

# A Study on Image Compression Technology

Cao Yueqin

Computer Department of Wenzhou Vocational and Technical College  
Wenzhou, Zhejiang Province, 325027

**Abstract-**With constant development of multimedia technology, large amount of image information is given more and more attention by people. High-quality image compression becomes the hot subject of current study. In this paper, basic on the algorithm of JPEG image compression code and applying the periodicity and symmetry of cosine function, we have made proper improvement to the method of discrete cosine transform reducing large amount of cosine and multiplication operations in computer and greatly improving the compression rate. At last, Matlab software was adopted to emulate and validate standard images. Results of experiments show that compression rate of images has been improved within a large compression scope.

**Keywords-** image treatment, compression technology, coding algorithm, simulation

## I. INTRODUCTION

Image is one of important media to carry multimedia information. According to statistics, people gain 70% of information from visual system. Image data is mainly featured by its large information volume, for example, an image of size 512\*512 and gray level of 8bit contains the data volume of 256K bytes; for a 4-channel satellite remote sensing probe, the volume of every image transmission can reach to 1M bytes [1]. In actual application of satellite remote sensing, it contains more windows and channels than that, for example, a synchronous satellite of the earth can detect one third of global areas, so it can be imagined that a huge amount of storage space is required to store these images [2] [3].

At present, JPEG still image compression standard works very well in medium-high rate, however, in the case of low bit rate, image reconstruction has serious block effect and thereby cannot meet the demand of image transmission through internet [4]. In this paper, we performed JPEG compression and reconstruction of standard images with Matlab basic on the principle of JPEG image compression, and then analyzed test results to gain conclusions.

## II. IMAGE COMPRESSION TECHNOLOGY

The goal of image compression can be saving the capacity of image memory, reducing the volume of signal channels as well as shortening the time required for image processing [5].

### A. Compression Technologies

JPEG is the format made for the embodiment of coding standards of color static images in terms of image database, color fax and printing. It is a universal standard for a wide

range of application. This compression standard is applicable to the digital image compression and coding of static images, chromatic, unicolor and multi-gray continuous tones. Comparing to other common file formats of same image quality, such as GIF, TIFF and PCX, JPEG possesses the higher compression rate than current compression algorithms of static images [6] [7].

### 1) Selection of Compression Technologies

There are two basic defined algorithms of JPEG, fuzzy compression algorithm which is basic on DCT, including baseline system, sequence method and expansion system (increasing mode and layering code); the one is lossless compression algorithm, also called as Spatial method [8]. DCT-typed algorithm offers higher compression rate, but is more complicated; space forecasting algorithm can embody medium compression. Selection of process technologies lies on the consideration of specific application demands and cost-efficiency. In view of excellent quality of JPEG, a great success is made only a few years after the standard is launched. At present, about 80% of images online are compressed by JPEG standard. It is very necessary to master the core algorithm of JPEG, the leading coding technology which serves as a basic for successive image processing. In this paper, we adopted the first lossy compression technology basic on DCT.

### B. Compression Principles

#### 1) JPEG Coding

The DCT-based lossy coding process is shown in figure 1. The baseline system of JPEG is a sequence DCT algorithm which introduces errors to quantization process, so the compression is lossy. They system removes the redundant information of vision and data with quantization and lossless compression coding by taking advantage of physiological properties of human's vision system. JPEG coding process can be generally described as: firstly, an image is divided into many 8\*8 blocks, namely each block having 8\*8=64 pixels; the resolution of image determines the number of blocks in such a way that higher resolution means more blocks, and vice versa; then, perform 2-D DCT transformation on each 8\*8 block to acquire the 64 DCT coefficients of frequency domain, and quantify the 64 coefficients. The required quantization table and specific values for entropy coding in table vary with the image to be coded. Recommended quantization table of JPEG standard is usually adopted. Quantization is processed by quantization table which is regulated by the characteristic of human eyes in JPEG organization. The process of dividing by quantization table to gain the 64 coefficients is quantization. After quantization, we still get a (8\*8) 64 coefficient whose low frequency is centralized in the 8\*8

coefficient in the upper left corner. At last, give entropy coding with entropy table. After that, we can get the compressed image data [9].

2) Discrete Cosine Transform (DCT)

DTC embodies image compression coding by orthogonal transformation. After sampling of analog images, it is called discretized brightness. Discretized pixels can be 256 \*256 (8-bit) or 512 \*512 (16-bit) or other sizes. If a whole image were to be transformed once, the calculation would be very complicated and time-consuming. Usual practice is that an image is divided into some subareas in horizontal and vertical directions, with each subarea of 4\*4 or 16\*16 pixels. With subarea as the unit of transformation, all pixels of every subarea form a spatial matrix [13] [14]. The matrix is described like formula 1.

$$[f]=\begin{Bmatrix} f(0,0) & f(0,1) & \dots\dots & f(0,n-1) \\ f(1,0) & f(1,1) & \dots\dots & f(1,n-1) \\ \dots\dots & \dots\dots & \dots\dots & \dots\dots \\ f(n-1,0) & f(n-1,1) & \dots\dots & f(n-1,n-1) \end{Bmatrix}$$

Formula 1

Elements of matrix f (i, j) represents the brightness of row i and line j. The transformation of N\*N spatial matrix is also N\*N order. Transformation is to multiply both sides of spatial matrix by transformation matrix [T] to acquire the matrix within transformation domain [F]:

$$[F]=[T]*[F]*[T]$$

As an orthogonal matrix [T] is full rank, they have inverse matrixes. Assuming that the inverse matrix of [T] is [T]-1, multiply the both sides of formula 3-5 by [T]-1, and we can get:

$$[F]=[T]^{-1}*F*[T]^{-1}$$

The above two formulas summarizes the basic thought and operation of transformation in transformation coding.

After a series of transformations, assume that f (x, y) is the data of original images, F (u, v) is transformed data, then the 2-D positive DCT of 8\*8 image can be defined as formula 2 [10][11].

$$F(u,v)=\frac{1}{4}c(u)c(v)\sum_{x=0}^7\sum_{y=0}^7f(x,y)*K$$

Formula 2

Thereinto:

$$c(u),c(v)=\begin{cases} \frac{1}{\sqrt{2}} & \dots \text{当 } u,v=0 \\ 1 & \dots \text{当 } u,v \neq 0 \end{cases}$$

$$K=c_0\frac{(2x+1)\pi}{16}c_0\frac{(2y+1)\pi}{16}$$

III. IMPROVEMENT AND EMBODIMENT OF IMAGE COMPRESSION TECHNOLOGIES

A. Defects of DCT

From formula 2, we can see that each transformed data is composed by the sum of 64 terms, each of which consists of image data f (x, y) (varying 0~255) and DCT coefficients. Transformation is the data of original images and coefficients of DCT are constant in each time. When the value of x and y is within 0-255, multiply a constant with cosine which is gained through computer calculation

creating 256\*256 possibilities of cosine. Then, we can know from calculation that the coefficient table is a 8\*8 dot matrix pixel. The value of U and V varies within 0-7, namely the number of data in computer operation is 8\*8\*256\*256=4194304. If each data is represented by floating point number, the capacity is 4194304\*4=16777216 bytes, about 16M bytes, so the data volume is comparatively large.

B. Improved DCT

1) Formation of Grouping Reference Tables

Firstly, divide x and y into blocks by the periodicity of cosine function which is featured by its periodicity. Analysis is made using the periodicity and symmetry of function. When the value of U and V is certain, substitute the value of X and Y into cosine formula; when x=0 and x=16, the value of the function is the same and their period is 2π. According to this rule, we divide x and y into blocks within 0-255 and get 15 blocks shown in table I. When x=94, we can see from table I that it is K14. In this way, we can reduce the calculation involving 256 functions to the one with only 16 functions, greatly decreasing the calculation volume of computer.

2) Creation of cosine coefficient table

Secondly, it is the creation of cosine coefficient table. In the above, we grouped the 256 sets of x and y values by their periodicity and 16 groups were got from 0 to 15. As the 8-bit plane image is a 2-D data, make the 16 x-y blocks into an address mapping table of 2-D cosine coefficients basic on the original block, as shown in table II. Then, when x=0, y=0, we can get K(0,0) from table II. Likewise, when x=66, y=100, table 1 shows that x is K2, y being k4. From table II, we can know that cosine mapping coefficient is (2, 4).

3) Creation of Cosine Values

At last, create cosine values according to mapping table of cosine coefficients. In the above, we only got the mapping value of coefficients of cosine function by referring to table according to value of x and y. In practical technology, what is required is the value of each cosine coefficient, not the coefficient itself, when u, v, x and y are given different values in the formula 2. Therefore, in each block of table II, we substituted the value of v and u varying 0~7 into the coefficients for calculation and got the mapping table of practical coefficient values like table III. For example, in coefficient K (0, 0), as the value of x and y is constant, the coefficients of formula 2 are the 8 numbers when u and v are assigned 0~7, so we have a table of 8\*8=64 cosine function values, as shown in table III. From table II, we can know that each block requires a cosine value table, so 15\*15=256 tables of such are required as well as 64\*256 operations. Compared with the 8\*8\*256\*256 cosine operations above, it is much less.

During calculation, firstly, multiply each DCT coefficient by 1/4 and possible value of image data varying within 0~255. Check the block value from table 1 and then the value of DCT coefficient table from table II with block value. From table II, we can see that the value of u and v determines which coefficient of cosine terms to be

multiplied with  $f(x, y)$ . By this relation, we can make an address mapping table to check related coefficients in DCT transformation. At the start of DCT transformation, the system firstly loads the coefficient table and address mapping table, then determines the position in address mapping table by the value of  $u$  and  $v$ , and draws the 64 values from the back, and locates the starting place of these values in coefficient table, and adds the value of  $f(x, y)$ , and then adds the values to finish the calculation of one system in DCT. From the above, we can see that in applying table inquiry method, the system multiplies the coefficients with all possible values of image, and then saves the results in the table which is loaded into the memory in operation. This way can avoid the operation of multiplication and thereby accelerates the operation rate comparing to regular algorithms.

C. Improvement Embodiment of DCT Technology

1) Matlab Simulation Experiment

The system module diagram of JPEG coding system is shown in Figure2. After an original image is inputted, it is buffered and coded by DCT. Coding result is through ZZ scanning, quantization, code stream impulse, Huffman coding and then outputted.

2) Experimental Results

To make a intuitionistic comparison, we performed an simulation experiment of JPEG standard algorithm and improved DCT algorithm with two images in the same computer. Coding time was calculated by setting time parameter and the result is shown in table IV. This experiment adopted image sorcel.bmp and sorcel2.bmp which is 256\*256 and 8-bit. Experiment result shows that the improved algorithm raised by this paper saves about 40% of time comparing to original algorithm.

IV. SUMMARIES

In view of low compression rate of JPEG algorithm caused by large amount of cosine calculation, we created the table of a series cosine values by periodicity of cosine function, then carried out DCT transformation through table reference and thereby greatly decreased large amount of cosine operation improving the compression rate of JPEG. An analysis was made basic on experiments. Many points

need to be improved in the process, for example, the compression effect on larger images is imperfect. It is expected to improve the compression quality of various sizes of images by modifying quantization process. We can still take advantage of improved multimedia performance of computers to divide the conventional 8\*8 data block to 16\*16 one.

ACKNOWLEDGEMENTS

Fund Project: A Study on Static Image Processing and Compression Algorithm ( WZY2011002 )

Author Resume: Cao Yueqin (1971), female (Han nationality), from Taizhou, Jiangsu.lecturer and master: main study field includes data digging and graphics.

REFERENCES

- [1] Liu Wei, Wang Hong Xing, Lossless Coding Methods and JPEG Standard Mode of Images [J], Modern Electronic Technology, 2002, 5: 7-10;
- [2] G.K.Wallace.The JPEG Still Picture Compression Standard[M].Commun.ACM,1991, 34(4):30-44;
- [3] Yan Jian, Huffman Algorithm and Its Application to Data Compression [J], Computer and Moderation, 1996,48:15-20;
- [4] V.K.Goyal.Theoretical Foundations of Transform Coding[J].IEEE Signal ProcessingMag.2001,18:9-21
- [5] S.W.Golomb.Run-length encoding[J].IEEE Trans.on Information Theory.1966,12:399-401
- [6] Shen Lansun, Image Coding and Asynchronous Transmission [M], Beijing: People's Posts and Telecom Press, 202:25-4;
- [7] Zhang Xianda, Modern Digital Signal Processing [M]. Beijing: Tsinghua University Press, 1999:433-448;
- [8] Tian Xiaoxia, Liu Shuai, Wang You Song, et al, A Study on JPEG Baseline Compression Algorithm and Optimization [J].Mechanics and Electronics, 2005(4):6-8;
- [9] Yan Yusong, Cheng Qian Sheng, Application of Reversible Biorthogonal Wavelet to Image Compression [J]. China Academic.Journal of Graph and Image, 1997;
- [10] Wang Yao, Zhu Guang Xi, Liu Wei, An Image Coding Scheme Combining Wavelet Pre-Treatment [J], Academic Journal of.Computer Science, 203,26(2): 240-243;
- [11] Eric Hamilton. JPEG File Interchange Format. www3.org/Graphics/JPEG/jff.txt.2005.10.6.

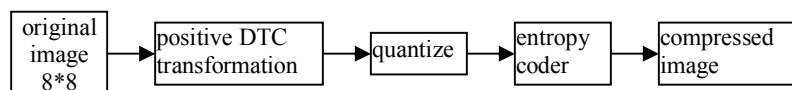


Figure 1. Block Diagram of JPEG Compression Coding

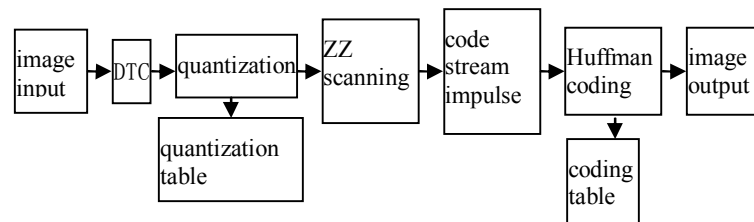


Figure 2. JPEG System Module

TABLE I. BLOCK TABLE OF X AND Y

K0	K1	K2	K3	K4	K5	K6	...	K14	K15
0	1	2	3	4	5	6	...	14	15
16	17	18	19	20	21	22	...	30	31
32	33	34	35	36	37	38	...	46	47
48	49	50	51	52	53	54	...		63
64	65	66	67	68	69	70	...	78	79
80	81	82	83	84	85	86	...	94	95
96	97	98	99	100	101	102	...	110	111
...	...	...	...	...	...	...	...	...	...
240	241	242	243	244	245	246	...	254	255

TABLE II. ADDRESS MAPPING TABLE OF X-Y COSINE ADDRESS

	0	1	2	3	...	15
0	K(0,0)	K(0,1)	K(0,2)	K(0,3)	...	K(0,15)
1	K(1,0)	K(1,1)	K(1,2)	K(1,3)	...	K(0,15)
2	K(2,0)	K(2,0)	K(2,0)	K(2,0)	...	K(0,15)
3	K(3,0)	K(3,1)	K(3,2)	K(3,3)	...	K(0,15)
4	K(4,0)	K(4,1)	K(4,2)	K(4,3)	...	K(0,15)
...	...	...	...	...	...	...
15	K(15,0)	K(15,1)	K(15,2)	K(15,3)	...	K(15,15)

TABLE III. TABLE OF COSINE FUNCTION VALUES

	0	1	2	3	4	5	6	7
0	C0*C0	C1*C0	C2*C0	C3*C0	C4*C0	C5*C0	C6*C0	C7*C0
1	C0*C1	C1*C1	C2*C1	C3*C1	C4*C1	C5*C1	C6*C1	C7*C1
2	C0*C2	C1*C2	C2*C2	C3*C2	C4*C2	C5*C2	C6*C2	C7*C2
3	C0*C3	C1*C3	C2*C3	C3*C3	C4*C3	C5*C3	C6*C3	C7*C3
4	C0*C4	C1*C4	C2*C4	C3*C4	C4*C4	C5*C4	C6*C4	C7*C4
5	C0*C5	C1*C5	C2*C5	C3*C5	C4*C5	C5*C5	C6*C5	C7*C5
6	C0*C6	C1*C6	C2*C6	C3*C6	C4*C6	C5*C6	C6*C6	C7*C6
7	C0*C7	C1*C7	C2*C7	C3*C7	C4*C7	C5*C7	C6*C7	C7*C7

TABLE IV. COMPARISON OF OPERATION RATE

	Original Algorithm	Improved Algorithm
Sorcel(10 <sup>-6</sup> s)	5.632	3.168
Sorcel(10 <sup>-6</sup> s)	7.426	4.502