Implementation of Image Compression and Decompression using JPEG Technique

Mr. Akash Kumar Gupta
Assistant Professor
Department of Electronics & Communication Engineering
Birla Institute of Technology, Mesra, Off Campus, Deoghar

Girija Shankar Sahu
UG Student
Department of Electronics & Communication Engineering
Birla Institute of Technology, Mesra, Off Campus, Deoghar

Ritesh Kumar
UG Student
Department of Electronics & Communication Engineering
Birla Institute of Technology, Mesra, Off Campus, Deoghar

Gaurav Navin Kindo
UG Student
Department of Electronics & Communication Engineering
Birla Institute of Technology, Mesra, Off Campus, Deoghar

Abstract

Even though the storage media has increase manifold in terms of size, yet we have to compress photos and videos to store them. The original data obtained by a camera sensor is more than required and thus not efficient. The idea of data compression is reducing the data correlation and replacing them with simpler data form. Therefore, compression is required to remove the extra unrequired information and make it efficient and low sized. Multiple steps go into the process of image compression. Firstly, we read the file into Matlab and divide it into multiple 8x8 matrices. Then, Discrete Cosine Transform (DCT) function is performed on each of the 8x8 matrices. We use a quantization table to perform quantization on the 8x8 matrice. The values of a 8x8 matrice is divided by values of quantization table and then rounded off to the lowest integer. The values of the matrice are then encoded using Difference coding and Huffman coding. Difference coding is used for encoding the DC coefficient i.e. index (0, 0). Huffman coding is further used for rest of the coefficients, which are known as AC coefficients. Huffman coding consists of Zigzag Coding which transforms the 8x8 matrice into a linear matrice. The resulting matrice shows the compressed image. The original image can then be recovered by performing decompression. The decompression steps include Decoding along with De-quantization, then Inverse Discrete Cosine Transform is performed. The resultant matrice thus provides us the recovered original image.

Keywords: DCT, IDCT, Zigzag Reordering, Quantization table, Huffman Coding

I. INTRODUCTION

JPEG is an image compression standard to store image in compressed format. It represents the Joint Photographic Experts Group. Excellent quality of JPEG is that it achieves high compression ratio. JPEG algorithm is best suited for photos and realistic scenes with smooth changes in tone and color painting. JPEG is not suitable for use with many edges and sharp changes, since this may result in many image artifacts in the resulting image. In these cases, it is best to use a lossless format such as PNG, TIFF or GIF. For this reason, JPEG is not in use for medical and scientific applications, where the image needs to be exact and slight error results into no reproduction of captured data. JPEG image may accept further losses, if it is frequently edited, and then save it. The operation of decompression and recompression can further reduce image quality. To solve this problem, the image should be edited and saved in a lossless format, only converted to JPEG format, just before the final transport to the required media. This ensures minimal loss due to frequent savings. Another advantage of this method is that it allows the user to customize the various levels of compression at runtime to fine tune the quality of compression ratio.

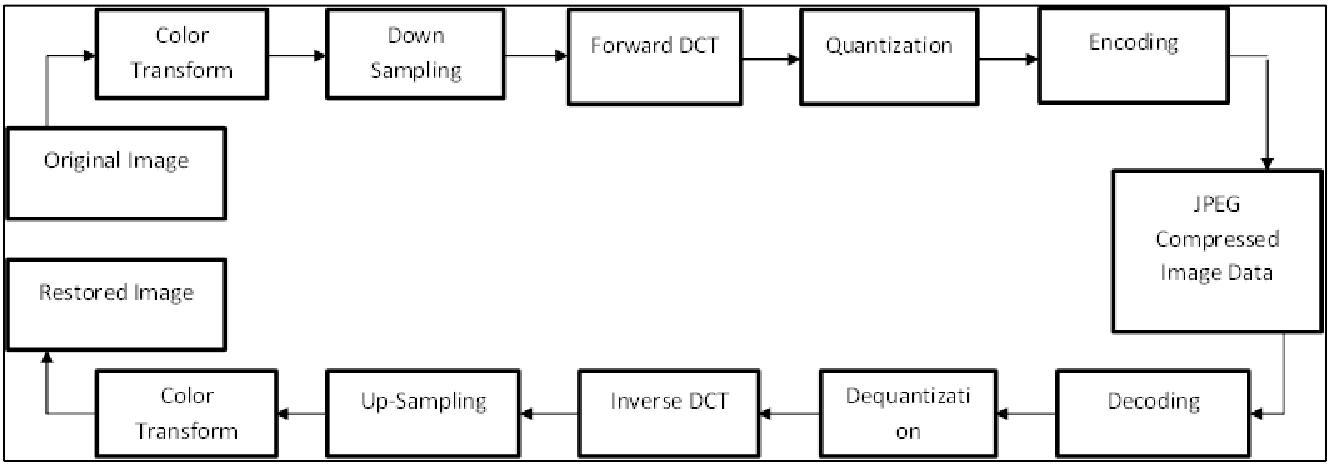

Fig. 1: Process Flow Chart

II. BLOCK DIAGRAM

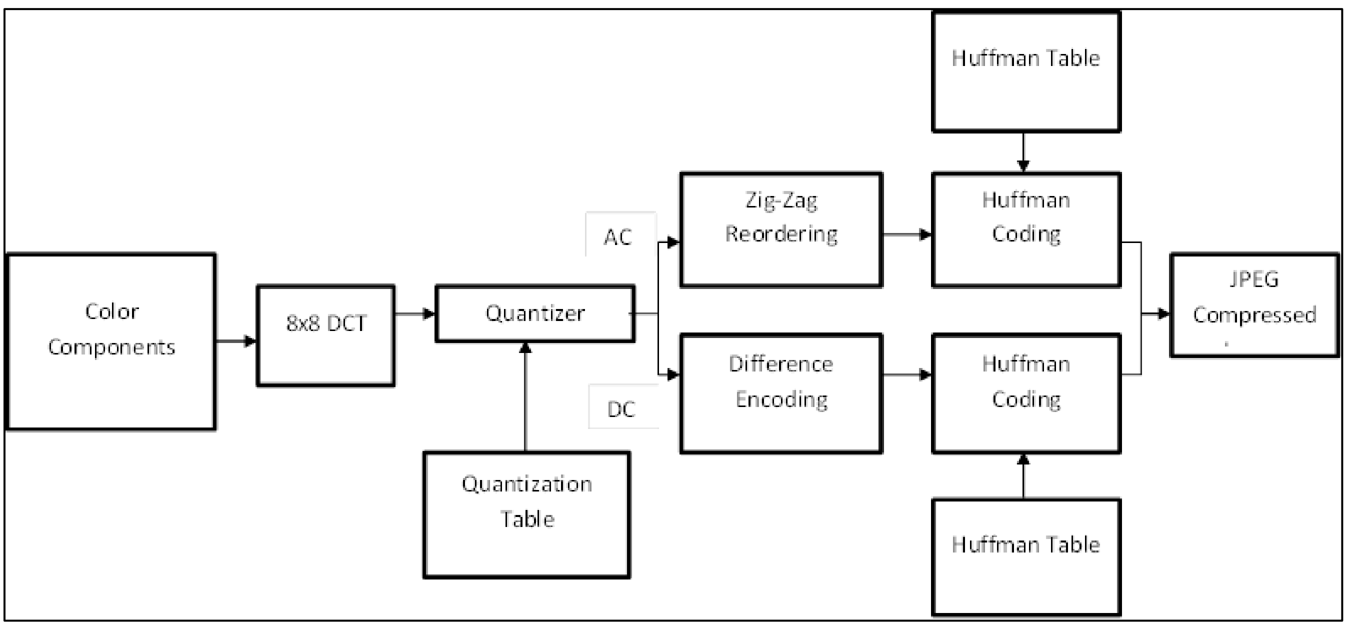

Fig. 2: Block Diagram of System

A. *Explanation of Block Diagram*

1) DCT

DCT refers to Discrete Cosine Transform. Discrete Cosine Transform is the real part of Fourier Transform Series. In DCT, the image in spatial domain is changed into frequency domain. The reason we prefer DCT over DFT (Discrete Fourier Transform) is because DFT is complex and has poor energy compaction. The spatial domain contains numbers that reflect the intensity of every channel at a given pixel whereas as a frequency domain contains the change of intensity from one pixel to the next. The frequency domain usually contains less information than the spatial domain. The reason for performing DCT is that our visual system is less sensitive to distortion around edges. The higher frequency is shown around the sharp edges whereas lower frequencies are present around smooth regions. Since, our eyes are less sensitive to high frequency components, thus they can be treated as redundant data during approximation and coding. Each pixel has 3 channels (Red, Green, Blue) and 8 bits are used to represent each channel. Thus we have 24 binary bits representing the color of the pixel. Since, 2^8 = 256, each value within the 8x8 square ranges from 0 to 255.

In order to apply the DCT, these values must be centered to zero. To do so, 128(2^8/2) is subtracted from each value.

The forward DCT equation is as follows:-

$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) cos\left[\frac{\pi(2x+1)u}{16}\right] cos\left[\frac{\pi(2y+1)v}{16}\right]$$

For u = 0, 1…….7 and v = 0, 1…..7

$$\text{Where } C(k) = \begin{cases} 1/\sqrt{2} & for\ k = 0 \\ 1 & otherwise \end{cases}$$

2) Quantizer

Quantizer is used to perform the quantization process. The transformed 8x8 block in now consists of 64 DCT coefficients. The first coefficient F (0, 0) is the DC component and the other 63 coefficients are AC component. The next step in the compression process is to quantize the transformed coefficient. Each of the 64 DCT coefficients is uniformly quantized. Each element in the quantization matrix is an integer between 0 to 255. Each DCT coefficient F (u, v) is divided by the corresponding quantized step-size parameter Q (u, v) in the quantization matrix and rounded to the nearest integer as:

$$\mathrm{Fq(u,v)} = \mathrm{Round}\left(\frac{\mathrm{F(u,v)}}{\mathrm{Q(u,v)}}\right)$$

The JPEG standard does not define any fixed quantization matrix. It is the prerogative of the user to select the quantization matrix.

In this process the high frequencies are removed from the original image. The bigger values in the quantization table is the bigger error introduced by this lossy process and the smaller visual quality.

Table – 3
Quantization Table

16	*11*	*10*	*16*	*24*	*40*	*51*	*61*
12	*12*	*14*	*19*	*26*	*58*	*60*	*55*
14	*13*	*16*	*24*	*40*	*57*	*69*	*56*
14	*17*	*22*	*29*	*51*	*87*	*80*	*62*
18	*22*	*37*	*56*	*68*	*109*	*103*	*77*
24	*35*	*55*	*64*	*81*	*104*	*113*	*92*
49	*64*	*78*	*87*	*103*	*121*	*120*	*101*
72	*92*	*95*	*98*	*112*	*100*	*103*	*99*

3) Zig-Zag Reordering

After doing 8x8 DCT and quantization over a block we have new 8x8 blocks which denotes the value in frequency domain of the original blocks. Now, we have to reorder the values into 1-D form to encode them. The DC coefficient f (0, 0) is encoded by difference coding. However, the AC terms are scanned in a Zig-Zag manner. The reason for this zig-zag traversing is that we traverse the 8x8 DCT coefficients in the order of increasing spatial frequencies. In consequence in the quantized vector at high spatial frequencies, we will have lot of consecutive zeros.

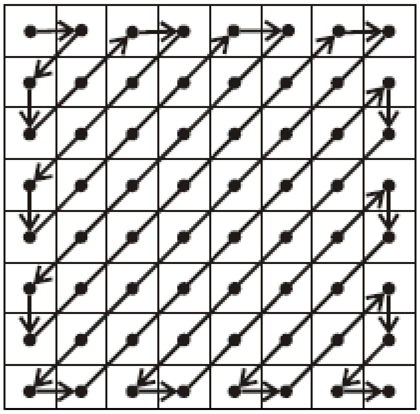

Fig. 4: Zig-Zag Reordering

4) Huffman Coding

DCT usually results in a matrix in which the lower frequencies appear at the top left corner of the matrix. Quantization then makes many of the higher frequencies round down to zero. Huffman coding takes advantage of the redundancies- such as a list of the numbers ending with a long list of zeros by inserting a new code in the place of a frequent combination of the numbers. In JPEFs a common code signifying that the rest of the coefficients in a matrix are zero is End of Block (EOB).

5) Difference Coding

Because the DC coefficients of the block are highly correlated with each other. Moreover DC coefficients contain a lot of energy so they usually have much larger values than AC coefficients. That is why we have to reduce the correlation before doing encoding The JPEG standard encodes the difference between DC coefficients. We compute the difference values between adjacent DC values by following equation.

$$\mathrm{Diff}_i = \mathrm{DC}_i - \mathrm{DC}_{i-1}$$

Note that the initial DC values is set to zero, Then the difference is Huffman encoded together with the encoding of AC coefficients.

6) Inverse DCT

$$f(x,y) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u,v)cos\left[\frac{\pi(2x+1)u}{16}\right]cos\left[\frac{\pi(2y+1)v}{16}\right]$$

For u = 0…7 and v = 0…7
F (u, v) is called the DCT coefficient.

III. ADVANTAGES

1) JPEG gives remarkable compression ratios.
2) JPEG Compression is capable of storing 24 bit per pixel color data instead of 8 bit per pixel color data which were used before.
3) It is the best technique to retain quality of an image even after compression.
4) It allows user to customize the various levels of compression at run time.
5) Works with both color and grayscale images.
6) Suitable for many applications like satellite, medical, general photography etc.

IV. DISADVANTAGES

1) JPEG is not suitable for use with many edges and sharp edges, since this may result in many image artifacts in the resulting image
2) JPEG compression technique is a lossy compression technique
3) JPEG image do not support layered images. Graphics designer need to work on layered images in order to manipulate and edit graphic images.
4) Only 8 bit images are supported by JPEG format. On the other hand, modern high resolution digital cameras support 10, 12, 14, or 16 bit images. If these images are in JPEG format, extra information is discarded, resulting in decreased image quality.

V. CONCLUSION

In this paper we have studied the main process of JPEG ways to encoding and decoding. Compression can be achieved by using the DCT technique which splits the image into different frequency components. Then the unnecessary information can be removed from the image by quantization. For recovering the original image from the compressed image, Dequantization is performed followed by the inverse DCT.

REFERENCES

[1] Pao-Yen Lin, "Basic Image Compression and Introduction to JPEG Standard", Graduate Institute of Communication Engineering National Taiwan University, Taipei, Taiwan, ROC.
[2] Ms. Anamika and Mrs. Preeti, "JPEG COMPRESSOR USING MATLAB", IJAERT, Volume 3, Issue 5, May 2015.
[3] Harsh Vardhan Dwivedi, B.Tech, Dept. of Electronics and Communication Engineering, NIT Rourkela, 2009.
[4] R. C. Gonzalez and R. E. Woods, Digital Image Processing 2/E. Upper Saddle River, NJ: Prentice-Hall, 2002.
[5] G. K. Wallace, 'The JPEG Still Picture Compression Standard', Communications of the ACM, Vol. 34, Issue 4, pp.30-44.
[6] L.Agostini, S. Bampi. "Pipelined Fast 2-D DCT Architecture for JPEG Image Compression "Proceedings of the 14th Annual Symposium on Integrated Circuits and Systems Design, Pirenopolis, Brazil. IEEE Computer Society 2001. pp. 226-231.
[7] Y. Arai. T. Agui. M. Nakajima. "A Fast DCT-SQ Scheme for Images". Transactions of IEICE, vol. E71. Na. 11. 1988. pp.1095-1097.
[8] Watson, A. B. Image Compression Using the Discrete Cosine Transform. Mathematica Journal, 4(1), 1994, p. 81-88.