

A New Color Image Quantization Algorithm Based on Fuzzy Kernel Clustering^①

MA Yu-jie^a

^a *Department of Computer Science, Shangqiu Normal Univeristy, Shangqiu, China*

Abstract

A new color image quantization algorithm based on fuzzy kernel clustering is studied in this paper. Firstly, the original image is quantized to 256 colors using the octree algorithm. Secondly, based on the quantitative relation of the NBS distance and the color difference of human vision, the initial clustering centers and number are determined automatically. Thirdly, the clustering center color values are modified by use of the fuzzy kernel clustering algorithm in the Munsell space. And then, a color image quantization effect is achieved. At last, simulations are performed on the presented algorithm, and the simulation result shows that the presented algorithm not only can solve the problem of giving the number of quantization in advance but also has better quantization effect than the octree algorithm and fuzzy c-means algorithm in the same quantization number.

Index Terms: Image; Quantization; Fuzzy Kernel Clustering; Octree; Fuzzy C-means

© 2011 Published by MECS Publisher. Selection and/or peer review under responsibility of the International Conference on E-Business System and Education Technology

1. Introduction

In multi-media application, industrial arts, dyeing and printing industry and some other fields, color image processing is one of the most urgent issues to be concerned. Generally speaking, there are 2^{24} categories of colors in a true color image. How to select k (k should be less than 256 in general) representative colors from this large amount of colors to show the original image as truly as possible, so that the number of colors in an image can be reduced without any damage of its visual effect, which is the main idea of color quantization. Color quantization plays a very important part in image compression, image segmentation and image restoration. In recent years, several color image quantization algorithms have been put forward and applied, such as c-means algorithm [1] and fuzzy c-means algorithm [2] based on clustering, and median-cut quantization algorithm [3] and octree quantization algorithm [4] based on cutting. Both octree algorithm and fuzzy c-means algorithm have achieved satisfactory results when they are applied to color quantization under certain circumstance, but they can reach such satisfactory result only when the colors of an quantized image is

^① This paper is sponsored by Henan Provincial Natural Science Funds Program (092300410220).

Corresponding author:

E-mail address:^a shqmyj123@163.com

known or the color number is determined previously. But under some other circumstances, processors do not know the necessary information previously, such as the type of an image or its color categories, or the information they get previously maybe not completely accurate, and as a result there is a great difference between the quantization effect of an image and its original visual effect. To get rid of the prior need of quantization number, reference [5] puts forward a self-adaptive color image quantization algorithm based on image content. Inspired by this reference, this paper firstly employs octree algorithm to quantize the original image into 256 categories of colors in RGB color space because in general the number of main categories of colors in each color image are limited to no more than 256, and then confirms the initial clustering center in Munsell color space according to the quantitative relation of NBS color distance and color difference in human vision. After the confirmation of the initial clustering center, the fuzzy c-means algorithm can be applied directly for clustering, but this algorithm is only applicable to spherical or ellipsoidal clustering, and the effect of clustering is determined by the distribution of samples to a very great extent. To be more applicable to the image quantization, this new algorithm brings in kernel function of SVM (support vector machine) [6], so that fuzzy kernel clustering method can be employed to cluster every pixel in Munsell color space in order to modify the colors and finish the quantization of an image.

2. Color Conversion from RGB Color Space to Munsell Color Space

Color is a very important visual property of an image. The visual consistency of colors refers to the smooth change of colors in color space, that is, when the triple coordinate of pixel color changes continuously, it seems that the color also changes continuously to human vision. Comparisons of lots of color space in experiments and practice demonstrate that the Munsell color space truthfully simulates the features of human color vision [7] and successfully keeps color's visual consistency. Therefore, this paper will discuss about image quantizing process in Munsell color space.

In order to convert color data in RGB color space (R, G, B) into color data in Munsell color space (H, V, C), we should firstly convert RGB color data into color data in XYZ color space as follows:

$$\begin{cases} X = 0.620R + 0.178G + 0.204B \\ Y = 0.299R + 0.578G + 0.144B \\ Z = 0.056G + 0.942B \end{cases} \quad (1)$$

Thirdly, we should convert color data in XYZ color space into color data in PQ color space as follows:

$$\begin{cases} P = f(X) - f(Y) \\ Q = 0.4[f(Z) - f(Y)] \\ \text{Where, } f(X) = 11.6X^{1/3} - 1.6 \end{cases} \quad (2)$$

And then, we should convert color data in PQ color space into color data in ST color space as follows:

$$\begin{cases} S = (8.880 + 0.966\cos\theta)P \\ T = (8.025 + 2.558\sin\theta)Q \\ \text{Where, } \theta = \arctan(P/Q) \end{cases} \quad (3)$$

Finally, we can get the color data as follows:

$$\begin{cases} H = \arctan(S/T) \\ V = f(Y) \\ C = \sqrt{S^2 + T^2} \end{cases} \quad (4)$$

If $A = (H_1, V_1, C_1)$ and $B = (H_2, V_2, C_2)$ is the color pair in Munsell color space, their NBS color distance (D_{NBS}) can be defined as follows:

$$D_{NBS}(A, B) = 1.2 \left\{ 2C_1C_2 \left[1 - \cos\left(\frac{2\pi}{100}\right)\Delta H \right] + (\Delta C)^2 + 4(\Delta V)^2 \right\}^{1/2} \quad (5)$$

Where, $\Delta H = |H_1 - H_2|$, $\Delta V = |V_1 - V_2|$, $\Delta C = |C_1 - C_2|$

In his research, Y. H. Gong [8] discovers that: colors will be considered as similar in human vision when their NBS color distance is below 3.0, while colors will be felt remarkably different in human vision when their NBS color distance is above 6.0.

3. Confirmation of the Initial Clustering Center

With consideration of the quantitative relation of NBS color distance and color difference in human vision, the automatic confirmation of clustering center and clustering number can be realized according to the following steps:

- (1) Quantizing the original image in RGB color space into 256 colors using octree algorithm, and then converting these 256 colors to Munsell color space.
- (2) Calculating the image's NBS color distance according to (5), and creating an array Dis to store information of colors with the NBS color distance above 3. The array members include *ColorIndex1*, *ColorIndex2* and *dis* (the first two items are colors' index number, while the third one is the NBS color distance between two colors, and the supposed number of array elements is m).
- (3) Confirming NBS threshold value: making a statistic of the pixel number of each category of colors to establish weight array A for the storage of the weight of each distance in Dis. In an image, when the pixel number of colors whose index number is *ColorIndex1* is defined as count1, and the pixel number of the colors whose index number is *ColorIndex2* is defined as count2, the array A(i) should be:

$$A(i) = \frac{count1 + count2}{256} \quad (6)$$

And then weighted average of image's NBS color distance should be:

$$Aver = \frac{\sum_{i=1}^m A(i) \times Dis(i).dis}{\sum_{i=1}^m A(i)} \quad (7)$$

$Aver$ is NBS threshold value of an image.

- (4) Confirming clustering number and clustering centers: putting the 256 categories of colors in line according to their pixels number from smallest to largest, and then choosing the color with the largest pixels number as the first clustering center to calculate the NBS color distance between this color and other colors with application of (5); then choosing the second clustering center according to the principle $D_{NBS} \geq Aver$, and then choosing the colors whose NBS color distance with the former two clustering center is large or equal to $Aver$ as the third clustering center according to the same principle; choosing clustering centers in this way successively till there is no more eligible colors. Then, we get a color collection with certain number of colors, in which the color distance of every two categories of colors is large or equal to $Aver$. And thus the clustering centers and clustering number are confirmed, and can be defined as:

$$Color(H, V, C) = \{Color_1, Color_2, Color_3, \dots, Color_n\} \quad (8)$$

Where, n is the clustering number, namely the number of colors in color collection.

4. Color Quantization Based on Fuzzy Kernel Clustering

4.1. Fuzzy kernel clustering method

We suppose that $X = \{x_k, k = 1, 2, \dots, N\} \in R^p$ is the sample collection to be classified, n is confirmed number of clusterings, $C = \{c_1, c_2, \dots, c_n\}$ is n clustering centers, and $u_{ik} (i = 1, 2, \dots, n, k = 1, 2, \dots, N)$ is the membership function of the k th sample to the i th category and meet the conditions:

$$0 \leq u_{ik} \leq 1, \sum_{i=1}^n u_{ik} = 1$$

Fuzzy clustering is the process of calculating the n clustering centers C in sample collection and membership functional matrix $U = [u_{ik}]_{n \times N}$, and the clustering criterion function of fuzzy c-means can be defined as:

$$J(U, C) = \sum_{i=1}^n \sum_{k=1}^N u_{ik}^m \|x_k - c_i\|_2 \quad (9)$$

In this formula, $m > 1$ is weighting index. The purpose of clustering is to minimize $J(U, C)$.

Let's define nonlinear mapping $\Phi: X \rightarrow F$, and thus $x \in R^p \rightarrow \Phi(x) \in R^q$, where, F is high-dimension feature space. The criteria function of fuzzy kernel clustering is [9]:

$$J(U, C) = \sum_{i=1}^n \sum_{k=1}^N u_{ik}^m \|\Phi(x_k) - \Phi(c_i)\|^2 = \sum_{i=1}^n \sum_{k=1}^N u_{ik}^m d_F^2(x_k, c_i) \quad (10)$$

$$d_F^2(x_k, c_i) = \|\Phi(x_k) - \Phi(c_i)\|^2 = K(x_k, x_k) - 2K(x_k, c_i) + K(c_i, c_i) \quad (11)$$

In this formula, $K(x_k, c_i)$ is kernel function, and $d_F(x_k, c_i)$ is the Euclidean distance of feature space. When the kernel function adopts Gaussian kernel, the clustering criteria can be simplified as:

$$J(U, C) = \sum_{i=1}^n \sum_{k=1}^N u_{ik}^m (2 - 2K(x_k, c_i)) \quad (12)$$

Where,

$$K(x_k, c_i) = \exp\left(\frac{-\|x_k - c_i\|^2}{\sigma^2}\right) \quad (13)$$

σ is the parameter of Gaussian function, which can be confirmed according to the given sample collection as[10]:

$$\sigma = \sqrt{\frac{\sum_{k=1}^N \|x_k - \bar{x}\|^2}{N}}, \quad \bar{x} = \frac{\sum_{k=1}^N x_k}{N} \quad (14)$$

To solve (12) according to Lagrange multiplier method, we can get the following results:

$$u_{ik} = \frac{(1 - K(x_k - c_i))^{1/(m-1)}}{\left(\sum_{i=1}^n (1 - K(x_k - c_i))^{-1/(m-1)}\right)^{-1}} \quad (15)$$

$$c_i = \frac{\sum_{k=1}^N u_{ik}^m K(x_k, c_i) x_k}{\sum_{k=1}^N u_{ik}^m K(x_k, c_i)} \quad (16)$$

4.2. Steps of color quantization based on fuzzy kernel clustering

After getting the initial clustering center $Color(H, V, C) = \{Color_1, Color_2, Color_3, \dots, Color_n\}$, we should convert the original image from RGB color space to Munsell color space, and then cluster each pixel in Munsell color space by using the fuzzy kernel clustering algorithm mentioned above so that we can get a reliable quantization result. The implementation steps are:

- (1) Making $s = 1$, $C^0 = \{Color_1, Color_2, Color_3, \dots, Color_n\}$, then choosing iteration stopping condition $\varepsilon \in (0, 1)$, and employing (14) to calculate σ ;
- (2) Substituting C^{s-1} in (15) to calculate U^s ;
- (3) Substituting C^{s-1} and U^s we get in (16) to calculate C^s ;
- (4) Stopping calculating when $\|C^s - C^{s-1}\| < \varepsilon$, or turning back to step (2) when $s = s + 1$.

5. Simulation Study

Simulation study of the algorithm put forward in this paper is also made in the Matab6.5 software environment, and several kinds of images are involved in this study. The Fig. 1(a), 2(a) and 3(a) are original true color images. The Fig. 1(b), 2(b) and 3(b) are the results of octree quantization; the Fig. 1(c), 2(c) and 3(c) are the results of fuzzy c-means quantization. The Fig. 1(d), 2(d) and 3(d) are the quantization results of the new algorithm put forward in this paper. Table 2 is the comparison of average quantization errors of original images and quantized images by using (17). And from this table, we can find that the color numbers of octree algorithm and fuzzy c-means algorithm is same with the color numbers confirmed automatically by the algorithm raised in this paper. The comparisons in table 1, image 1, image 2 and image 3 demonstrate that the quantization results of algorithm put forward in this paper are obviously better than the results of octree algorithm and fuzzy c-means algorithm when the quantization levels are the same.

$$MSN = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N d[s(i, j), q(i, j)]^2 \quad (17)$$

Where, $s(i, j)$ and $q(i, j)$ refer to the color values of original pixel and quantized pixel, M and N are the width and height of the image, and $d[s(i, j), q(i, j)]$ are the Euclidean distance of two pixels.

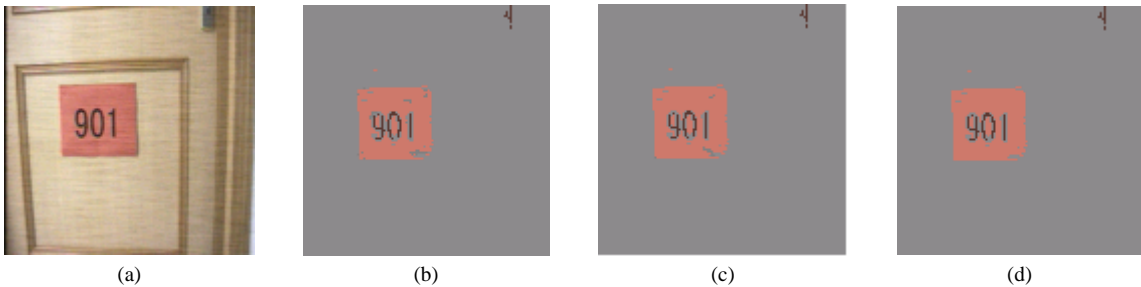


Fig. 1. Door-Plate

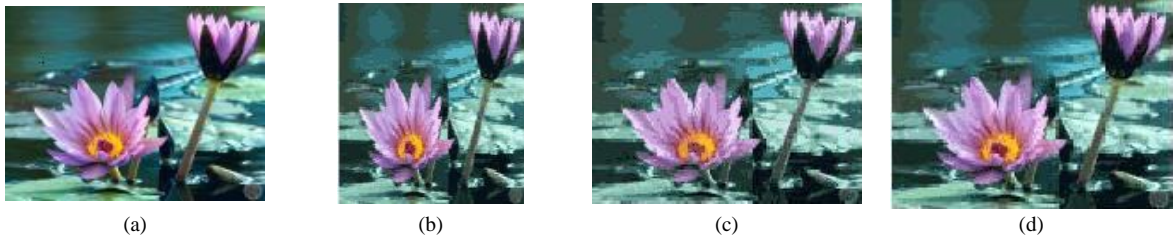


Fig. 2. Lotus Flower

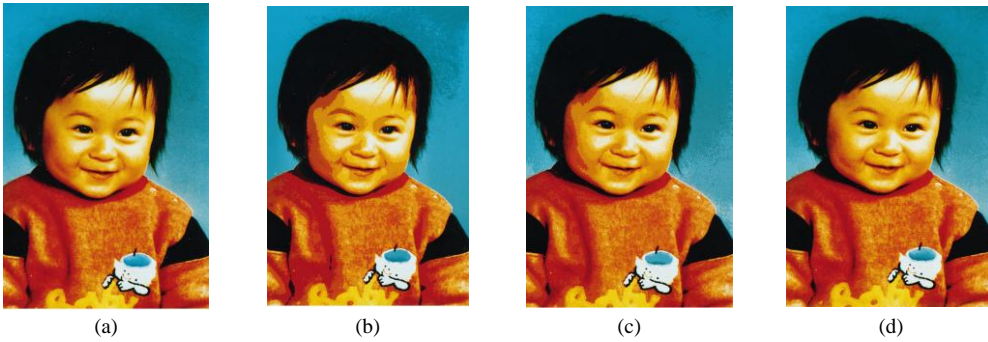


Fig. 3. A Girl

Table 1. A Comparison of Images' Average Quantization Error

Original Image	Color Number after Quantization	Octree Algorithm	Fuzzy C-Means Algorithm	New Algorithm in this Paper
Door-Plate	3	25.61	23.57	16.58
Lotus Flower	86	15.16	11.08	9.24
A Girl	256	13.18	12.45	6.97

6. Conclusion

Inspired by the factor mentioned in [7-8] that Munsell color space truthfully simulates the features of human color vision and the quantitative relation of NBS color distance and color difference in human vision, this paper put forward a method of automatic confirming color number for quantization. After the confirmation of color number and initial clustering center, we can bring in kernel function, and cluster each pixel in Munsell color space by fuzzy kernel clustering algorithm in order to fulfill the quantization. And the quantization results are obviously better than octree algorithm and fuzzy c-means algorithm when the quantization levels are the same.

References

- [1] Hideo Kasuga. Color quantization using the fast k-means algorithm [J]. *Systems and Computers*, 2000, 31(8):1120-1128.
- [2] Ozdemir D, Akarun, L. A fuzzy algorithm for color quantization of images [J]. *Pattern Recognition*, 2002, 35, 1785-1791.
- [3] Heckbert P. Color image quantization for frame buffer display [J]. *Computer Graphics*, 1982, 16(2):297-307.
- [4] Gervautz M, Purgathofer W. A simple method for color quantization: octree quantization [C] *Proceeding of Graphics Gems International*. San Diego: Academic Press Professional, 1998, 8(6):219-230.
- [5] Wang Xangyang, Hu Fengli, Liu Chunhui. A new adaptive color image quantization algorithm [J] *Journal of Liaoning Normal University*, 2007, 30(3):310-314. [王向阳, 胡峰丽, 刘春辉. 一种基于图像内容的自适应色彩量化算法[J]. *辽宁师范大学学报*, 2007, 30(3):310-314.]
- [6] Wu Zhongdong, Gao Xinbo, Xie Weixin. A study of a new fuzzy clustering algorithm based on the kernel method [J]. *Journal of Xidan University*, 2004, 31(4):533-537. [伍忠东, 高新波, 谢维信. 基于核方法的模糊聚类[J]. *西安电子科技大学学报*, 2004,31(4):533-537.]
- [7] Ma W Y, Manjunath S. Edgeflow: A framework for boundary detection and image segmentation [J]. *IEEE Trans on Image Processing*, 2000, 9(8):1375-1388.
- [8] Gong Y H, Proietti G. Image indexing and retrieval based on human perceptual color clustering[C]. *The International Conference on Computer Vision*, Mumbai, 1998.
- [9] Zhang D Q, Chen S C. Fuzzy C-means and possibilistic C-means algorithms under kernel based robust metric [J]. *Pattern Recognition and Artificial Intelligence*, 2004, 17(4):390-395.
- [10] Wu K L, Yang M S. Alternative c-means clustering algorithm [J]. *Pattern Recognition*, 2002, 35 (10):2267-2278.