Application binary state arithmetic encoding in JPEG

[1]Ondrej KOVÁČ (1st year), [2]Kornel RUMAN (1st year), [3]Ján VALISKA (1st year)
Supervisor: [4]Ján MIHALÍK

[1,3,4] Dept. of Electronics and Multimedia Communications, FEI TU of Košice, Slovak Republic
[2] Dept. of Technologies in Electronics, KTE TU of Košice, Slovak Republic

[1]ondrej.kovac@tuke.sk , [2]kornel.ruman@tuke.sk , [3]jan.valiska@tuke.sk, [4]mihalik.jan@tuke.sk

***Abstract*— This paper shortly describes basic image JPEG format. This paper also describes issue of decomposition of DCT image spectral coefficients to its bit levels and their encoding using binary state arithmetic code. In the conclusion, the advantage for replacement of conventionally used Huffmann code by binary state arithmetic encoding of the picture´s bit levels DCT spectral coefficients image is contemplated .**

***Keywords*— JPEG, DCT compression, bit plane and arithmetic encoding.**

I. INTRODUCTION

The transmission speed of connection has been increasing, but given the fact that the amount of data transferred also increases, it is compulsory to enhance the methods of compression applied to the multimedia data.

This article is focused on an algorithm of source encoding in JPEG. In JPEG, the Huffman encoding is being used. We will try to replace this algorithm using the binary state arithmetic encoding.

II. JPEG WITH DISSIPATION ENCODING

Block diagram of image encoder is on figure 2.1. There can be colorful or grey scale image on the input. This structure corresponds with intern-image transformation encoding with DCT block size 8x8 pixels and with entropic encoding.[1]

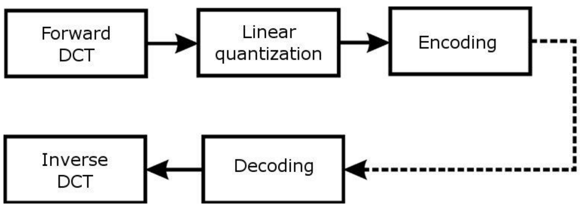


Fig. 2.1. Block diagram image codec JPEG with dissipation encoding.

At the beginning, the image is divided to 8x8 block pixels and after this, DCT is applied. All spectral coefficients (SC) are quantized with different or equal quantization step, which changes, depending on statistical properties or application. Quantized SC are scanned in a diagonal order called with quantization step. [2] Quantized SC are scanned in diagonal order called Zig-zag scanning. Zig-zag is showed on figure 2.2. By using this scanning, we achieve that SC with high energy end small frequency will be scanned before SC with high frequency but small energy do so. SC with zero energy are detected and counted. This allows making couples "number of zeros and non-zero values of SC". These couples are entropic encoded by Huffman code. [3]

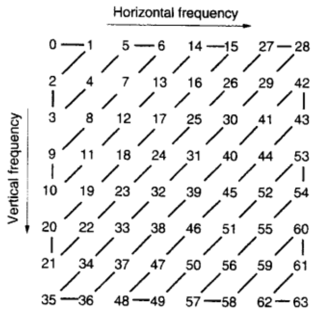


Fig. 2.2 Zig-zag scanning

Each one dimensionally represented block must be ended with special couple [0][0] called EOB (end of block).

A. *Used quantizers*

By quantization in space of SC, each of SC is quantized, using its own quantizer. This is showed on figure 2.3.

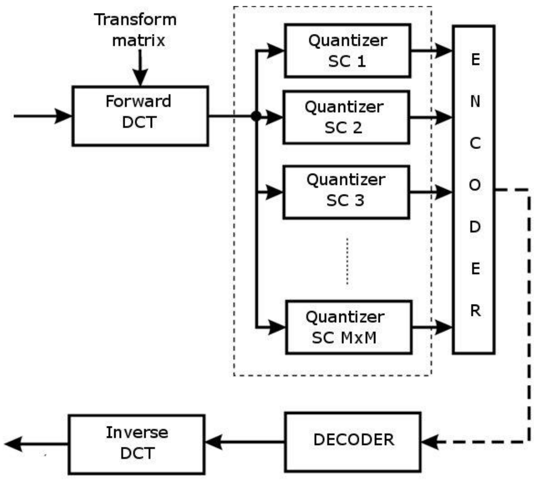


Fig. 2.3 Block diagram of transformation encoding system

In this letter we will use two types of quantizers. We will use one for DC spectral coefficients (DSC) and one for AC spectral coefficients (ASC).

1) ASC quantization

For quantization of ASC, the used algorithm is described by the equation 2.1

$$q_{ij} = 8Y_{ij}//MK\Delta_{ij} \qquad (2.1)$$

Where q_{ij} represents an index of quantization and takes its values on interval <-255,255>. Y_{ij} is a value of original spectral coefficient and MK is a scaling factor. Δ_{ij} is quantization step. Operation // means dividing with rounding on nearest integer.
Results of this operation are quantization indexes. [2]
Reverse operation is described by equation 2.2.

$$\tilde{Y}_{ij} = q_{ij}MK\Delta_{ij}/\,8 \qquad (2.2)$$

2) DSC quantization

For quantization DSC, algorithm represented by equation 2.3, is used.

$$q_{00} = Y_{00}//\,8 \qquad (2.3)$$

Usually quantization of DSC is performed with fix set step. The obtained quantization level is showed in equation 2.4.

$$\tilde{Y}_{00} = 8q_{00} \qquad (2.4)$$

III. Image Decomposition to Bit Plane

In general, each image can be expressed like two dimension rasters of pixels and each pixel can be described by N bits as follows:

$$\text{pixel} = p_0 2^0 + p_1 2^1 + p_2 2^2 + \ldots\ldots + p_{N-1} 2^{N-1}$$

Individual bits P_i correspond to the respective weights 2^i for i=0,1, ... ,N-1. If we pick up only P_0 from in a such way expressed pixel from entire polytonal image, we get its bit plane of zero weight (BP_0). Using the same process all other bit-planes until BP_{N-1} are created.

In a transformed space, there are not only spectral coefficients with positive value. Because of this fact, there are two various ways for decomposition [2].

- *Decomposition of the real bit-plane (RBP)*
- *Decompositions of the absolute bit-plane and plane of sign (ABP)*

We not decompose image of spectral coefficients, but image of quantization indexes in range <-255; 255>.

A. Decomposition of the real bit-plane (RBP)

In this decomposition, we get bit-planes $-BP_{N-1}$ ... $-BP_{0.}$ .... BP_0 ... $BP_{N-1.}$ Figure 3.1. shows the block diagram of RBP decomposer.

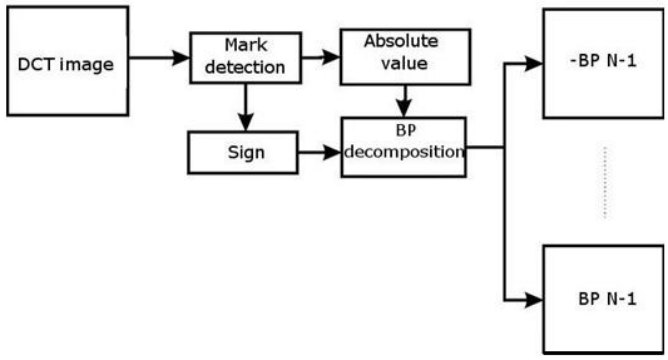


Fig. 3.1 Block diagram of RBP decomposer

Each of spectral coefficients of transformed image is tested on a sign and the result of this is a mark defining either positive or negative bit plane. After this, the image is transformed into absolute value and decomposed with identical method like decomposition of image in Gaussian space.

B. Decompositions of the absolute bit-plane and plane of sign (ABP)

In this decomposition, we get bit-planes $BP_{0,}$ BP_1 ... BP_{N-1} and $BP_{s.}$ Block diagram of ABP decomposer is showed on figure 3.2.

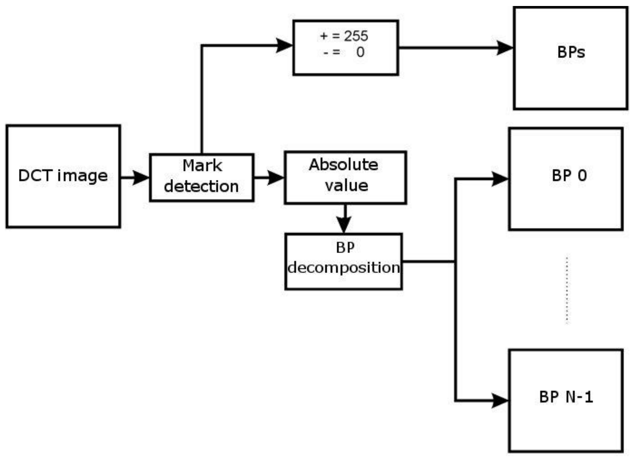


Fig. 3.2 Block diagram of ABP decomposer

Each of spectral coefficients of transformed image is tested on the sign and the result shows value of pixel in bit-plane BP_s. This value is 255, or white for positive and 0, or black for negative mark. After this, the image is transformed on absolute value and decomposed with identical method like in previous case.

IV. ARITHMETIC ENCODING

A. Arithmetic code without using multiplication

1) Encoding

Procedure of the arithmetic encoding (AE) without using multiplications is described with equations 4.1 – 4.4. [4]

$$A(sk) = A(s).2^{-Q} \tag{4.1}$$
$$A(sm) = < A(s) - A(sk) > \tag{4.2}$$
$$C(sm) = C(s) \tag{4.3}$$
$$C(sk) = C(s) + A(sm) \tag{4.4}$$

Where **C(s)** represents binary variable of sequence **S** with 0 initial value before encoding first symbol, and expresses low limit of probabilistic subinterval. **C(sm)** is binary variable of sequence **S** for the more probable symbol and **C(sk)** for less probable symbol of sequence **S**.

Size of probabilistic subinterval is extended by variable **A(s)**, which initial value is 0.111..…11. Variables **A(sk)** and **A(sm)** are binary variables of size of probabilistic subinterval for more and less probable symbol.

This method is based on approximation of probability of less probable symbol by value of 2^{-Q}. Variable **C(s)** is obtained by gradual encoding of symbols of binary variable **S**. When all symbols of binary variable **S** are encoded we get two binary variables, **C(s)** and **A(s)**. Resulting arithmetic code is a binary value from interval <C(s), C(s) + A(s)). This value is made, so it contained the smallest possible amount of valid binary numbers.

2) Decoding

Procedure of decoding is described by equations 4.5 – 4.8.

$$A(sk) = A(s).2^{-Q} \tag{4.5}$$
$$A(sm) = < A(s) - A(sk) > \tag{4.6}$$
$$C(s) < A(sm) => y = m; C(sy) = C(s) \tag{4.7}$$
$$C(s) \geq A(sm) => y = k; C(sy) = C(s) - A(sm) \tag{4.8}$$

Input encoded sequence is decoded with gradual recursion of equations 4.5 – 4.8. Decoder must have information about length of encoded word. This information could be broadcasted in transmission channel separated from encoded sequence, or the length can be given in advance.

B. Binary state arithmetic encoding (BSAC) of binary images

In arithmetic encoding without using multiplications, it is necessary to know value of **Q** from expression2^{-Q}.

This value must be given for each symbol of sequence. This information is contained in the model [5]. Model is practically an estimator estimating probability of actual symbol occurrence. JBIG algorithm is used For binary images. Ten points template of JBIG algorithm is on figure 4.1.

		c9	c8	c7		
	c6	c5	c4	c3	c2	
	c1	c0	O			

Figure. 4.1. Ten points template of JBIG algorithm.

State **S(y)** of actual point (indicated by circles) is estimated by equation 4.9.

$$S(y) = \sum_{i=0}^{n-1} ci.2^i \tag{4.9}$$

Where **n** is a number of elements in template and **y** is an actual point. After testing the whole image, it is evaluated how many times a particular state occurred and which value actual point had. After that, the most frequent occurred value is determined Finally, conditional probability will be calculated (eq. 4.10.) of point with value “1” or “0” in individual states.

$$p(c = v \mid S_k) = p(S_{v,k}) = \frac{n_{v,k}}{n_k} \tag{4.10}$$

$\mathbf{S_k}$ is value given by equation 4.9 and mean value of actual pixel. $\mathbf{S_{v,k}}$ is state of $\mathbf{S_k}$ with value of pixel equal to **v**. $\mathbf{n_k}$ is a number of states $\mathbf{S_{v,k}}$. The final model of image is formed by the resulting probabilities. Conditional probability $\mathbf{S_{v,k}}$ are approximated by value 2^{-Q}.

V. EXPERIMENTAL RESULTS

Experimental images LENA and BABOON fig. 5.1 and 5.2, respectively, were transformed by DCT in to images of spectral coefficients fig. 5.3 and 5.4. After that, they were quantized with two scales. Lower value of scale is set so that the IDCT (inverse discrete cosine transform) subjective quality would by approximately the same as the original images. High level is set to maximum possible value therefore 32. On fig. 5.5 and 5.6 images after quantization and IDCT are showed.

Fig. 5.1 Image Lena

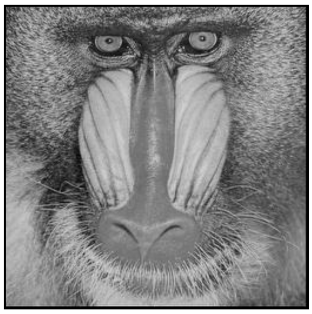


Fig. 5.2 Image Baboon

Fig. 5.3 DCT image Lena

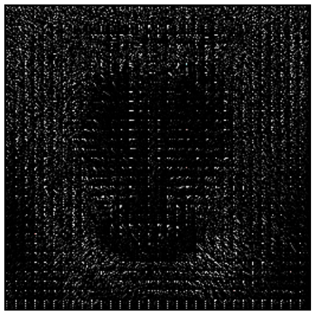
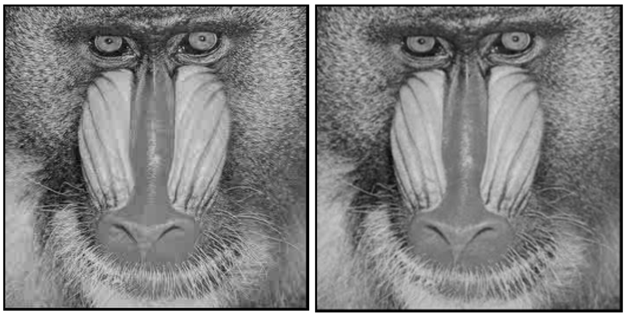


Fig. 5.4 DCT image Baboon

a) b)

Fig. 5.5 IDCT image Lena a) MK = 15 b) MK = 32



a) b)
Fig. 5.6 IDCT image Baboon a) MK = 17 b) MK = 32

Table 1 shows middle length of code words of images encoded by BSAC and HC and also differences between middle lengths of code words.

TABLE I
DIFFERENCE BETWEEN MIDDLE LENGTH OF CODE WORD OF HC AND BSAC

Image			Middle length of code word		Difference between HC and BSAC	
	Image	Scale	BSAC bit/sc	HC bit/sc	Reduction of code word	Extension of code word
RBP	LENA	15	1,7679	1,8800	0,1121	-
		32	1,1489	1,4680	0,3192	-
	BABBON	17	2,3959	2,2130	-	0,1829
		32	1,6004	1,6100	0,0096	-
ABP	LENA	15	1,7982	1,8800	0,0818	-
		32	1,2082	1,4680	0,2598	-
	BABBON	17	2,4980	2,2130	-	0,2850
		32	1,7526	1,6100	-	0,1426

From the table, it can be concluded that maximum saving is achieved by using BSAC aplicated on RBP for images with low frequency content therefore for an image with lot of monotone areas.

For an image with high frequency content, it is better to use conventional encoding, therefore Huffman encoding.

BSAC implicated on the image decomposed to ABP is possible, but it gets lower compression like application on RBP. From the table, it is also evident that with the increase of MK there is increasing compression in both encoding methods compression. This increase is stronger in BSAC than in HC. This is illustrated in table 2.

TABLE II
INCREASING OF COMPRESSION FOR BSAC AND HC WITH INCREASING OF SCALE OF QUANTIZATION

		Middle length of code word			Increasing of compression		
		BSAC		HC bit/sc	HC	BSAC	
Image	Scale	RBP	ABP			ABP	RBP
LENA	15	1,7679	1,7982	1,880			
	32	1,1489	1,2082	1,468	0,412	0,590	0,6191
BABOON	17	2,3959	2,4980	2,213			
	32	1,6004	1,7526	1,610	0,603	0,7454	0,7955

VI. CONCLUSION

In the article, we discussed the modification of standard JPEG using arithmetic binary-state encoding applied on bit-planes of DCT image instead common used Huffman code. With experiments we arrive at the results which suggest that increased degree of compression for images with low frequency content. Conversely, this encoding doesn't achieve satisfactory results for images with high frequency content. It was found out that it is better to encode the real bit-plane of DCT image than absolute bit-plane. We achieved the best results using BSAC on image quantized with high MK. It can be inferred, that this modification of JPEG can be used, but it is necessary to add other algorithms which detect level of details of image and decides what kind of encoding to use. Another disadvantage is the necessary decomposition of the image to bit-plane and encoding of these using more difficult arithmetic code and a need to determine actual state and conditional probability. These disadvantages cause increased demands on hardware performance.

REFERENCES

[1] Mihalík,J.: Image Coding in Videocommunications. Mercury-Smékal, Košice, 2001. ISBN 80-89061-47-8 (244 pages). (In Slovak)

[2] Kováč O.: Transformačné kódovanie obrazov. Bakalárska práca FEI TU Košice, 2009.

[3] Mihalík,J.-Štefanišin,R.: Entropické kódovanie v štandardných videokodekoch.Acta Electrotechnica et informatica, Vol.5, No.2, 2005, s.43-50, ISSN 1335-8243.

[4] Gladišová,I.-Mihalík,J.-Zavacký,J.: Bezstratová kompresia obrazu pomocou stavového binárneho aritmetického kódovania jeho bitových rovín. Slaboproudý obzor, roč. 62, č.1-2, 2006, str.18-21, ISSN 0037-668X.

[5] Gladišová,I. - Mihalík,J.:Stavové aritmetické kódovanie binárnych obrazov. Acta Electrotechnica et Informatica, Vol.3, No.3, Košice, 2003, p.36-44, ISSN 1335-8243.