PROGRESS IN COLOR, COLORANTS AND COATINGS

available online @ www.pccc.icrc.ac.ir

Prog. Color Colorants Coat. 2(2009), 95-101

Ministry of Science, Research and Technology
Institute for Color Science and Technology

A New Method for Color Gamut Mapping by Genetic Algorithm

S. Gorji Kandi[*1], M. Amani Tehran[2]

[1] Assistant Professor, Department of Color Imaging and Image Processing, Institute for Color Science and Technology, P.O.Box: 16765-654, Tehran, Iran.

[2] Associate Professor, Department of Textile Engineering, Amirkabir University of Technology, P.O. Box.15875-4413, Tehran, Iran.

ARTICLE INFO

Article history:
Received: 29-7-2009
Accepted: 2-12-2009
Available online: 5-12-2009

Keywords:
Color gamut mapping
Genetic algorithm
Image difference.

ABSTRACT

To reproduce an image, it is necessary to map out of gamut colors of the image to destination gamut. It is clear that the best color gamut mapping introduces the perceptually closest image to the original one. In this study, a new color gamut mapping is purposed by the aid of Genetic Algorithm (GA). The color difference between the original and mapped images based on S-LAB formula was chosen as fitness function. The proposed algorithm was applied in CIELAB color space and special genetic operators were developed to meet the aim of gamut mapping. To increase the rate of convergence and have a faster algorithm, one of the initial population chromosomes can be obtained from the result of clipping method. The results showed that the new method introduces smaller color difference between the reproduced and original images in comparison with the common clipping method. The other advantage of the genetic color gamut mapping is that any new criterion for color image difference can be easily used as a fitness function. In addition, by this method the final colors are not restricted to the gamut surface and they may be included into the gamut. Prog. Color Colorants Coat. 2(2009), 95-101. © Institute for Color Science and Technology.

1. Introduction

Color gamut of a color device or color media is an important parameter which plays an efficient roll in color reproduction. The need for color gamut mapping arises from the differences between the size, shape and location of gamuts of different imaging devices.

According to the definitions used by the CIE TC 8–03, "Color gamut mapping is a method for assigning colors from the reproduction medium to colors from the original medium or image". Several gamut mapping methods have been proposed from the basic clipping of colors onto the nearest point on the gamut surface, to complex transformation of color space specially based on

*Corresponding author: sgorji@icrc.ac.ir

the choice of lines along which the mapping is applied [1-8]. In recent years, new methods have been introduced to map input gamut to the destination one using neural network or considering gamut mapping as an optimization problem [9-10]. In addition, in the case of color constancy which new colors map to colors with least changes based on human color constancy, Interactive Genetic Algorithm (IGA) has been applied [11].

The optimal aim of gamut mapping is to preserve the appearance of the original image as much as possible. It can be reached by finding the perceptually most similar image among the set of reproducible images or minimizing the perceptually difference between the original and mapped images. There are two problems to achieve this goal. The first one is to have a good equation for evaluating the perceptual color image differences. In recent years, S-LAB [12] was proposed to this aid. The second is to find a computational method for mapping according to this criterion. In the present study, we introduced a novel color gamut mapping applying genetic algorithm to access the best mapped based on the similarity between the original and mapped images.

2. Experimental

As mentioned above, the goal of this paper is to apply genetic algorithm for color gamut mapping to achieve the least perceptually color image differences. For this purpose, suitable genetic parameters and operators should be introduced; these subjects are concluded in this section.

2.1. Finding out of gamut colors and clipping method

Before gamut mapping, the out of gamut colors should be identified. For this purpose it is necessary to have the gamut boundary (GBD). In this paper we use two gamuts which their boundaries were available and introduced by Morovic [13]. He applied Segment Maxima GBD method which gives the surface of a gamut in terms of a matrix containing the most extreme colors for each segment of color space [6]. This segmentation is usually carried out in spherical coordinates using the following formulae:

$$r = \left[\left(L^* - L_E^*\right)^2 + \left(a^* - a_E^*\right)^2 + \left(b^* - b_E^*\right)^2 \right]^{1/2}$$

$$\alpha = \tan^{-1}\left(\left(b^* - b_E^*\right) / \left(a^* - a_E^*\right) \right) \tag{1}$$

$$\theta = \tan^{-1}\left[\left(L^* - L_E^*\right) / \left(\left(a^* - a_E^*\right)^2 + \left(b^* - b_E^*\right)^2 \right)^{1/2} \right].$$

Where E is the center of the gamut and is set to [50 0 0] in most cases, r is the distance of a color from the center, α is the hue angle having a range of 360° and θ is the angle in a plane of constant α having a range of 180°. Figure 1 shows the parameters of spherical coordinates together with the segments of a GBD in CIELAB [6].

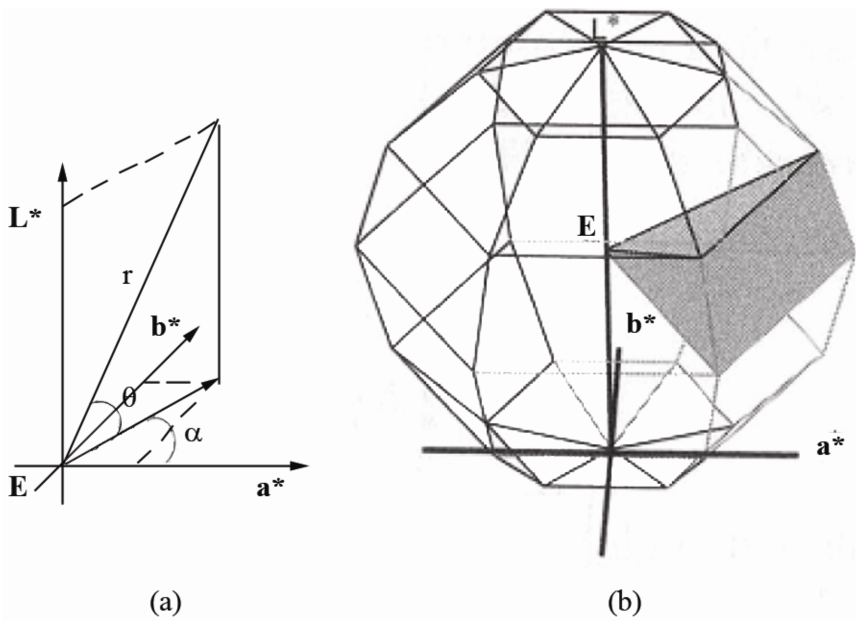

Figure 1: Overview of segment maxima GBD in CIELAB; (a) spherical coordinates (b) the segments of a GBD [6].

To find out whether a color (*c*) is out of gamut and to map it in case of being out of gamut, the proposed method by Morovic & Luo was applied. In this method, the LGB (the equation of the line (*l*) from the center along which the gamut boundary is defined) should be calculated for each color. It was done along lines having constant spherical angles (α and θ). To do it, three neighboring members of GBD by which the triangle is formed and intersects at *l* should be found. It can be simply done by considering the values of α and θ for GBD colors. The intersection of *l* with the plane determined by three neighborhoods is computed (*c'*). The r value of *c'* is calculated, if it is bigger than the r value of *c*, *c* is a out-of-gamut color and based on clipping method it is mapped to *c'*.

2.2. Applying genetic algorithm for gamut mapping

Before introducing the suggested gamut mapping method, it is practical to have a brief description of a simple genetic algorithm method.

2.2.1. Genetic algorithm [14-16]

Genetic Algorithm (GA) developed by Holland is a stochastic global search method that mimics the metaphor of natural biological evaluation. GA works with a population of individuals, each representing a possible solution to a given problem. All individuals of one generation are evaluated by a given fitness function. The highly fit individuals are given opportunities to reproduce new offspring by cross breeding with other individuals in the population. After a number of iterations, the population consists of individuals that are well adapted in terms of the fitness function.

The simplest form of Genetic Algorithm involves three types of operators: selection or reproduction, crossover and mutation. Selection operator selects chromosomes in the population for reproduction according to their fitness function values. The fitter the chromosomes, the more times it is likely to be selected to reproduce. This operator may be implemented in several ways such as roulette wheel and rank based selection.

Crossover operator takes two individuals and cuts their chromosome strings at some randomly chosen position to produce two ''head'' segments and two ''tail'' segments. The tail segments are then swapped over to produce two full length chromosomes. The two offsprings each inherit some genes from each parent.

Mutation operator is a random process where some of the bits in a chromosome are replaced by another (0/1 and 1/0) to produce a new genetic structure. The rate of mutation is often seen as providing a guarantee that the probability of searching any given string will never be zero.

The basic steps of a standard Genetic Algorithm are as follows:

1. Generate a random population of L bit chromosomes of specific size (n). It should be consider that the population size affects the efficiency and performance of GA.
2. Calculate the fitness of each chromosome in the population.
3. Select a pair of chromosomes from the current population for mating by a random selection method.
4. Based on probability of crossover, the operator is applied on the selected pair that was chosen for crossover operation; then, the correspondent bit(s), selected for mutation based on probability of bit mutation, were mutated.
5. Repeat steps 2-4 until the production of next generation exceeds the size of the previous generation.
6. Go through steps 2-6 until the termination criteria is met.

If the GA has been correctly implemented, convergence is met. When the population converges, the average fitness will approach that of the best chromosome.

2.2.2. Gamut mapping using genetic algorithm

At first it is necessary to specify genes. The color coordinates of out of gamut colors of the image in CIELAB color space is defined as the genes. Considering that the color characteristic is three dimensional (L*, a*, b*), the genes would be three dimensional too. In this way, in the case of 8 out of gamut colors, for instance, the chromosome will be consisted of 8 genes. Coding step to binary numbers is not necessary and the gene values are represented in decimal format. The population size is set to about 60.

Fitness function is defined as the inverse of the color image difference between the original and mapped images computed by S-LAB formula. The following equation shows that:

$$fittness = 1/(S-LAB(I_{original}, I_{mapped}) + \varepsilon) \qquad (2)$$

Where $I_{original}$ and I_{mapped} represent the original and mapped images and ε is an insignificant value to avoid division by zero.

To the goal of gamut mapping, special genetic operators have been defined as follows:

Cross over:

Two methods have been developed to do crossover. After selecting two chromosomes for crossover, one of these methods is chosen by random and applied.

1) Two offsprings are produced by linear combination of the two selected chromosomes by the ratios of 3:1 and 1:3. In this way, the two offsprings each inherit some characteristics from each parent. Figure 2 shows an example of this crossover for one of the genes (color coordinates) of the two selected chromosomes; similar operation is applied on the other genes.

$$\begin{matrix} L_1^* \\ a_1^* \\ b_1^* \end{matrix} , \begin{matrix} L_2^* \\ a_2^* \\ b_2^* \end{matrix} \Rightarrow \begin{matrix} \left(2L_1^* + L_2^*\right)/3 \\ \left(2a_1^* + a_2^*\right)/3 \\ \left(2b_1^* + b_2^*\right)/3 \end{matrix} , \begin{matrix} \left(L_1^* + 2L_2^*\right)/3 \\ \left(a_1^* + 2a_2^*\right)/3 \\ \left(b_1^* + 2b_2^*\right)/3 \end{matrix}$$

Figure 2: The introduced cross over operator base on the linear combination of two chromosomes.

2) Considering that the defined chromosomes are three dimensional (L*, a* and b*), the second crossover operator is defined to swap the values of a specified dimension. To do it, at first, a number (n) is chosen between 1 to 3 by random showed the changeable dimension, and then the values of the chosen dimension is exchanged for the two chromosomes. Figure 3 shows this operation for one of the genes of the two chromosomes. As illustrated, crossover operator swaps the b* values.

$$\begin{matrix} L_1^* \\ a_1^* \\ b_1^* \end{matrix} , \begin{matrix} L_2^* \\ a_2^* \\ b_2^* \end{matrix} \xrightarrow{n=3} \begin{matrix} L_1^* \\ a_1^* \\ b_2^* \end{matrix} , \begin{matrix} L_2^* \\ a_2^* \\ b_1^* \end{matrix}$$

Figure 3: The second proposed cross over operator.

Mutation:

After selecting a chromosome for mutation, a random number is chosen between 1 and the total number of genes existed in a chromosome. Then a* and b* values of the selected gene is exchanged. The L* value of the mentioned gene might be subtracted from 100 by 50 percentage chance. Mutation can be applied for more than one gene. The number of genes, which are chosen for mutation, is selected by random between one and the number of genes.

It should be mentioned that after applying each genetic operation, the new genes are checked to be in the gamut. If a produced gene is out of the destination gamut, it will be mapped to its nearest color on the gamut surface. So, all the offsprings would be in the gamut.

3. Results and discussion

Figure 4 shows a color image together with its mapped images by the genetic gamut mapping and clipping mapping. In this figure, image (a) is the original image which contains 28 colors and the size of the image is 128×128 pixels. Images (b) and (c) show the results of mapping the out of gamut colors of this image to a specified gamut by the proposed genetic and clipping method, respectively. The color image differences between the original image and the mapped ones are computed by S-LAB color image difference formula. Using genetic algorithm for gamut mapping, the image differences can reduce from 4.49 to 3.46 S-LAB units. Figure 5 shows another example similar to Figure 4. The original image has 20 colors and 210×176 pixels. Color image differences between the original and mapped images are 6.06 S-LAB units by GA gamut mapping; clipping method shows 7.74 S-LAB units.

Advantages of using genetic algorithm for color gamut mapping

1. This method attempts to reduce the perceptual color image difference between the original and mapped images, which is not completely achievable by the common methods. In addition, the fitness function of genetic algorithm can be easily substituted by any better and newer term, so it can be always updated based on the newest color image difference equations.
2. One of the problems of common clipping algorithm is choosing the suitable direction for mapping. However, the obtained colors of genetic operators are

not dependent on a specific direction and they are only chosen to decrease the color image difference of the two images.

The problem of genetic algorithm is that the speed of convergence decreases by increasing the number of genes. To overcome this problem as much as possible the result of common clipping method can be used in the initial population. In this way, at first, clipping method is applied and then genetic runs while one of its initial population is the result of clipping. This solution can noticeably improve the rate of convergence.

Figure 4: (a): original image, (b): the mapped image by clipping gamut mapping (S-Lab: 4.49), (c): mapped image by the proposed genetic gamut mapping (S-Lab: 3.46).

Figure 5: (a): original image, (b): the mapped image by clipping gamut mapping (S-Lab: 7.74), (c): mapped image by the proposed genetic gamut mapping (S-Lab: 6.06).

4. Conclusions

The main goal of an ideal gamut mapping is to find the closest image to the original one from a set of reproduced images. This aim has not been achieved despite of the variety of proposed gamut mapping methods. In this study, we suggested a new gamut mapping method applying genetic algorithms. The algorithm is implemented in CIELAB color space and the out of gamut colors form the genes. The inverse of color difference between the original and mapped images based on S-LAB equation is used as fitness function and the genetic operators are defined by special terms. The proposed method is an approach to minimize the difference between the original and mapped images. The results show an improvement in comparison with the common clipping method. In addition, the suggested algorithm has the capability of replacing its fitness function by introducing any better perceptual color image difference formula. Furthermore, the problem of GA is its time consuming procedure in comparison to clipping method, which can be almost improved by putting the result of clipping method as one of the initial population.

5. References

1. P. Green, L. MacDonald, Colour Engineering. John Wiley & Sons, New York, 2002.
2. J. Morovic, M. R. Luo, The fundamentals of gamut mapping: A survey. *J. Imaging Sci. Technol.* 45(3)(2001), 283-290.
3. G. J. Braun, M. D. Fairchild, Color Gamut Mapping in a Hue-Linearized CIELAB Color Space, In Proceedings of IS&T/SID 6th Color Imaging Conference, Scottsdale, (1998) 163-168.
4. G. Marcu, Gamut Mapping in Munsell Constant Hue Sections, In Proceedings of IS&T/SID 6th Color Imaging Conference. Scottsdale, (1998) 159-162,.
5. P. G. Herzog, Specifying color gamut boundaries, Electronic Imaging (SPIE/IS&T International Technical Group Newsletter), 9(1)(1999) S. 1796-1806.
6. J. Morovic, M. R. Luo, Calculating Medium and Image Gamut Boundaries for Gamut Mapping. *Color Res. Appl.* 25(6)(2000), 394-401.
7. F. Ebner, M. D. Fairchild, Gamut mapping from below: Finding minimum perceptual distances for colors outside the gamut volume. *Color Res. Appl.* 22(1997), 402-413.
8. R. Kimmel, D. Shaked, M. Elad, I. Sobel, Space-dependent color gamut mapping: a variational approach. Image Processing, *IEEE Transactions on*, 14(6)(2005), 796-803.
9. H. S. Lee, D. Han, Implementation of real time color gamut mapping using neural network. Soft Computing in Industrial Applications, 2005. SMCia/05. In Proceedings of the 2005 IEEE Mid-Summer Workshop on, 28(30)(2005), 138-141.
10. J. Giesen, E. Schuberth, K. Simon, D. Zeiter, P. Zolliker, Framework for Image-Dependent Gamut Mapping, In Proceedings of the 18th Annual IS&T/SPIE Symposium on Electronic Imaging (EI), (2006), 6058-05.
11. C. Munteanu, A. C. Rosa, M. J. Galán, E. Royo, Evolutionary Color Constancy Algorithm Based on the Gamut Mapping Paradigm, In Proceedings of *EUROCAST2005* (2005), 404-409.
12. X. Zhang, B. A. Wandell, A spatial extension to CIELAB for digital color image reproduction, In Proceedings of the SID Symposiums, 27(1996) 731-734.
13. J. Morovic, CIE Division 8 - TC8-03: Gamut Mapping, http://www.colour.org/tc8-03/pgma.html
14. A. M. S. Zalzal, P. J. Fleming, Genetic Algorithms in engineering systems. The Institution of Electrical Engineering, In: Atherton DP, Irwin GI, editors, 1997.
15. D. E. Goldberg, Genetic Algorithms in search, optimization, and machine learning. Addison-Wesley Inc, 1989.
16. M. Mitchell, An introduction to Genetic Algorithms. Massachusetts Institute of Technology, 1999.