



JPEG Algorithm Implementation Using ARM Processor

Mahavir Kasar , Y.D. Chincholkar

E&TC Dept., SCOE, PUNE

India

Abstract- In this paper it is illustrated an image encoder is designed using an embedded platform more specifically a JPEG decoder using ARM7TDMI processor. Here gray scale image is used and it is coded by using MATLAB software for comparing the results with standard one and results obtained using Keil software. Successfully putting a new application of JPEG on ARM7 processor.

Keywords- JPEG, DCT, IDCT, LZW, ARM, RLE

I. Introduction

Evolution of embedded systems recently has a strong trend towards application-specific, single-chip solutions. In the domain of embedded system the ARM processor core is a leading RISC processor architecture. It supports a unique feature of code size reduction. Image Compression is a method, which reduces the size of the data to reduce the amount of space required to store the data. The Discrete cosine transform (DCT) is a method for transforms a signal or image from spatial domain to frequency component. It is a widely used technique in image compression. In this paper, we are proposing a lossless discrete cosine transform (DCT) compression technique for two-dimensional images. In the several scenarios, the utilization of the proposed technique for image compression resulted in comparable or better performance, when compared to the different modes of the lossless JPEG standard.

II. ARM Processor

An embedded environment constrains the energy consumption, the memory space, and the execution time of an application. Real-time embedded systems are a challenge to implement. On the other hand, the complexity of the applications running on an embedded platform keeps increasing. Examples include next-generation cell phones, portable game-boys and digital assistants, smartcards, and medical monitoring devices. Looking forward to new era of image processing, modernized and fastest ways to compress the images by using JPEG, in the world of photography and its best outputs in terms of image compression, world is definitely in need of new approaches. To achieve this we need to evaluate the performance of ARM PROCESSOR by using JPEG as one of the application [3]. This is an embedded approach for the compression which can be useful and suitable for the application like digital camera. Now a day's Digital camera has become fastest and best means in the world of photography, and so as the images created by it. Fig 1 shows generalized block diagram of ARM based JPEG encoder [11]. As any Images file created by digital is to be compressed by using JPEG standard and then further this file will be processed through serial communication by using ARM processor. JPEG acts as a JPEG decoder for the ARM processor. In this paper, evaluating the performance of ARM processor family with JPEG. Here, image file is taken from windows and then converted into grayscale image by using MATLAB then is process through ARM.

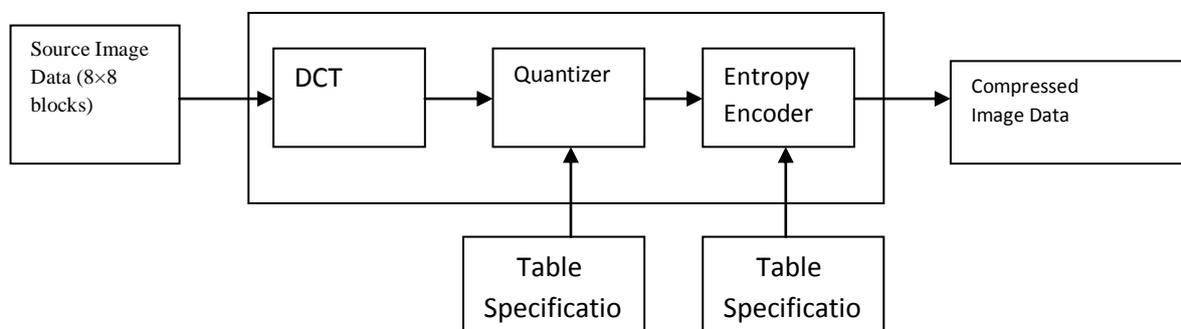


Fig. 1 DCT based JPEG Encoder.

Compression- Compression is a method, which is reducing the size of the data to reduce the amount of space required to store the data. Compression may be in the form of Data compression, Image compression, Audio data compression, and video compression and Bandwidth compression in Telecommunications. Compression categorized in two broad ways:

1) **Lossless Compression:** It never removes any information from the original image. These are referred to as bit-preserving or reversible compression also lossless compression frequently involves some form of entropy encoding and that are based in information theoretic techniques. The following algorithms are lossless.

- 1) Huffman compression 2) LZW compression 3) RLE compression

2) **Lossy Compression:** It creates smaller files by discarding access image data from the original image. Video and Audio compression techniques are most suited to this form of compression. The following algorithm is Lossy-JPEG compression. The advantage of lossy methods over lossless methods is that in some cases, a lossy method can produce a much smaller compressed file than any known lossless method.

III. Some Basic Compression Methods

Joint Photography Expert Groups: JPEG's proposed standard aims to be generic, to support a wide variety of applications for continuous tone images. The DCT is usually applied to reduce spatial redundancy in order to achieve good compression performance. Some of the applications of two-dimensional DCT technique involve image compression and compression of individual video frames. The JPEG process is a widely used form of lossy image compression that centers on the Discrete Cosine Transform. DCT is also useful for transferring multidimensional data from spatial domain to frequency domain to decorrelate pixels, where different operations, like spread spectrum, data compression, data watermarking can be performed in performed manner [1]. The JPEG is used for both color and black and-white images. To meet the differing needs of many applications, the JPEG standard includes two basic compression methods, each with various modes of operation. A DCT based method is specified for "lossy" compression, and a predictive method for "lossless" compression. JPEG features a simple lossy technique known as the Baseline method, a subset of the other DCT based modes of operation. The Baseline method has been by far the most widely implemented JPEG method to date, and is sufficient in its own right for a large number of applications. JPEG has following modes of operation:

- a) Sequential encoding: each image component is encoded in a single left to right top to bottom scan.
- b) Progressive encoding: the image is encoded in multiple scans for applications in which transmission time is long, and the viewer prefers to watch the image build up in multiple coarse to clear passes.
- c) Lossless encoding: the image is encoded to guarantee exact recovery of every source image sample value (even though the result is low compression compared to the lossy modes).
- d) Hierarchical encoding: the image is encoded at multiple resolutions so that lower resolution versions may be accessed without first having to decompress the image at its full resolution.

The JPEG compression: Joint Photographic Expert Group (JPEG) which is commonly used method of compression for photographic images. JPEG compression can be used in a variety of file formats:

- 1) EPS-files 2) EPS DCS-files 3) JFIF-files 4) PDF-files

The process may be acquired as such given under:

- 1. The image first is broken into 8x8 blocks of pixels.
- 2. The DCT is applied to each block, it is working from left to right, top to bottom.
- 3. Each block is compressed using quantization table.
- 4. The array of compressed blocks that comprise the image is stored in a drastically reduced amount of space.
- 5. When desired, the image is reconstructed through decompression, known as a process that uses the Inverse Discrete Cosine Transform (IDCT).

Fig.1 shows the key processing steps which are the heart of the DCT Based modes of operation [7]. These figures illustrate the special case of Single component (grayscale) image compression. Color image compression can then be approximately regarded as compression of multiple grayscale images, which are either compressed entirely one at a time, or are compressed by alternately interleaving 8x8 sample blocks from each in turn. For DCT sequential mode codec's, which include the Baseline sequential codec, the simplified diagrams indicate how, single component compression works in a fairly complete way. Each 8x8 block is input, makes its way through each processing step, and yields output in compressed form into the data stream [8]. DCT progressive mode codec's, an image buffer exists prior to the entropy coding step, so that an image can be stored and then parceled out in multiple scans with successively improving quality. For the hierarchical mode of operation, the steps shown are used as building blocks within a larger framework. The DCT coefficient values can thus be regarded as the relative amount of the 2D spatial frequencies contained in the 64point input signal. The coefficient with zero frequency in both dimensions is called the "DC coefficient" and the remaining 63 coefficients are called as "AC coefficient". Each of the 64 coefficients is then quantized using one of the 64 corresponding values from a quantization table. After quantization, the DC coefficient and the 63 AC coefficients are prepared for entropy encoding as shown in fig 2.

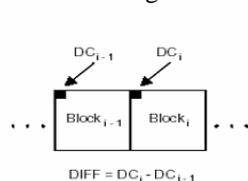


FIG.2 DIFFERENTIAL DC ENCODING

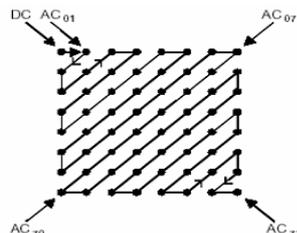


FIG.3 ZIG-ZAG SEQUENCE

The previous quantized DC coefficient is used to predict the current quantized DC coefficient, and the difference is encoded. The 63 quantized AC coefficients undergo no such differential encoding, but are converted into a one-dimensional zig-zag sequence, as shown in fig 3.

Preparation of Quantized Coefficients for Entropy encoding

The quantized coefficients are then passed to an entropy encoding procedure that compresses the data further. One of two entropy coding procedures can be used. If Huffman encoding is used, Huffman table specifications must be provided to the encoder. If arithmetic encoding is used, arithmetic coding conditioning table specifications may be provided; otherwise the default conditioning table specifications shall be used. It uses Huffman encoding as mentioned previously [10].

Run-Length Encoding (RLE): RLE stands for Run Length Encoding. It is a lossless algorithm that only furnishes decent compression ratios in specific types of data. It is a form of data compression in which the same data value occurs in many consecutive data elements (known as *Runs*) are stored as a single data value and count. This is most useful on data that contains many such runs: for example, simple graphic images such as icons, line drawings, and animations. It may increase the file size because, that doesn't have many runs, and is not useful with files. RLE compression can be used in the following file formats:

- 1) TIFF files
- 2) PDF files

Huffman Coding: The Huffman compression algorithm is invented by David Huffman, formerly a professor at MIT. Huffman compression is a lossless compression algorithm that is apotheosis for compressing text or program files. This credibly explains why it is used a lot in compression programs like ZIP or ARJ. Huffman encoding can be further optimized in two different ways:

- Adaptive Huffman code dynamically changes the code words concordant to the change of probabilities of the symbols.
- Extended Huffman compression can encode groups of symbols rather than single symbols [12].

IV. Proposed Methodology

The Discrete Cosine Transform: DCT Attempts to decorrelate the image data after decorrelation each transform coefficient can be encoded without dropping off compression efficiency. The DCT and some of its important properties.

The One-Dimensional DCT: The DCT of a list of n real numbers $s(x)$, where $x=0, 1, \dots, n-1$, is the list of Length n given by:

$$D(i, j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x)p(y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right] \quad \text{For } u=0, 1, 2, N-1$$

Similarly, the inverse transform is defined as- $f(x) = \sum_{u=0}^{N-1} \alpha(u) C(u) \cos\left[\frac{\pi(2x+1)u}{2N}\right]$

Thus, the first transform coefficient is the coefficient is the average value of the sample sequence.

The Two-Dimensional DCT: The Discrete Cosine Transform (DCT) is one of many transforms that takes its input and transforms it into a linear combination of weighted basis functions. These basis functions are commonly the frequency. The 2-D Discrete Cosine Transform is just a one dimensional DCT applied twice, once in the x direction, and again in the y direction. One can imagine the computational complexity of doing so for a large image. Thus, many algorithms, such as the Fast Fourier Transform (FFT), have been created to speed the computation. The DCT computes the I, j th entry of the DCT of an image.

$$D(i, j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x, y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right]$$

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u = 0 \\ 1 & \text{if } u > 0 \end{cases}$$

N is the size of the block that the DCT is applied on. The equation calculates one entry (i, j) of the transformed image from the pixel values of the original image matrix. For the standard 8×8 block that JPEG compression uses, N equals 8 and x and y range from 0 to 7 . Therefore $D(i, j)$ would be as in equation:

$$D(i, j) = \frac{1}{4} C(i)C(j) \sum_{x=0}^7 \sum_{y=0}^7 p(x, y) \cos\left[\frac{(2x+1)i\pi}{16}\right] \cos\left[\frac{(2y+1)j\pi}{16}\right]$$

Because the DCT uses cosine functions, the resulting matrix depends on the horizontal and vertical frequencies. Therefore an image block with a lot of change in has a very random looking resulting matrix of a large value for the first element and zeroes for the other element.

V. Experimental Result

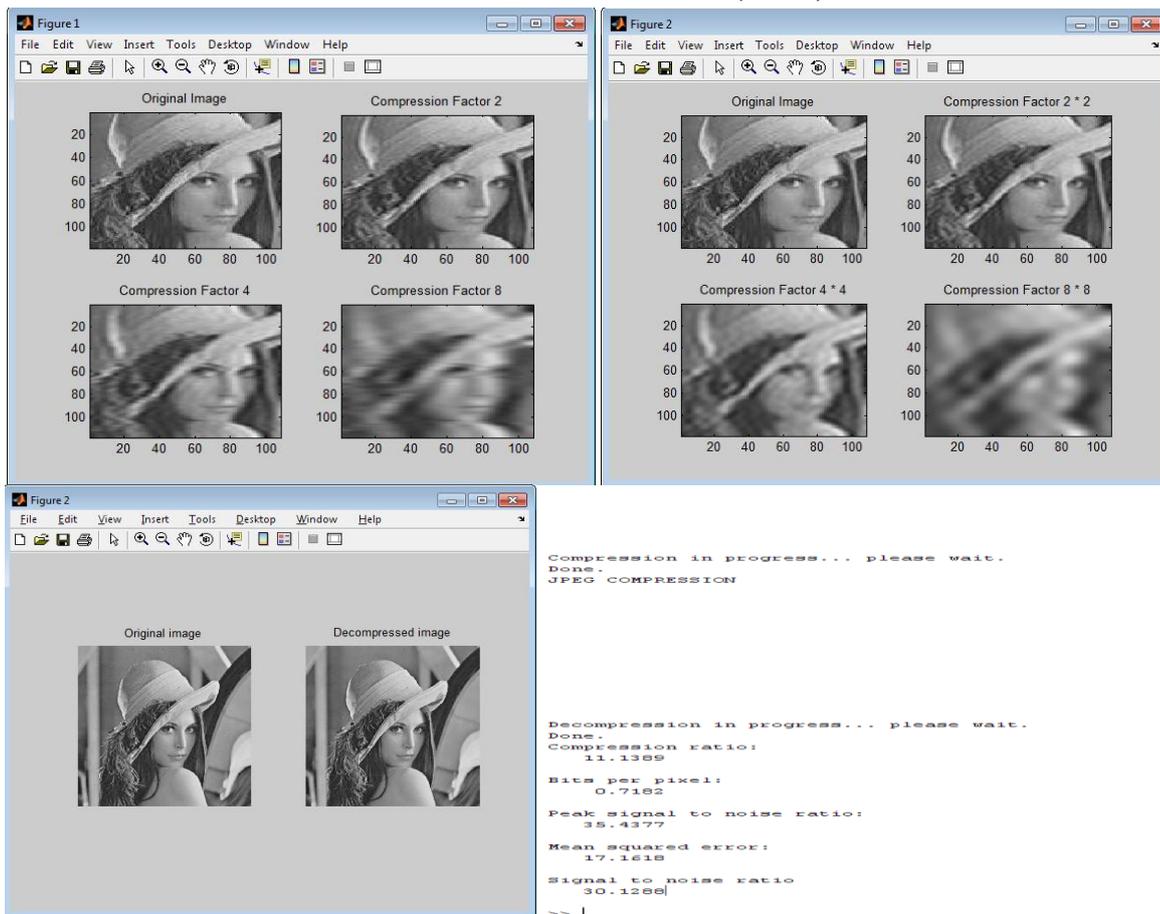
To evaluate the performance of the proposed scheme, 2-D DCT is applied on Lena's image (352×352) as a test image. First DCT applied to rows only through compression factor 2, compression factor 4 and compression factor 8, and observe the result. After that, DCT applied to rows*column. The results are then compared with various compression

methods. We used Peak Signal-to Noise Ratio (PSNR) and Mean Square Error (MSE) for a compressed image. This ratio is often used as a quality measurement between the original and compressed image.

$$MSE = \frac{1}{nm} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - I'(i, j)]^2$$

Where $x(m, n)$ and $y(m, n)$ are the two images of the size $m \times n$. In this case x is the original image and y is the compressed image. To compute the PSNR, first calculate the mean-squared error using the following equation:

$$PSNR(db) = 10 \log \left(\frac{255^2}{MSE} \right)$$



VI. Conclusion

In this paper, there are 256 possible shades of gray in a black and white picture, and a difference of say 8 is hardly differentiated by the human eye. DCT takes advantage of redundancies of the data by grouping pixels with similar frequencies. Thus we can also conclude that the difference between original and decompressed image goes on decreasing as there is an increase in image resolution at the same compression ratio. This image compression scheme for images has been presented based on the DCT. The anticipated results obtained relevant reconstructed image quality as well as preservation of significant image details, while on the other hand accomplishing high compression rates. High compression ratio and better image quality accomplished which is better than existing methods. This paper has concentrated on development of efficient and effective algorithm for still image compression. Fast and lossless compression algorithm using DCT is developed. Results show that reduction in encoding time with little degradation in image quality compare to subsisting method. Compression ratio is also increased, while comparing the proposed method with other methods. Our future work involves improving image quality by increasing PSNR value and lowering MSE value.

References

- [1] G.K. Wallace, "The JPEG still picture compression standard", IEEE Transactions on Consumer Electronics, 34 (4):30-44, 1991.

- [2] Chiang-Lung Liu, "A JPEG-Compliant Method for Image Authentication and Recovery", Journal of Information Assurance and Security 4 (2009) 622-628.
- [3] Andrew N. SLOSS, Dominic SYMES, Chris WRIGHT Design and optimizing system Software" "ARM System Developer's Guide,".
- [4] Jason Andrews,"ARM SoC Verification Matrix Improves HW/SW Co-Verification", Verisity Design, Inc., Mountain View, USA
- [5] Asadollah Shahbahrami, Ramin Bahrapour, MobinS abbaghi Rostami, Mostafa Ayoubi Mobarhan, "Evaluation of Huffman and Arithmetic Algorithms for Multimedia Compression Standards", International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.4, August 2011.
- [6] Tumeo, A.; Monchiero, M.; Palermo, G.; Ferrandi, F.; Sciuto, "A Self-Reconfigurable Implementation of the JPEG Encoder ", Application -specific Systems, Architectures and Processors, 2007. ASAP. IEEE International Conf. on 9-11 July 2007
- [7] Ken Cabeen and Peter Gent," Image Compression and Discrete Cosine Transform," math 45, college of Redwoods.
- [8] Syed Ali Khayam,"The Discrete Cosine Transform (DCT): Theroy and Application /"Department of Electrical & Computer Engineering, Michigan State University, March 2003
- [9] C.Saravanan, R. Ponalagusamy,"Lossless Grey-scale Image Compression using Source Symbols Reduction and Huffman Coding", International Journal of Image Processing (IJIP), Volume (3):Issue5
- [10] Information technology," Digital Compression and coding of continuous tone still images", requirements and guidelines, CCITT rec ITU T.81.
- [11] Chang-An Tsai, Yu-Te Chou, Yu-Tsang Chang, Lan-Da Van, and Chun-Ming Huang, "ARM-Based SoC Prototyping Platform Using Aptix", Exploring Innovation in Education and Research, ©iCEER-2005
- [12] Huffman, D.A., A method for the construction of minimum-redundancy codes. Proc. Radio Eng. 40(9), pp.1098-1101, 1952.