

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Paper / Case Study

Available online at: www.ijarcsms.com

Basics of Jpeg Still Image Compression

Kopal Rathore

M.Tech (C.S.E)

Banasthali Vidyapith, Banasthali, India

Abstract: This paper describes the basics of JPEG still image compression. A video is basically a sequence of frames. When multiple frames pass in front of our eyes, we visualize video rather than still images. These frames may be stored as they are captured resulting a large file size. In order to reduce the size of captured file, compression is used. This paper is the outcome of our study of MJPEG compression. MJPEG uses still image JPEG compression. The compression process involves Discrete Cosine transform, Quantization, Entropy encoding. The purpose of this paper is to give an implementation level understanding.

Keywords: DCT, Quantization, zigzag reordering, Huffman coding.

I. INTRODUCTION

The term image compression is generally used to reduce the size of multimedia without effecting or compromising the originality of it. The reduction in size can provide a facility to store more and more data in storage like hard disk etc. With the growth of computer applications the need of storage is increase for that we such an efficient techniques due to which such need can be easily fulfilled. Image compression is an application of data compression that encodes the original image with few bits. The main purpose of image compression is to reduce the redundancy of the image and efficiently store or transmit the data. Figure.1 shows the block diagram of the JPEG image compression system. The main objective of such system is to reduce the storage capacity as much as possible, and the decoded image through which it can be similar to the original image [5].

The JPEG standard is one of the techniques that is mostly used and a part of lossy compression that based on DCT. The DCT (Discrete cosine transformation) is based on the workings of separating images into parts of various different Frequencies. During the process of quantization, where the each part of compression really occurs, the less important or unnecessary frequencies are discarded and the remaining most important or necessary frequencies are used to retrieve the image in the reconstruction process . As a result, reconstructed multimedia data is a data with distortion but later this distortion is recovered by applying some techniques.

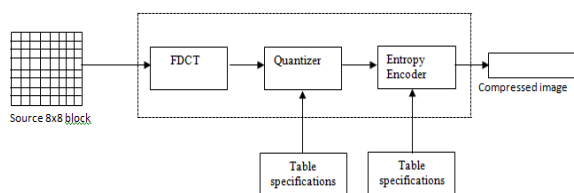


Figure 1. JPEG Encoder

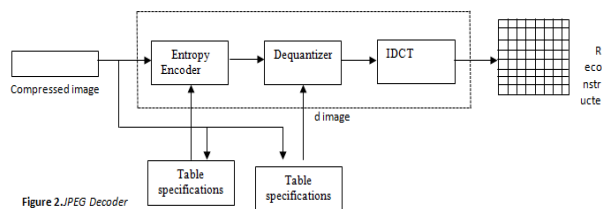


Figure 2. JPEG Decoder

Fig 1: Encoder & Decoder [1]

II. JPEG METHOD OF COMPRESSION

The following is the overview of JPEG encoding process in image compression using DCT describes in the following steps:

1. The image is firstly broken into 8x8, 16x16 or 32x32 blocks of pixels and do Color Space Transform.(RGB to YCbCr).
2. It Works from left most corners to right most or top to bottom, the DCT can be applied to each block of image.
3. Each block is starts compressing by performing quantization process.
4. Zigzag Scanning.
5. DPCM on DC component.
6. RLE on AC component.
7. Entropy Coding-Huffman or Arithmetic.

III. COLOR SPACE CONVERSION

In order to achieve good compression performance, correlation between the color components is first reduced by converting the *RGB* color space into a decorrelated color space. In baseline JPEG, a *RGB* image is first transformed into a luminance-chrominance color space [4] such as YC_bC_r . The advantage of converting the image into luminance-chrominance color space is that the luminance and chrominance components are very much decorrelated between each other. Moreover, the chrominance channels contain much redundant information and can easily be subsampled without sacrificing any visual quality for the reconstructed image. The transformation from *RGB* to YC_bC_r is based on the following mathematical expression:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299000 & 0.587000 & 0.114000 \\ -0.168736 & -0.331264 & 0.500002 \\ 0.500000 & -0.418688 & -0.081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

The value $Y = 0.299R + 0.587G + 0.114B$ is called the luminance. It is the value used by monochrome monitors to represent an *RGB* color. Physiologically, it represents the intensity of an *RGB* color perceived by the eye. The eye is most sensitive to the Green component then it follows the Red component and the last is the Blue component. The values C_b and C_r are called chrominance values. So we take an 8*8 image. And we convert this RGB image to YCbCr through the previous matrix. So we get the value of Luminance (Y).

Y=	130	130	130	130	130	130	130	130
	130	130	130	130	130	130	130	130
	255	196	130	130	130	130	130	130
	255	146	130	130	130	130	130	130
	130	130	130	130	130	130	130	130
	130	130	130	130	207	255	255	54
	130	130	130	130	130	130	173	74
	255	215	138	130	130	130	130	130

IV. DISCRETE COSINE TRANSFORM

To apply the DCT, the image is divided into 8x8 blocks of pixels. If the width or height of the original image is not divisible by 8, the encoder should make it divisible. The 8x8 blocks are processed from left-to-right and from top-to-bottom [2].

The purpose of the DCT is to transform the value of pixels to the spatial frequencies. These spatial frequencies are much related to the level of detail present in an image. High spatial frequencies correspond to high levels of detail, while lower frequencies correspond to lower levels of detail. The mathematical definition of DCT is:

Forward DCT:

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

for $u = 0, \dots, 7$ and $v = 0, \dots, 7$

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

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 $x = 0, \dots, 7$ and $y = 0, \dots, 7$

The $F(u, v)$ is called the DCT coefficient, and the DCT basis is :

$$\omega_{x,y}(u, v) = \frac{C(u)C(v)}{4} \cos \left[\frac{\pi(2x+1)u}{16} \right] \cos \left[\frac{\pi(2y+1)v}{16} \right]$$

Then we can rewrite the inverse DCT to :

$$f(x, y) = \sum_{u=0}^7 \sum_{v=0}^7 F(u, v) \omega_{x,y}(u, v) \quad \text{for } x = 0, \dots, 7 \quad \text{and } y = 0, \dots, 7$$

And $f(x, y)$ is the matrix of values Y from the original image. Let's take an 8x8 block of pixel values from an image:

Y=

130	130	130	130	130	130	130	130
130	130	130	130	130	130	130	130
255	196	130	130	130	130	130	130
255	146	130	130	130	130	130	130
130	130	130	130	130	130	130	130
130	130	130	130	207	255	255	54
130	130	130	130	130	130	173	74
255	215	138	130	130	130	130	130

The DCT can now be applied on original matrix which results in following matrix.

F(u, v) =

1138.5	74.4	39.9	103.2	-19.1	35.5	-22.4	14.3
-36	-3.8	12	-34.2	30.3	-13.6	18.4	-8.1
-19	7.1	1.5	-23.1	-4.4	-11.3	-8.4	-5.7
-35	-92.1	-83.4	8.4	-51.1	-11.3	-9.6	13.6
-9.6	31	37.6	-27.3	37.8	-14	17.6	-9.5
-4.16	14.2	7.5	10.8	4.6	19.2	-1.7	7.4
78.4	11.2	6.9	45.8	-14.2	-3.1	-5.4	0.7
-40.3	14.5	12.2	-38.3	10.9	-12.2	-4.3	-8.6

This matrix now contains 64 DCT coefficients. The upper left coefficient correlates to the low frequencies. The lower right values represent higher frequencies. As human eye is more sensitive to low frequencies, the next step of quantization will reflect this fact.

V. QUANTIZATION

The transformed 8×8 block 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 DC component $F(0,0)$ is essentially the sum of the 64 pixels in the input 8×8 pixel block multiplied by the scaling factor $(1/4)C(0)C(0)=1/8$ as shown in equation 3 for $F(u,v)$. The next step in the compression process is to quantize the transformed coefficients. Each of the 64 DCT coefficients is uniformly quantized. The 64 quantization step-size parameters for uniform quantization of the 64 DCT coefficients form an 8×8 quantization matrix. Each element in the quantization matrix is an integer between 1 and 255. Each DCT coefficient $F(u,v)$ is divided by the corresponding quantizer step-size parameter $Q(u,v)$ in the quantization matrix and rounded to the nearest integer as

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

The JPEG standard does not define any fixed quantization matrix. It is the prerogative of the user to select a quantization matrix. We have a quantization matrix for Luminance:

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

The resulted matrix after division is:

71	6	3	6	0	0	0	0
-3	0	0	-1	1	0	0	0
-1	0	0	0	0	0	0	0
-2	-5	-3	0	-1	0	0	0
0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Here, zeros represent the higher frequencies that have been discarded. This results in lossy compression. Only the remaining non zero coefficients will be used to reconstruct the image.

VI. ZIGZAG REORDERING

Now the 8x8 matrix has been quantized and more than half of the DC coefficients are equal to zero.

JPEG incorporates run-length coding to take advantage of this by encoding quantized coefficients in the zigzag sequence as shown in figure below [1].

frequency of occurrence. Smaller codeword represents the most common symbol. Huffman coding is also uniquely decodable, which means that the code generated from a string cannot represent something other than the original input when it is decoded [1].

X. CONCLUSION

Image compression is an extremely important part of modern computing. By having the ability to compress images to a fraction of their original size, valuable (and expensive) disk space can be saved. Further, Image compression has played an important part in the transportation of images from one computer to another becomes easier and less time consuming. The JPEG image compression algorithm provides a very effective way to compress images with minimal loss in quality. We have introduced the basic concepts of image compression and the overview of JPEG standard. It is very useful for storing pictures as visually unimportant information can be discarded. It has become the most popular image format. In this paper, the ability of the JPEG image compression to retain image quality while drastically reducing the file size is introduced through the implementation of DCT, quantization, reordering and Huffman coding.

References

1. Anmol Jyot Maan, "AN INTRODUCTION TO JPEG IMAGE COMPRESSION ALGORITHM", Submitted in December 2013 for publication in International Journal of Electrical, Electronics and Data Communication.
2. K. Deepthi, "Design and Implementation of JPEG Image Compression and Decompression", International Journal of Innovations in Engineering and Technology (IJJET).
3. S. V. Viraktamath "Performance Analysis of JPEG Algorithm" 978-1-61284-653-8/11/\$26.00 ©2011 IEEE.
4. Muzhir Shaban AL-Ani, Fouad Hammadi Awad, "THE JPEG IMAGE COMPRESSION ALGORITHM", submitted in International Journal of Advances in Engineering & Technology, May 2013.
5. V.S.Arun, "Image Compression using JPEG Algorithm", Digital Signal and Image Processing Centre, MSRSAS, Bangalore.