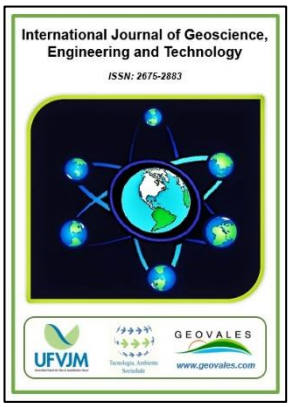
International Journal of Geoscience, Engineering and Technology
ISSN: 2675-2883
DOI:
Volume 8 – Nº 1 – Year: 2023 – pp.80-87



Received:
October 18, 2023

Accepted:
October 30, 2023

Published:
October 27, 2023

Reducing the block effect in color images (RGB) using genetic algorithms

Daniel Moraes Santos[1]

[1] Federal University of the Jequitinhonha e Mucuri Valleys, Teófilo Otoni, Brazil.

Email address
daniel.moraes@ufvjm.edu.br (Daniel Moraes Santos)

Abstract

The block effect is a typical effect in reconstructed images that have been encoded by a block-based discrete cosine transform (DCT). In highly compressed images and videos, block effects are easily noticeable as a discontinuity between relatively homogeneous regions. The JPEG standard has long been used as a reference algorithm for compressing monochrome images, this compression could also be applied separately to color image components (RGB). The purpose of this article is to present a method for generating quantization tables for the JPEG algorithm, which reduces the blocky effect presented by the JPEG standard. These quantization tables were found by applying the genetic algorithm technique, which is based on the functioning of genetic selection in humans. In other words, through the crossing of chromosomes, mutation, and natural selection of the most suitable results.

Keywords: Block Effect, Genetic Algorithm (GA).

1. Introduction

The block-based Discrete Cosine Transform (B-DCT) has been adopted in most image and video compression standards, including JPEG (*JPEG* ,1990) and MPEG (Coding of Moving Pictures and Associated Audio: Recommendation, by ITU, 1993), because it has three advantages: 1) good energy compression; 2) low computational cost and 3) easy hardware implementation (JAIN, 1989). The B-DCT scheme takes advantage of the spatial correlation property of images to divide images into 8x8 blocks, transforming each block from the spatial domain to the frequency domain using the discrete cosine transform (DCT). The block effect is located at the edges of the images, where the highest frequency components of the image are located (Pearson and Whybray, 1984).

For an encoding using B-DCT, the high frequency DCT coefficients tend to be removed because of the coarse quantization of high frequencies during encoding. 8x8 blocks are coded independently, the reconstructed image may generate discontinuity along block boundaries, simply referred to as block artifact.

Many algorithms have been proposed to reduce block artifacts. These algorithms can be grouped into two categories. One is the use of different coding schemes, such as the interleaved block transform (Pearson and Whybray, 1984; Farrelle and Jain, 1986), the lapped transform (Hinman, Bernstein and Staelin, 1984; Malvar and Staelin, 1989) and the combined transform (Zhang, Pickholtz and Loew, 1993). The other is postprocessing the reconstructed image (Zakhor, 1992; O'rourke and Stevenson, 1995; Yang, Galatsanos and Katsaggelos, 1993; Shen and Kuo, 1999; Park and Lee, 1999).

The algorithm proposed in this article is included in the first category mentioned.

Currently, reducing this effect becomes very important because there is an increasing need for the reconstructed image to be the same as the original image. This is because these images can be used in medical diagnoses, where precision and very good visualization of the reconstructed image are required.

The three stages of the JPEG compression process (Sherlock, Nagpal and Monro, 1994; Wallace, 1991), in Figure (1), are made up of the

discrete cosine transform (DCT), quantization of the DCT coefficients and entropy coding. Figure (2) presents the stages of the inverse process or decompression: entropy decoding, dequantization and inverse discrete cosine transform (IDCT). It is important to mention that in this article stages 3 (encoding) and stage 1 (decoding) are not implemented, as the focus of this work is to generate a quantization table that reduces the effect of blocks.

As already mentioned, in the JPEG encoder the image is subdivided into 8x8 blocks, subimages of 64 pixels. Blocks are processed from left to right, top to bottom. Each of the subimage pixels are subtracted by half the highest gray level, that is, 2(n-1) (Gonzalez and Woods, 2000).

Following this procedure FDCT 2-D is applied to each one of the blocks. After this, the 64 DCT coefficients are quantized through a Quantization-T of 64 coefficients, where the quantization operation is implemented as a division of each DCT coefficient by its corresponding quantizer step size, followed by rounding it to the nearest integer (Wallace, 1991; Fan and Queiroz, 2002).

The JPEG baseline coder allows users to redefine the quantization table to control the compression ratio and the quality of the reconstructed image. Methods for determining the quantization tables are usually based on rate-distortion theory. These methods achieve better performance than the JPEG standard tables. However, the quantization tables are image-dependent and the complexity of the encoders is rather high (Fan and Queiroz, 2002; Chang, Wang and Lee, 2002; Fan and Queiroz, 2003; Vander Kam, Wong and Gray, 1999).

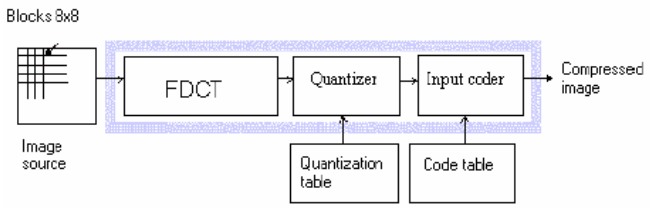


Figure 1 – Simplified JPEG coder.

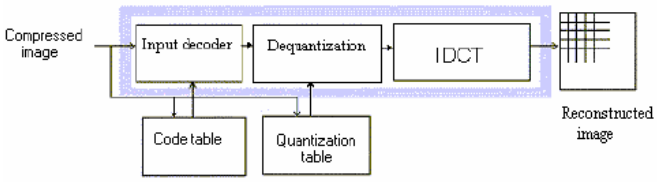


Figure 2 – Simplified JPEG decoder.

The gray scale images are formed by pixels whose values represent the amount of brightness, while for the color images three values are necessary for specification of the pixel characteristics, that is, we have in the color coding (YCrCb) the luminance component Y, the blue chrominance component Cb and the red chrominance Cr, the most common coding method used by the JPEG algorithm. If the image is already coded using the RGB system it will be transformed to the YCrCb system. The purpose of this article is to compare the performance of the proposed quantization tables obtained by GA with the standard JPEG tables, and to do this it is not necessary to filter the chrominance components, commonly used to reduce spatial bandwidth. The luminance of images and chrominance are represented using 8 bits/pixel, with values, which range from 0 to 255, in other words, 256 possible values.

2. Architecture of The Proposed Algorithm and Quantization Table

Figure (3) clarifies the compression process of the standard JPEG and the proposed architecture. In this figure, the FDCT is applied to three matrixes (YCbCr), later by use of the Quantization-T, generated by the Genetic Algorithm the quantization of the subimages takes place.

In this article two types of quantization table were used: Quantization-T for luminance Table (1) and the other for chrominance Table (2) both JPEG standard.

Table 1 – Standard Quantization-T 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

Table 2 – Standard Quantization-T for chrominance.

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

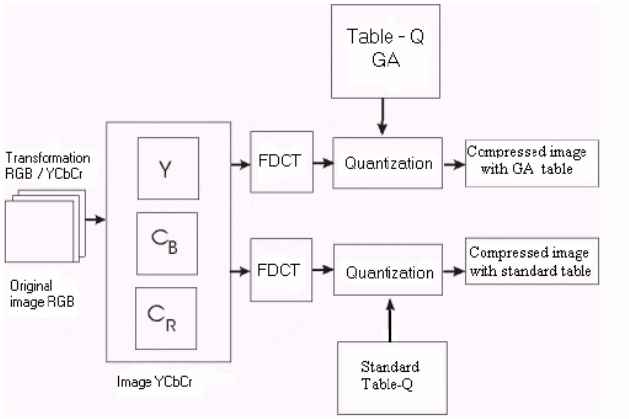


Figure 3 – Architecture of the proposed and standard compression system.

After the quantization of the DCT coefficients there are some stages which follow such as: separation of the DC coefficient from the AC coefficients, reorganization of the coefficients using the zigzag sequence and entropy encoding, which are not subjects of this work. The focus is to find a more efficient Quantization-T than the one used by the JPEG standard.

3. Genetic Operators

Genetic algorithms (GA) are search algorithms (Wallace, 1992) that are based on the reproductive mechanism or be it in genetics combination and natural selection. The results of the genetic algorithm come from an initial population of points, where it follows several probabilistic decision rules, which are derived from human genetics.

Where some points of this initial population have a greater probability of surviving. This initial population generates other individuals that have a greater probability of "fitness" than the first individuals, so each generation of populations tends to have optimal evolution.

This article works with two operators, "crossover" and "mutation". The first is responsible for the genetic change between a pair of chromosomes, where genetic information is exchanged between them. The second is responsible for the convergence of results, where this genetic operator does not exchange information with anyone, it simply inverts information (states, bits, etc.) of a given chromosome (Goldber, 1989).

The Software developed is based on the GA theory and is composed of 6 initial parameters:

- The number of chromosomes (pop) of a given population;
- The number of generations;
- The original image;
- The crossover rate (px);
- The mutation rate (pm);
- The number of bits/pixel.

Three of six initial parameters are fixed, the crossover rate, the mutation rate, which more commonly has the values 0.6 and 0.1 and thirdly 8 bits/pixel number.

If the number of chromosomes is 16 and 32, the algorithm will generate randomly 16 and 32 Quantization-T (which include the JPEG standard table) that will serve as the initial population. The number of generations is 10, 20 and 30.

4. Algorithm Operation

The proposed algorithm works with a quality variable, which is the peak signal-to-noise ratio (PSNR), where the higher the PSNR, the smaller the influence of noise on the reconstructed image, thus improving the block effect. If the algorithm is working with 16 chromosomes, a vector of 16 positions will be created, where these positions represent the mean squared errors (MSE) between the original image and the reconstructed image. With the MSE it is possible to find the PSNR through Equations (1) and (2).

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(x_{i,j} - \hat{x}_{i,j} \right)^2 \qquad (1)$$

Where M is the number of lines and N the number of columns in the image.

$$PSNR = 10 * \log \frac{\left(2^n - 1\right)^2}{MSE} \, dB \quad (2)$$

The final result is based on a fitness function, which is found by multiplying the PSNR values (aptSNR) with the DC coefficients (aptDC). This step is necessary to be able to find the total fitness function of the population, which can be found by summing the individual fitnesses.

Therefore, it is possible to find the probability function, where it indicates the probability of each string being chosen, using Equation (3).

$$prob(i) = \frac{apt(i)}{\sum_{i=1}^{pop} apt(i)} \quad (3)$$

The probability vector prob(i) is made up of "pop" (pop = 16). With this, it is possible to find the accumulated probability vector CS(i), where this vector is obtained by adding each of the positions of the probability vector above (prob(i)), as follows in Equation (4).

$$CS(i) = \sum_{k=1}^{i} prob(i) \quad (4)$$

After creating the accumulated probability vector CS(i), the algorithm generates a random T value, through a uniform density probability function (rv), where this function is limited between the values [0,1]. This operation is repeated "pop" times. With the value of T, the decision rule applies: if CS(i-1) < T < CS(i), the string is selected.

This method described for selecting the most suitable strings is called the roulette wheel method, where this roulette wheel is divided into "pop" areas, where each area is intended for a possibility, the larger the area the greater the probability of a string being selected.

At the end of this step, the algorithm generates an array of PS cells (parent selection) of dimension pop x 1, where each position in this vector represents an 8x8 matrix.

5. Natural Selection

The genetic algorithm is based on an initial population (chromosomes), where genetic operators are applied to this initial population, creating a new population.

Natural selection begins by generating a random number to select the most suitable chromosome, this process is repeated a second time to select the other chromosome that will pair with the first selected. This process is repeated until (pop/2) pairs of chromosomes are found, where pop is the number of chromosomes in the initial population.

A random value (Pcruz) is generated between [0,1] for each pair of chromosomes, then the decision rule is executed, if px > Pcruz, the crossover is performed, where px is the crossover rate.

For the mutation stage, a random value (Pmut) is also generated to apply in the decision rule. If pm > Pmut, a mutation occurs in the selected chromosome and gene.

At the end of these steps, a vector with pop positions is generated, where each position represents a possible 8x8 quantization table, with 64 positions.

Based on the tables generated after the genetic operations and also on the initial population tables, the MSE mean squared error is calculated for the two quantization table vectors.

This makes it possible to calculate the peak signal-to-noise ratio (PSNR) for the two vectors. Therefore, the PSNR result is applied to find the fitness function of each table by multiplying by their DC coefficient, as indicated in Equation (5).

$$S_{apt}(i) = aptSSNR(i) * aptSDC(i) \quad (5)$$

Comparing the coefficients from the parents' table (initial population) with the coefficients from the children's table (population after genetic operators), the table with the highest values is selected as the best quantization table.

6. Results

The proposed genetic algorithm was applied to an RGB image, 256 x 256 pixels, called Lenna, in Figure (4).

Figure 4 – Original Image.

This original (RGB) image was decomposed into three images each (YCbCr), luminance, b-chrominance and r-chrominance, respectively in Figures (5), (6) and (7).

Figure 5 – Original Luminance Image.

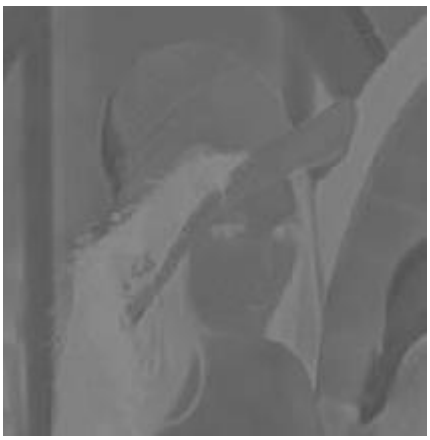


Figure 6 – Original b Chrominance Image.

Figure 7 – Original Chrominance Image.

The genetic algorithm proposed by this article was applied to the three images decomposed from the original RGB image, performing the quantization using the quantization table found by the GA. To make it possible to compare with the JPEG standard table, the images in question were also quantized with the JPEG quantization table.

It is possible to notice the reduction of the block effect in the images quantized by the genetic algorithm, as can be concluded from the figures (8) and (9).

Figure 8 – AG Luminance Image.

Figure 9 – JPEG Standard Luminance Image.

Visually, it is possible to notice the decrease in the block effect in Figure (8) dequantized by GA in relation to 9 which was dequantized by the standard JPEG Algorithm.

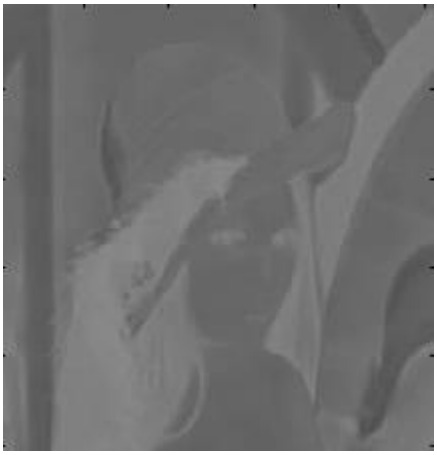


Figure 10 – Chrominance image b AG.

Figure 11 – Chrominance Image b JPEG Standard.

It can be noted that the edge effect for the proposed genetic algorithm visually is much smaller than that of the JPEG standard. This is much clearer for b-chrominance images. It was also very clear in the r-chrominance images, as can be seen in Figures (12) and (13).

Figure 12 – Chrominance Image r AG.

Figure 13 – Chrominance Image r JPEG Standard.

The GA "b" and "r" chrominance images showed better performance in reducing the block effect in relation to the luminance of the algorithm in question, as they presented a much higher peak signal-to-noise ratio (PSNR) than the image luminance.

7. Conclusion

Analyzing the images presented above, we can see a reduction in blocking artifacts in the images that use the compression tables generated by the Genetic Algorithm compared to the images that used the standard JPEG compression tables. The difference is much more evident in the chrominance images, as the images presented by the Genetic Algorithm have a higher peak signal-to-noise ratio than the images presented by the standard JPEG, but the luminance images from the Genetic Algorithm also showed a reduction in chrominance artifacts. blocking compared to standard JPEG images. But as the peak signal-to-noise ratio of the AG images did not show as large a difference as in the chrominance images, this difference is visually smaller.

References

Chang, L.W., Wang, C.Y. and Lee, S.M., 2002. Designing JPEG Quantization Tables Based on Human Visual System. In: *Proceedings 1999 International Conference on Image Processing.* https://doi.org/10.1109/ICIP.1999.822921

Fan, Z. and Queiroz, R.L., 2002. Maximum Likelihood Estimation of JPEG Quantization Table in the Identification of Bitmap Compression History. In: *Proceedings 2000 International*

Conference on Image Processing. https://doi.org/10.1109/ICIP.2000.901117

Fan, Z. and Queiroz, R.L., 2003. *Identification of Bitmap Compression History: JPEG Detection and Quantizer Estimation*. IEEE Transactions on Imaging Processing, 12(2), pp.230-235. https://doi.org/10.1109/TIP.2002.807361

Farrelle, P. and Jain, A., 1986. Recursive block coding - A new approach to transform coding, *IEEE Transactions on Communications*, 34(2), pp. 161–179. https://doi.org/10.1109/TCOM.1986.1096509

Goldberg, D.E., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. 1st ed. Addison-Wesley Professional.

Gonzalez, R.C. and Woods, R.E., 2000. *Processamento Digital de Imagens*. 1st ed. São Paulo: Edgard Blucher.

Hinman, B., Bernstein, J., and Staelin, D., 1984. Short-space Fourier transform image processing. In: *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 481-484. https://doi.org/10.1109/ICASSP.1984.1172374

ITU – International Telecommunication Union, 1993. *Coding of Moving Pictures and Associated Audio: Recommendation H.262: ISO/IEC 13 818*, ISO / IEC / JTC1 / SC29, 1993.

Jain, A.K., 1989. *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall.

JPEG Technical Specification, Revision 8, ISO / IEC / JTC1 / WG8, JPEG Group, 1990.

Malvar, H.S. and Staelin, D.H., 1989. *The LOT: Transform coding without blocking effects*. IEEE Transactions on Acoustics, Speech, and Signal Processing, 37(4), pp. 553-559. https://doi.org/10.1109/29.17536

O'Rourke, T. P. and Stevenson, R. L., 1995. *Improved image decompression for reduced transform coding artifacts*. IEEE Transactions on Circuits and Systems for Video Technology, 5(6), pp. 490-499. https://doi.org/10.1109/76.475891

Park, H.W. and Lee, Y. L., 1999. *A postprocessing method for reducing quantization effects in low bit-rate moving picture coding*. IEEE Transactions on Circuits and Systems for Video Technology, 9(1), pp. 161-171. https://doi.org/10.1109/76.744283

Pearson, D. and Whybray, M., 1984. *Transform coding of images using interleaved blocks*. IEE Proceedings F (Communications, Radar and Signal Processing), 131(5), pp. 466-472. https://doi.org/10.1049/ip-f-1.1984.0072

Shen, M. Y. and Kuo, C.-C. J., 1999. Real-time compression artifact reduction via robust nonlinear filtering. In: *Proceedings 1999 International Conference on Image Processing*, 2, pp. 565-569. https://doi.org/10.1109/ICIP.1999.822958

Sherlock, B. G., Nagpal, A. and Monro, D. M., 1994. A model for JPEG Quantization, *International Symposium on Speech, Image Processing and Neural Networks*, pp. 176-176.

Vander Kam, R.A., Wong, P.W. and Gray, R.M., 1999. *JPEG - Compliant Percentual Coding for a Grayscale Image Printing Pipeline*. IEEE Transations on Image Processing, 8(1), pp. 1-14. https://doi.org/10.1109/83.736675

Wallace, G.K., 1991. *The JPEG Still Picture Compression Standard*. Communications of the ACM, 34(4), pp. 31-34. https://doi.org/10.1145/103085.103089

Wallace, G.K., 1992. The JPEG Still Picture Compression Standard. In: *IEEE Transactions on Consumer Electronics*, 38(1), pp. 18-34.

Yang, Y., Galatsanos, N.P., and Katsaggelos, A.K., 1993. *Regularized reconstruction to reduce blocking artifacts of block discrete transform compressed images*. IEEE Transactions on Circuits and Systems for Video Technology, 3(6), pp. 421-432. https://doi.org/10.1109/76.260198

Zakhor, A., 1992. *Iterative procedures for reduction of blocking effects in transform image coding*. IEEE Transactions on Circuits and Systems for Video Technology, 2(1), pp. 91-95. https://doi.org/10.1109/76.134377

Zhang, Y., Pickholtz, R., and Loew, M., 1993. *A new approach to reduce the 'blocking effect' of transform coding (image coding)*. IEEE Transactions on Communications, 41(2), pp. 299-302. https://doi.org/10.1109/26.216502