithm designed to yield maximum clock speed

Index Terms: *Data Compression, Discrete Cosine Transform, Entropy Encoding, JPEG.*

--- *** ---

1. INTRODUCTION

Today we are talking about digital networks, digital representation of images, movies, video, TV, voice, digital library-all because digital representation of the signal is more robust than the Analog counterpart for processing, manipulation, storage, recovery, and transmission over long distances, even across the globe through communication networks. In recent years, there have been significant advancements in processing of still image, video, graphics, speech, and audio signals through digital computers in order to accomplish different application challenges. As a result, multimedia information comprising image, video, audio, speech, text, and other data types has the potential to become just another data type[1][2][3]. Still image and video data comprise a significant portion of the multimedia data and they occupy the lion's share of the communication bandwidth for multimedia communication. As a result, development of efficient image compression techniques continues to be an important challenge to us, both in academia and in industry.

In an abstract sense, we can describe "data compression" as a method that takes an input data D and generates a shorter representation of the data c(D) with a fewer number of bits compared to that of ***D.*** The reverse process is called "decompression", which takes the compressed data c(D) and generates or reconstructs the data D' as shown in Figure 1.1. Sometimes the compression (coding) and decompression (decoding) systems together are called a "CODEC," as shown in the broken box in Figure 1.1.

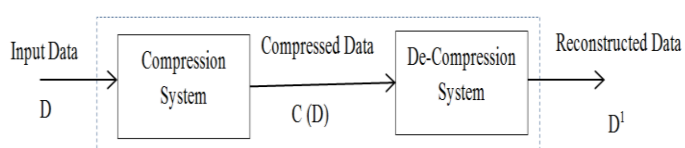

Fig-1: Codec

2. PREVIOUS WORK

Usually we need to apply lossless data compression techniques on text data or scientific data. For example, we cannot afford to compress the electronic copy of this text book using a lossy compression technique. It is expected that we shall reconstruct the same text after the decompression process. A small error in the reconstructed text can have a completely different meaning. We do not expect the sentence "You should not delete this file" in a text to change to "You should now delete this file" as a result of an error introduced by a lossy compression or decompression algorithm. Similarly, if we compress a huge ASCII file containing a program written in C language, for example, we expect to get back the same C code after decompression because of obvious reasons. The lossy compression techniques are usually applicable to data where high fidelity of reconstructed data is not required for perception by the human perceptual system. Examples of such types of data are image, video, graphics, speech, audio, etc. Some image compression applications may require the compression scheme to be lossless (i.e., each pixel of the decompressed image should be exactly identical to the original one). Medical imaging is an example of such an application where compressing digital radiographs with a lossy scheme could be a disaster if it has to make any compromises with the diagnostic accuracy [2][3].

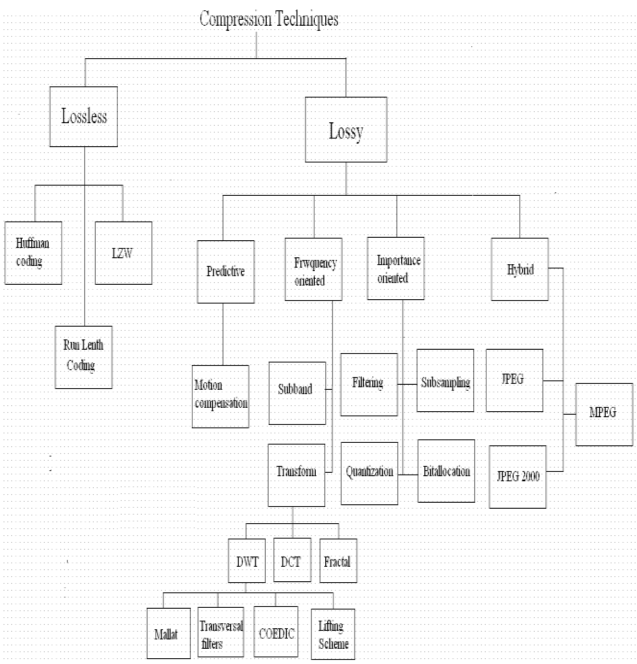

Fig-2: Classification of Compression Techniques

2.1 Image Compression:

Image compression is an important topic in the digital world, whether it can be commercial photography, industrial imagery, or video. A digital image bitmap can contain considerably large amounts of data causing exceptional overhead in both computational complexity as well as data processing. Compression is important to manage large amounts of data for network, Internet, or storage media.

Data compression itself is the process of reducing the amount of information into a smaller data set that can be used to represent, and reproduce the information. Types of image compression include loss less compression, and lossy compression techniques that are used to meet the needs of specific applications. JPEG compression can be used as a loss less or a lossy process depending on the requirements of the application both lossless and lossy compression techniques employ reduction of redundant data [5][6].

The Joint Photographic Experts Group produced the well-known image format JPEG, a widely used image format. JPEG provides solid baseline compression algorithm that can be modified numerous ways to fit any desired application. The JPEG specification was released initially in 1991, although it does not specify a particular implementation.

2.2 Redundancy Coding

To compress data, it is important to recognize redundancies in data, in the form of coding redundancy, inter-pixel redundancy, and psycho-visual redundancy. Data redundancies occur when unnecessary data is used to represent source information. Compression is achieved when one or more of these types of redundancies are reduced. Intuitively, removing unnecessary data will decrease the size of the data, without losing any important information. However, this is not the case for psycho-visual redundancy [8].

The most obvious way to reduce compression is to reduce the coding redundancy. This is referring to the entropy of an image in the sense that more details used than necessary to convey the information. Loss less redundancy removal compression techniques are classified as entropy coding. Other compression can be obtained through inter-pixel redundancy removal. Each adjacent pixel is highly related to its neighbors, thus can be differentially encoded rather than sending the entire value of the pixel. Similarly adjacent blocks have the same property, although not too the extent of pixels.

In order to produce error free compression, it is recommended that only coding redundancy is reduced or eliminated. This means that the source image will be exactly the same as the decompressed image. However, inter-pixel redundancies can also be removed, as the exact pixel value can be reconstructed from differential coding or through run length coding [7].

Images contain both low frequency and high frequency components. Low frequencies correspond to slowly varying colour, whereas high frequencies represent the detail within the image. Intuitively, low frequencies are more important to create a good representation of an image. Higher frequencies can largely be ignored to a certain degree. The human eye is more sensitive to the luminance (brightness), than the chrominance (colour difference) of an image. Thus during compression, chrominance values are less important and quantization can be used to reduce the amount of psycho-visual redundancy [8]. Luminance data can be quantized, but more coarsely to ensure that important data is not lost. Several compression algorithms use transforms to change the image from pixel values representing color to frequencies dealing with light and dark of an image, not frequencies of light. Many forms of the JPEG compressions algorithm make use of the discrete cosine transforms.

Most multimedia systems combine both transform coding and entropy coding into a hybrid coding technique.

2.3 Lossy Compression:

Loss less compression techniques work by removing redundant information as well as removing or reducing information that can be recreated during decompression. Loss less compression is ideal, as source data will be recreated without error. However, this leads to small compression ratios and will most likely not meet the needs of many applications. Compression ratios are highly dependent on input data, thus loss less compression will not meet the requirements of applications requiring a constant data rate or data size. Loss less techniques employs entropy encoders such as Huffman encoders. Huffman produced an efficient variable length-coding scheme. Loss less compression techniques work by removing redundant information as well as removing or reducing information that can be recreated during decompression. Loss less compression is ideal, as source data will be recreated without error. However, this leads to small compression ratios and will most likely not meet the needs of many applications. Compression ratios are highly dependent on input data, thus loss less compression will not meet the requirements of applications requiring a constant data rate or data size. Loss less techniques employs entropy encoders such as Huffman encoders. Huffman produced an efficient variable length-coding scheme [2][3].

Lossy compression algorithms attempt to maximize the benefits of compression ratio and bit rate, while minimizing loss of quality. Finding optimal ways to reach this goal is a severely complicated process as many factors must be taken into account. Variables such as a quality factor which is used to scale the quantization tables used in JPEG can either reduce the resulting image quality with higher compression ratios, or conversely improving image quality with lower compression ratios. JPEG, although lossy, can give higher compression ratios than GIF while leaving the human observer with little to complain of loss in quality [3]. Compression induced loss in images can cause both missing image features as well as artifacts added to the picture. Artifacts are caused by noise produced by sources such as quantization, and may show up as blocking within the image. Blocking artifacts within an image can become apparent with higher compression ratios. The edges are representative of blocks used during compression, such as 8x8 blocks of pixels, used in JPEG.

3. EXPERIMENTAL SET UP

3.1 Discrete Cosine Transform (DCT):

The discrete cosine transform is the basis for the JPEG compression standard. For JPEG, this allows for efficient compression by allowing quantization on elements that is less sensitive. The DCT algorithm is completely reversible making this useful for both loss less and lossy compression techniques.

The DCT is a special case of the well-known Fourier transform. Essentially the Fourier transform in theory can represent a given input signal with a series of sine and cosine terms. The discrete cosine transform is a special case of the Fourier transform in which the sine components are eliminated. For JPEG, a two-dimensional DCT algorithm is used which is essentially the one-dimensional version evaluated twice. By this property there are numerous ways to efficiently implement the software or hardware based DCT module. The DCT is operated two dimensionally taking into account 8 by 8 blocks of pixels. The resulting data set is an 8 by 8 block of frequency space components, the coefficients scaling the series cosine terms, known as basic functions. The First element at row 0 and column 0, is known as the DC term, the average frequency value of the entire block. The other 63 terms are AC components, which represent the spatial frequencies that compose the input pixel block, by scaling the cosine terms within the series. There are two useful products of the DCT algorithm. First it has the ability to concentrate image energy into a small number of coefficients. Second, it minimizes the interdependencies between coefficients. These two points essentially state why this form of transform is used for the standard JPEG compression technique. By compacting the energy within an image, more coefficients are left to be quantized coarsely, impacting compression positively, but not losing quality in the resulting image after decompression. Taking away inter-pixel relations allows quantization to be non-linear, also affecting quantization positively [5]. DCT has been effective in producing great pictures at low bit rates and is fairly easy to implement with fast hardware based algorithms. An orthogonal transform such as the DCT has the good property that the inverse DCT can take its frequency coefficients back to the spatial domain at no loss. However, implementations can be lossy due to bit limitations and especially apparent in those algorithms in hardware. The DCT does win in terms of computational complexity as there are numerous studies that have been completed in different techniques for evaluating the DCT.

The discrete cosine transform is actually more efficient in reconstructing a given number of samples, as compared to a Fourier transform. By using the property of orthogonally of cosine, as opposed to sine, a signal can be periodically reconstructed based on a fewer number of samples. Any sine based transform is not orthogonal, and would have to take Fourier transforms of more numbers of samples to approximate a sequence of samples as a periodic signal. As the signal we are sampling, the given image, there is actually no real periodicity [2][3][5]. If the image is run through a Fourier transform, the sine terms can actually incur large changes in amplitude for the signal, due to sine not being orthogonal. DCT will avoid this by not carrying this information to represent the changes. In the case of JPEG, a two-dimensional DCT is used, which correlates the image with 64 basis functions.

3.2 Procedure for doing the DCT on an 8x8 Block:

Before we begin, it should be noted that the pixel values of a black-and-white image range from 0 to 255 in steps of 1, where 0 and pure white by represent pure black 255. Thus it can be seen how a photo, illustration, etc. can be accurately represented by these256 shades of gray. Since an image comprises hundreds or even thousands of 8x8 blocks of pixels, the following description of what happens to one 8x8 block is a microcosm of the JPEG process what is done to one block of image pixels is done to all of them, in the order earlier specified. Now, let‘s start with a block of image-pixel values. This particular block was chosen from the very upper- left-hand corner of an image.

We are now ready to perform the Discrete Cosine Transform, which is accomplished by matrix multiplication

$$D = T\,M\,T$$

In matrix M is first multiplied on the left by the DCT matrix T which is the coefficient matrix; this transforms the rows. The columns are then transformed by multiplying on the right by the transpose of the DCT matrix.

3.2.1 Quantization:

Our 8x8 block of DCT coefficients is now ready for compression by quantization. A remarkable and highly useful feature of the JPEG process is that in this step, varying levels of image compression and quality are obtainable through selection of specific quantization matrices. This enables the user to decide on quality levels ranging from 1 to 100, where 1 gives the poorest image quality and highest compression, while 100 gives the best quality and lowest compression. As a result, the quality/compression ratio can be tailored to suit different needs. Subjective experiments involving the human visual system have resulted in the JPEG standard quantization matrix. With a quality level of 50, this matrix renders both high compression and excellent decompressed image quality[6][7].

Quantization is achieved by dividing each element in the transformed image matrix D by the corresponding element in the quantization matrix, and then rounding to the nearest integer value. For the following step, quantization matrix Q50 is used.

Recall that the coefficients situated near the upper-left corner correspond to the lower Frequencies œ to which the human eye is most sensitive œ of the image block. In addition, the zeros represent the less important, higher frequencies that have been discarded, giving rise to lossy part of compression. As mentioned earlier only the non-zero components are used for reconstruction of the image. The number of zeros given by each Quantization matrices varies.

3.2.2 Encoder:

The quantized matrix is now ready for the final step of compression. The entire matrix coefficients are coded into the binary format by the Encoder. After quantization it is quite common that maximum of the coefficients are equal to zeros. JPEG takes the advantage of encoding quantized coefficients in zigzag order. Entropy coding is a special form of lossless data compression. It involves arranging the image components in a "zigzag" order employing run-length encoding (RLE) algorithm that groups similar frequencies together, inserting length coding zeros, and then using Huffman coding on what is left[4][8].

The JPEG standard also allows, but does not require decoders to support, the use of arithmetic coding, which is mathematically superior to Huffman coding. However, this feature has rarely been used as it was historically covered by patents requiring royalty-bearing licenses, and because it is slower to encode and decode compared to Huffman coding.

Arithmetic coding typically makes files about 5-7% smaller. The previous quantized DC coefficient is used to predict the current quantized DC coefficient. The difference between the two is encoded rather than the actual value. The encoding of the 63 quantized AC coefficients does not use such prediction differencing.

4. SIMULATION RESULTS

The test bench is developed in order to test the modelled design. This developed test bench will automatically force the inputs and will make the operations of algorithm to perform.

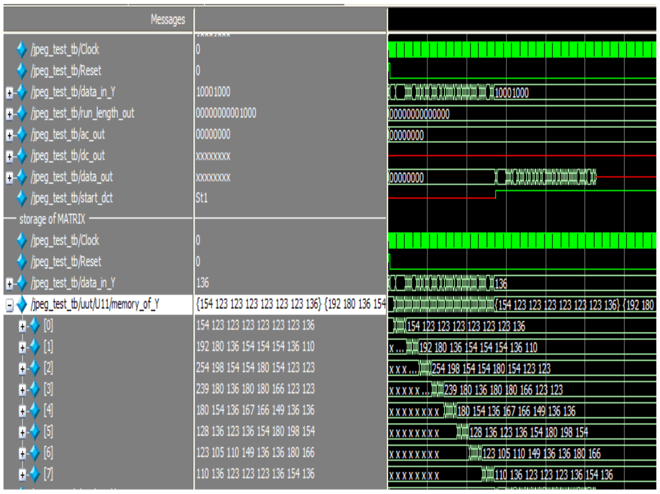

Fig-3: Storage of all pixcel values.

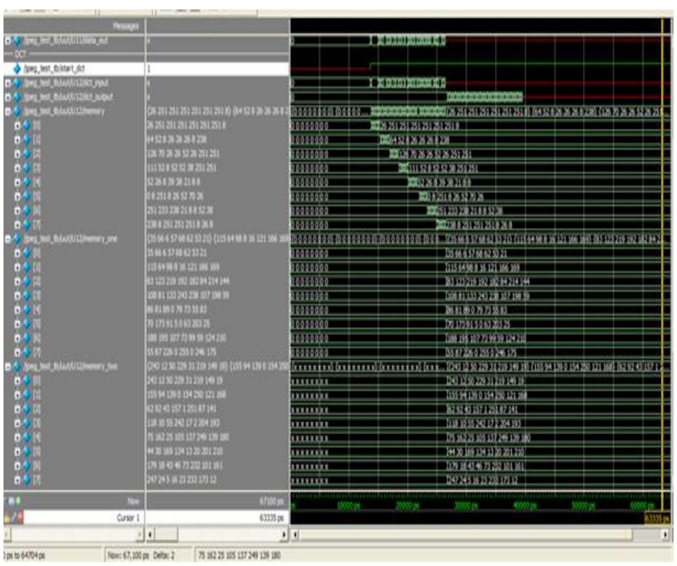

Fig-4: Descrete Cosine Transform.

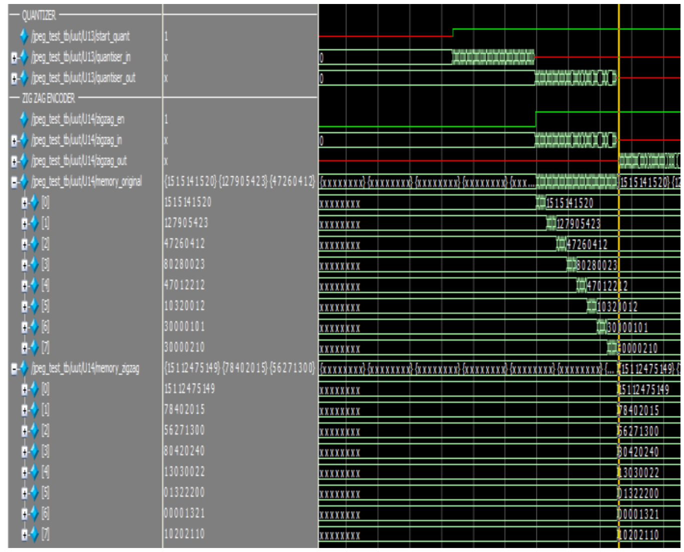

Fig-5: Quantization and Encoding

5. CONCLUSION

The emerging JPEG continuous-tone image compression standard is not a panacea that will solve the myriad issues which must be addressed before digital images will be fully integrated within all the applications that will ultimately benefit from them. For example, if two applications cannot exchange uncompressed images because they use incompatible color spaces, aspect ratios, dimensions, etc. then a common compression method will not help. For these applications, the thorough technical evaluation, testing, selection, validation, and documentation work which JPEG committee members have performed is expected to soon yield an approved international standard that will withstand the tests of quality and time. As diverse imaging applications become increasingly implemented on open networked computing systems, the ultimate measure of the committee's success will be when JPEG-compressed digital images come to be regarded and even taken for granted as "just another data type," as text and graphics are today.

6. REFERENCES

[1]. Ahmed, N., T. Natarajan, and K. R. Rao. 1974. On image processing and a discrete cosine transform. IEEE Transactions on Computers C-23(1): 90–93.

[2]. Chen, W. H., and W. K. Pratt. 1984. Scene adaptive coder. IEEE Transactions on Communications COM-32: 225–232.

[3]. Jain, A. K. 1989. Fundamentals of digital image processing. Prentice Hall: Englewood Cliffs, NJ.

[4]. Puri, A. 1992. Video coding using the MPEG-1 compression standard. Society for Information Display Digest of Technical Papers 23: 123–126.
[5]. Shannon, C. E., and W. Weaver. 1949. The mathematical theory of communication. Urbana: University of llinois Press.
[6]. Wallace, G. 1991. The JPEG still picture compression standard. Communications of the ACM 34(4): 30–44.
[7]. Watson, A. B. 1993. DCT quantization matrices visually optimized for individual images. Proceedings of the SPIE 1913:202–216. (Human Vision, Visual Processing, and Digital Display IV. Rogowitz ed. SPIE. Bellingham,WA).
[8]. The International Telegraph and Telephone Consultative Committee (CCITT). "Information Technology – Digital Compression and Coding of Continuous-Tone Still Images – Requirements and Guidelines". Rec. T.81, 1992.
[9]. W. Pennebaker, J. Mitchell. JPEG Still Image Data Compression Standard, Van Nostrand Reinhold, USA, 1992.
[10]. "Home site of the JPEG and JBIG committees" <http://www.jpeg.org/> (21/04/01).
[11]. V. Bhaskaran, K. Konstantinides. Image and Video Compression Standards Algorithms and Architectures – Second Edition, Kluwer Academic Publishers, USA, 1999.
[12]. J. Miano. Compressed Image File Formats – JPEG, PNG, GIF, XBM, BMP, Addison Wesley Longman Inc, USA, 1999.
[13]. W. Chen, C. Smith, S. Fralick. "A Fast Computational Algorithm for the Discrete Cosine Transform". IEEE Transactions on Communications, v. COM-25, n. 9, p. 1004-1009, 1977.
[14]. Y. Arai, T. Agui, M. Nakajima. "A Fast DCT-SQ Scheme for Images". Transactions of IEICE, vol. E71, n°. 11, 1988, pp. 1095-1097.
[15]. E. Feig, S. Winograd. "Fast Algorithms for the Discrete Cosine Transform". IEEE Transactions on Signal Processing, v. 40, n. 9, p. 2174-2193, 1992.
[16]. M. Kovac, N. Ranganathan. "JAGUAR: A Fully Pipeline VLSI Architecture for JPEG Image Compression Standard". Proceedings of the IEEE, vol. 83, n°. 2, 1995, pp. 247-258.
[17]. A. Madisetti, A. Willson Jr. "A 100 MHz 2-D 8 x 8 DCT/IDCT Processor for HDTV Applications". IEE Transactions on Circuits. and Systems for Video Technology, v.5, n.2, p.158-165, Apr. 1995.
[18]. C. Wang, C. Chen. "High-Throughput VLSI Architectures for the 1-D and 2-D Discrete Cosine Transforms". IEE Transactions on Circuits and Systems for Video Technology, v.5, n.1, p.31-40, Feb. 1995.
[19]. Y. Lee, T. Chen, L. Chen., M. Chen, C. Ku. "A Cost-Effective Architecture for 8 x 8 Two-Dimensional DCT/IDCT Using Direct Method". IEEE Transactions on Circuits and Systems for Video Technology, v.7, n.3, p.459-467, Jun. 1997.

B. Raghu Kanth was born at Vijayawada Andhra Pradesh. He obtained his Bachelor`s degree from JNTU, HYD in ECE Department in 2007. He has completed his M.Tech degree in VLSI from K.L.University at Vijayawada. He has more than two and half years of teaching experience and having seven International Journal publishing. His area of interest is in Analog VLSI, Low-power VLSI and Testing