A Non-uniform Image Compression Using Genetic Algorithm

M.R Bonyadi\textsuperscript{1}, E.Dehghani\textsuperscript{2}, and Mohsen Ebrahimi Moghaddam\textsuperscript{3}
\textsuperscript{1,2}Electrical and Computer Engineering Deptment
Shahid Beheshti University, Teharn, Iran
Phone: (98) 21-22431728 Fax: (98)21-22431804 E-mail: m.moghadam@sbu.ac.ir

Keywords: Image Compression, Genetic Algorithm, JPEG Method, Zonal Mask

Abstract - Most of the image compression standards are based on frequency domain transforms that is followed by quantization. Therefore, the ways to threshold a transformed block of image are the interest of many researchers in order to find a good balance between image quality and compression ratio. In this paper we proposed a lossy compression method based on JPEG which applies an adaptive threshold for each block. The threshold mask function is determined by using Genetic Algorithm. The proposed Genetic Algorithm uses Gray Level Difference (GLD) as a feature. The experimental results showed that proposed method had higher PSNR than some related works and it is comparable with JPEG2000.

1. INTRODUCTION

Nowadays, the public interest in using digital camera and digital images has increased. So the need for making images more compressed to be easy to transport is felt [2]. Not surprisingly, extensive researches have been carried out to make the compression methods more efficient because uncompressed multimedia data require considerable storage capacity and need high transmission bandwidth. Thus, image compression is a very important factor for better utilization of network bandwidth and computer storage [3]. Most of the image compression methods are lossy approaches, and researchers are interested to improve image quality while increasing compression ratio.

For improving the DCT based approaches, several attempts have been made. In [4], a method based on Fractal in combination with DCT has been proposed. The concept was based on the ability of DCT to remove inter-pixel redundancies and the capability of fractal transforms to capitalize long-range correlations in the image. The authors claimed that the hybrid coder performs optimally. In particular, they used the fractal transform to yield an optimal segmentation. They also showed that the fractal transforms are more efficient for representing high frequency information in compare with DCT. However, the DCT can properly represent the low frequency information.

Fuzzy clustering for compressing images is addressed in [5]. The proposed method in [5] works based on variable block size approach using vector quantization and its local fractal dimension. The approach is based on the analysis of the complexity of regions of an image which was measured by LFD. They implemented variable block size using LFD values and constructed a codebook for a vector quantization. The proposed algorithm invoked the fuzzy clustering algorithms such as FCM (Fuzzy c-mean) and fuzzy k-mean.

Another paper utilizes fuzzy idea for compressing images is presented in [6]. This method deploys the Fuzzy ART network. This network learns and memorizes the characteristics of the uninteresting areas and/or interesting areas selected by the bottom-up saliency map (SM) model. Essentially, the method compressed interested areas by lossless encoding and uninterested areas by lossy encoding. This algorithm improves the performance of JPEG standard just slightly.

Moreover, the neural network based algorithms for image compression has attracted a lot of attentions. One of the most recent works has been presented in [7].

In this paper, we propose a JPEG based compression method which applies an adaptive threshold for each block. The threshold mask function is determined by using Genetic Algorithm. The proposed Genetic Algorithm uses Gray Level Difference (GLD) as a feature. The experimental results showed that the image PSNR and the compression ratio of proposed method is better than most of the recently presented methods. In addition, the proposed method can adjust its parameter with regards to the user specified compressed image size.

The rest of paper is organized as follow: in section 2 some basic principles and fundamental concepts which deployed in the proposed method are presented. Section 3 consists of proposed method and its simulation results. Finally in section 4 we conclude our conclusion.

2. BASIC PRINCIPLES AND FUNDAMENTALS

In this section genetic algorithm (GA) and DCT (discrete cosine transform) as the most important principle and background are described.

2.1 Genetic algorithm

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection [8]. It is categorized as a stochastic algorithm. GA is especially efficient when the search space of a problem has very rough landscape riddled with many local optima [9]. It overcomes the problems occur in some other search algorithms such as Hill Climbing. Reaching a Flat area is the most common problem in this algorithm and related ones. The most important components of GA are as follow:
Fitness function: It defines what improvement means and assigns a quality measure to a chromosome and evaluates it.

Initial population: its role is to hold (represent) possible solutions.

Selection: it could be done in two ways: Fitness proportional selection and rank-based selection. In the first one the selection probability depends on the absolute fitness value of the individual(solution) compare to other individual and in the latter it preserve the constant pressure by sorting the population on the bases of fitness and then allocating selection probability according to rank [8].

Variation operators: these operators create new individual (solution) by selecting one or more individual from population.

Mutation: this operator use only one parent and create one child by applying some kind of randomized change to the representation.

Crossover: is a process whereby a new individual solution is created from the information contained within two or more parent solution [8].

2.2 Discrete cosines transform (DCT) and Standard JPEG

Discrete Cosine Transform (DCT) can be applicable in different area of image processing such as image transformation, feature extraction, and image compression [10]. Some compression methods like JPEG perform compression by transforming images into the spatial-spectral domain using DCT. In the JPEG image compression algorithm, the input image is divided into 8-by-8 blocks, and the two-dimensional DCT is computed for each block (FDCT). The DCT coefficients are then quantized, coded, and transmitted. The JPEG receiver (or JPEG file reader) decodes the quantized DCT coefficients, computes the inverse two-dimensional DCT of each block (IDCT), and then puts the blocks back together into a single image [11]. For typical images, many of the DCT coefficients have values close to zero; these coefficients can be discarded without seriously affecting the quality of the reconstructed image. In the following equation, DCT coefficients denoted by the variables $S_{xy}(0 < x, y < 7)$. The coefficient $S_{00}$ is known as the DC coefficient because it represents the average value of the block, whereas remaining coefficients are known as the AC coefficients. The FDCT and IDCT are defined as below:

$$S_{uv} = \frac{1}{4} C_x C_y \sum_{x=0}^{7} \sum_{y=0}^{7} S_{xy} \cos \left( \frac{(2x + 1)\pi u}{16} \right) \cos \left( \frac{(2y + 1)\pi v}{16} \right) \tag{1}$$

$$S_{xy} = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} C_u C_v S_{uv} \cos \left( \frac{(2x + 1)\pi u}{16} \right) \cos \left( \frac{(2y + 1)\pi v}{16} \right) \tag{2}$$

Where $C_x$ and $C_y$ are defined by:

$$C_v = \begin{cases} \sqrt{2} & \text{for } v = 0 \\ 1 & \text{for } v \neq 0 \end{cases} \tag{3}$$

Discarding the values in the DCT of a block can be performed by using a zonal mask [12]. A zonal mask is constructed by placing a 1 in the locations with maximum variance and a zero in all other locations. By multiplying the transformed block into its zonal mask, the locations with less variances will be discarded.

3. Proposed approach

In this section the proposed method is introduced in detail. The Genetic Algorithm is applied to determine the threshold mask function. Our goal is to increase the PSNR of compressed image while desired size is met.

3.1 Parameters

As it mentioned in section 2.2, the Zonal mask would be needed to perform JPEG compression. The number of the 1’s in the zonal mask (D) corresponds with Bpp (Bit per pixel). Therefore, the position of 1’s in the Zonal mask should be adjusted such that it has fewer effects on the transformed coefficients that carry the most image information. The number of 1’s in the Zonal mask can be modeled using a multivariate function:

$$D = f(S_1, S_2, ..., S_n) \tag{4}$$

Where $S_i$ is a variable that has a role in determining 1’s in the Zonal mask (D). These variables can be a feature such as variance, standard deviation (STD), Detrended Fluctuation Analysis (DFA), any Fractal dimensions [13] or any other fluctuation parameters. The function $f$ is an operator which correlates those parameters to each other. Here, we used a variable that is maximum gray level difference (GLD) in corresponding area. Therefore, It is obtained in each block independently.

To clarify this feature, consider “Fig.1.” depicting the gray level of a 32*32 block of a normalized image. The GLD is obtained by subtracting the max pixel value from min pixel value (0.8824-0.1843=0.698).

Regarding to image normalization the interval of its value is [0, 1].

![Fig. 1. The value of GLD in a 32 * 32 block is 0.8824-0.1843](image)

Let us have a closer look at the behavior of this feature by applying various Zonal masks for compression. “Fig.2.” demonstrates the relation between a block of a sample image PSNR v.s different Zonal masks which their 1’s are different. The block GLDs are different. Also, the LSE (Least square error) line is shown.

According to the “Fig.2.” we can infer that:

This shows that as D (number of 1’s in the Zonal mask) varies there is less effect on the PSNR of blocks with small GLD and more effect on the PSNR of blocks with big GLD.
The proposed method employs GA. Population is randomly proposed as fitness function: the following function was divided to the number of gens, then by using the gen no and the range of GLD is used. However, the range of GLD is values correspond to D values. To interpret each gen value, them represents a mapping point from GLD to D. The gen genome consists of some positive integers such that each of them represents a mapping point from GLD to D. The gen values correspond to D values. To interpret each gen value, the range of GLD is used. However, the range of GLD is divided to the number of gens, then by using the gen no and the gen value the mapping point is estimated. A sample of genome is illustrated in “Fig.3.(c)”

At second step, the fitness function as the most important part of GA should be designed. The following function was proposed as fitness function:

\[
\text{minimize: } \text{Fit}(C_i) = \begin{cases} 
1 & \text{if } \text{PSNR} < l \\
\text{Size} & \text{if } \text{PSNR} \geq l
\end{cases}
\]  

(5)

Where \( C_i \) and \( l \) are the \( i \)th chromosome and the requested PSNR, respectively. The GA tries to take this PSNR while decreases the size. Because, the genotype for the GA is considered as some points on the GLD-D plane, the GA could find the proper mapping points. To determine the 1’s in the Zonal mask; GLD in each block is calculated and the points that are estimated by GA are used.

Because the results of GA are discrete points, to determine D value for intermediate points Spline interpolation is used.

The fitness progression of the GA for a sample image is shown in “Fig.3.a”. The final mapping generated by GA and Spline interpolation is shown in “Fig.3.b”. “Fig. 3(a)” shows that the GA converges in 80 epochs. We set GA parameters as follow: the genome contains 8 points, population consists of 10 genome, scatter crossover with \( P_c = 0.9 \) as proved in [8] was used, and random mutation with \( P_m = 0.2 \) is employed. Population is randomly initialized; the selection operator is based on the roulette wheel algorithm.

The results of the compressed Lena image by using the proposed method are listed in “Table 1”.

Moreover, to make the method comparable with related works; we changed the fitness function and the following one was used. It finds the maximum value of PSNR where the size is not more than \( K \):

\[
\text{minimize: } \text{Fit}(C_i) = \begin{cases} 
1 & \text{if } \text{Size} > K \\
\frac{1}{\text{PSNR}} & \text{if } \text{Size} < K
\end{cases}
\]  

(6)

The desired image size can be specified by this equation and GA tries to get this size while increasing PSNR. The results of this minimization for a sample image are shown in “Fig.4.” where setting of GA parameters is as mentioned setting.

The size of the Lena image compressed by the proposed method when (6) is used as fitness function was equal to 1.44 Bpp and its PSNR was 32.9dB. This shows this Fitness function decreased a little the PSNR of result image but the ability of specifying size is added to the method. To simulate the algorithm (6) was used as Fitness.

“Fig.5.” shows Lena image when GA and JPEG methods was used to compress it. Theses results do not include Huffman coding and RLC.

### Table 1: The results of JPEG in comparison with proposed method

<table>
<thead>
<tr>
<th>Picture</th>
<th>JPEG</th>
<th>GA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>PSNR(dB)</td>
<td>Size(Bpp)</td>
</tr>
<tr>
<td>Lena</td>
<td>29.921</td>
<td>1.44</td>
</tr>
</tbody>
</table>

Also, “Fig.2.” shows that the PSNR of blocks with small GLD is inherently more than blocks with big GLD.

Therefore we introduced the threshold mask function based on GLD and gray level variance of a image block.

### 3.2 GA optimization and simulation results

The goal of each compression algorithm is optimizing the PSNR of the compressed image while high compression ratio is obtained. Therefore, the problem of compression can be considered as an optimization problem. Hence, the Genetic algorithm can be used to solve this problem. As we mentioned in section 2.1 other search space algorithms such as Hill Climbing can be used here but GA has more benefits because the problem of flat areas is solved in this algorithm.

At first, we should design genotypes for GA [8]. The genome consists of 10 genome, scatter crossover with parameters as follow: the genome contains 8 points, shows that the GA converges in 80 epochs. We set GA and simulate the algorithm (6) was used as Fitness.
“Table 2” demonstrates the PSNR and proposed GA method in comparison with standard JPEG and JPEG2000.

“Fig.6.” shows the Lena image compression results (after Huffman coding) by considering various sizes using the proposed GA method, JPEG, FLGA1/FLGA2 [5], Fractal/DCT [4], and JPEG2000.

4. CONCLUSION AND FUTURE WORKS

In this paper an image compression method based on standard JPEG was proposed. The procedure of an adaptive threshold mask function using Genetic Algorithm was presented. As a contribution of our work, a feature which analyzes the fluctuations in a block was used in order to determine the threshold mask function. Although the results of proposed method were not as good as JPEG2000 which is based on wavelet-transforms, but it is still promising in regard to most of the presented methods and even comparative with JPEG2000. In future we are going to use the proposed algorithm in embedded systems such as sensor networks.

REFERENCES

[1] K. Sayood, Introduction to Data Compression, Morgan Kaufmann Publisher, Boston, 2000