

Computing the Number of Groups for Color Image Segmentation Using Competitive Neural Networks and Fuzzy C-Means

Farid García-Lamont¹(✉), Jair Cervantes¹, Sergio Ruiz¹,
and Asdrúbal López-Chau^{1,2}

¹ Centro Universitario UAEM Texcoco, Universidad Autónoma del Estado de México, Av. Jardín Zumpango s/n, Fraccionamiento El Tejocote, CP 56259 Texcoco, Estado de México, Mexico
{fgarcial, alchau}@uaemex.mx,
chazaral7@gmail.com, jsergioruizc@gmail.com

² Centro Universitario UAEM Zumpango, Universidad Autónoma del Estado de México, Camino Viejo a Jilotzingo continuación Calle Rayón, 55600 Zumpango, Estado de México, Mexico

Abstract. Fuzzy C-means (FCM) is one of the most often techniques employed for color image segmentation; the drawback with this technique is the number of clusters the data, pixels' colors, is grouped must be defined a priori. In this paper we present an approach to compute the number of clusters automatically. A competitive neural network (CNN) and a self-organizing map (SOM) are trained with chromaticity samples of different colors; the neural networks process each pixel of the image to segment, where the activation occurrences of each neuron are collected in a histogram. The number of clusters is set by computing the number of the most activated neurons. The number of clusters is adjusted by comparing the similitude of colors. We show successful segmentation results obtained using images of the Berkeley segmentation database by training only one time the CNN and SOM, using only chromaticity data.

Keywords: Color characterization · Color spaces · Competitive neural networks

1 Introduction

Image segmentation by color features has been employed for different purposes; for instance, analysis of rocks [1], food [2, 3], medicine [4, 5], among others [6, 7]. The algorithms of related works are based on FCM [8–10]; the FCM require a priori knowledge of the number of clusters the image should be segmented.

In related works the number of clusters is defined by the user [11, 12]; in other ones the number is set by computing the “dominant” colors of the image [10, 13–15]. Basically, the techniques introduced in [13, 14] work as follows: a SOM is trained with the colors of the image, represented in the RGB space, and then the image is processed by the SOM where in a histogram the activation occurrences of each neuron are collected. Finally the number of clusters is obtained by computing the number of peaks of the histogram.

Using this technique involves the SOM is trained every time a new image is given, which it may be time consuming. On the other hand, most of the works employ the RGB space to represent colors but this space is light sensitive, i.e., two colors with the same chromaticity but with different intensities can be grouped in different clusters [16]. Despite other works employ other color spaces, where the chromaticity is separated from the intensity, the intensity data is also used to process the images, but similar effects are obtained as if the RGB space is employed [10, 11, 14, 17].

Our proposal consist on computing the number of dominant colors by processing the chromaticity of colors, emulating the way the humans perceives colors; that is, humans recognize colors mainly by the chromaticity then by the intensity [16, 18]. Also, humans are capable to recognize the different regions within an image by just identifying the chromaticity features of such sections without using the intensity data of the colors; it is important to mention humans employ the knowledge acquired previously to recognize colors, i.e., they do not need to learn to recognize the colors every time they need to identify a color.

Hence, the contribution of this paper is an approach where an unsupervised neural network (NN) is trained with chromaticity samples of different colors. The image is processed by the NN using the chromaticity data of the colors which is obtained by mapping, previously, the image to the HSV color space. The activation occurrences of each neuron of the NN are collected in a histogram, the number of neurons with the highest activation occurrences, number of clusters, is obtained; the image is segmented using the FCM with the number of clusters obtained.

The paper is organized as follows: in Sect. 2 we introduce our approach for segmentation. The experiments performed and the results obtained are presented in Sect. 3. In Sect. 4 the results are discussed; the paper closes with conclusions and future work in Sect. 5.

2 Proposed Approach

By observing their environment humans can recognize different regions within a scene by the chromaticity features, because humans identify colors mainly by the chromaticity features, later by the intensity [16, 18]. For instance, we can state the color of squares (a) and (b) of Fig. 1 is green because both squares have the same chromaticity although the square (a) is brighter than square (b). On the other hand, we can claim the colors of squares (c) and (d) are different because the chromaticities of both squares are different despite the intensities are the same.

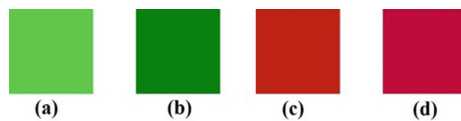


Fig. 1. Color of squares (a) and (b) with the same chromaticity but with different intensities; color of squares (c) and (d) with different chromaticity but with the same intensity (Color figure online)

Humans do not have to learn the colors every time they need to recognize a given color; they just use the knowledge acquired previously. Thus, the steps of our proposal are: (1) train a NN with chromaticity samples of different colors; (2) map the image to the HSV space and extract the chromaticity of each pixel's color of the given image; (3) the chromaticity of each pixel is feeded to the NN trained previously; (4) the activation occurrences of each neuron are collected in a histogram; (5) the number of peaks of the histogram is computed, it defines the number of clusters; (6) the number of clusters is updated by comparing the chromaticity of the neurons with the highest activation occurrences; (7) the image is segmented with the number of clusters obtained in step 6 using FCM.

The number of clusters is updated because there may be neurons with similar chromaticity and occurrence number; that is, the colors these neurons recognize are almost the same, so, they belong to the same section and they must be grouped in the same cluster.

2.1 Chromaticity Characterization

The RGB (Red, Green, Blue) color space is based in a Cartesian coordinate system where colors are points defined by vectors that extend from the origin, where black is located in the origin and white in the opposite corner to the origin, see Fig. 2.

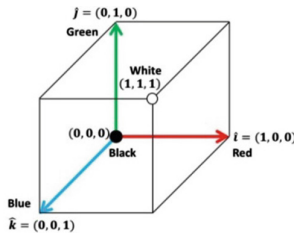


Fig. 2. RGB color space (Color figure online)

The color of a pixel p is written as a linear combination of the basis vectors red, green and blue [16]:

$$\phi_p = r_p \hat{i} + g_p \hat{j} + b_p \hat{k} \tag{1}$$

Where r_p , g_p and b_p are the red, green and blue components, respectively. The orientation and magnitude of a color vector defines the chromaticity and the intensity of the color, respectively [16]. This color space is sensible to non-uniform illumination; even if two vectors with the same chromaticity but with different intensities, they represent different colors.

Thus, we employ the HSV (Hue, Saturation, Value) color space to represent colors because the chromaticity is decoupled from the intensity [16]; also, in [18] is claimed

this space emulates the human perception of color. Figure 3 shows the cone shaped of the HSV space, where the color of a pixel p in the HSV space is written as [16]:

$$\varphi_p = [h_p, s_p, v_p] \tag{2}$$

Where h_p , s_p and v_p are the hue, saturation and value components, respectively. The hue is the chromaticity, saturation is the distance to the glow axis of black-white, and value is the intensity. The real ranges of hue, saturation and value are $[0, 2\pi]$, $[0, 1]$, and $[0, 255]$, respectively.

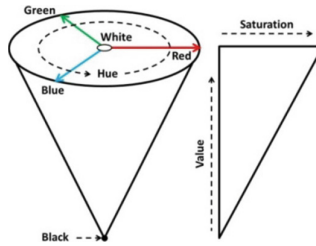


Fig. 3. HSV color space (Color figure online)

The chromaticity is distributed around the circumference of the cone; black is located at the cone’s tip and white at the center of the base at the bottom of the cone. Black and white are not chromaticities, they can be defined as a low value color and low saturation color, respectively.

The chromaticity is characterized as a vector because of the case when the hue is almost 0 or 2π . Consider squares (c) and (d) of Fig. 1, their hue values are $\pi/100$ and $19\pi/10$, respectively. Numerically both values are very different but the chromaticities of both squares are similar; if the chromaticity of both squares is classified only by the scalar values, the chromaticity would be recognized as if they were very different. Thus, this problem is overcome as follows; let φ_p be the color of a pixel in the HSV space represented as in Eq. (2), the chromaticity is modeled as:

$$\psi_p = [\cos h_p, \sin h_p] \tag{3}$$

2.2 Neural Network Training

In this study we perform experiments with a CNN and a SOM. These kind of unsupervised NNs are based on finding the index of the winning neuron before external stimuli. The difference between these NNs lies in the training. In a CNN, only the weight vector of the winning neuron is updated; while in a SOM, where the neurons are set in a specific array, the weight vector of the winning neuron and the weight vectors of the neighbor neurons are updated. The weight vectors of the neurons are updated with the Kohonen learning rule [19].

Because of the fuzzy nature of color, it is not possible to recognize all the colors of the spectrum; hence, the color spectrum is “divided in” a finite amount of colors. The number of colors the NN can recognize depends on its size; in this paper we perform tests with a CNN with 16 neurons and a SOM also with 16 neurons set in a 4×4 array.

Therefore, both NNs can recognize up to 16 different colors. Both NNs are trained with the elements of the set Ψ built with chromaticity samples as follows:

$$\Psi = \left\{ \psi_k = [\cos \theta_k, \sin \theta_k] \mid \theta_k = \frac{2\pi}{256}k : k = 0, 1, \dots, 255 \right\} \quad (4)$$

2.3 Computing the Number of Clusters

Computing the number of clusters or sections of the image involves performing the following operations to each pixel of the image:

1. The pixel $\phi = [r, g, b]$ is mapped to the HSV space obtaining the vector $\varphi = [h, s, v]$
2. Verify if the color of the pixel is black by comparing its intensity with respect to a threshold value. That is, if $v < \delta_v$ then the occurrence for black is collected in the histogram, go to step 6.
3. If the color is not black then it is verified if the color of the pixel is white by comparing its saturation with respect to a threshold value. In other words, if $s < \delta_s$ then the occurrence for white is collected in the histogram, go to step 6.
4. If $v > \delta_v$ and $s > \delta_s$ then the color is a chromaticity and modeled it as Eq. (3).
5. The vector ψ is processed by the NN; the index of the winning neuron is collected in the histogram.
6. End.

Where δ_s and δ_v are the threshold values for saturation and value, respectively. Experimentally we found the best values are $\delta_s = \mu_s - \sigma_s$ and $\delta_v = \mu_v - \sigma_v$; where μ_s and μ_v are the mean of the saturation and intensity values of the image, respectively; σ_s and σ_v are the standard deviation of the saturation and intensity values of the image, respectively.

The number of clusters is defined by the number of bins of the histogram greater than zero. In the histogram there are also collected the activation occurrences of the neurons which are activated a few times; it implies that there are irrelevant small parts within the image that are considered they form important parts of the image.

Hence, there must be selected just the bins whose number is large enough to represent a significant section within the image. Thus, let P be the set of indexes of the normalized histogram bins whose number is greater or equal than threshold δ_H :

$$P = \{k \in \mathbb{N} \mid H(k) \geq \delta_H\} \quad (5)$$

Where $H(k)$ is the value of the normalized histogram at bin k . Therefore, the total of clusters c is the number of elements of the set P ; in other words:

$$c = |P| \quad (6)$$

2.4 Adjusting the Number of Clusters

As stated before, there may be neurons with almost the same activation occurrences and with similar chromaticity. That is, there is a section within the image with a common color that activates two neurons several times; therefore, such section of the image is segmented in two parts.

Due to the colors are almost the same it means they must be grouped in the same cluster. Thus, the colors of the neurons with the highest activation occurrences are compared by computing the orientation difference between the weight vectors of the neurons as follows:

$$\Delta\theta_{i,j} = \cos^{-1} \left(\frac{\mathbf{w}_i \cdot \mathbf{w}_j^T}{\|\mathbf{w}_i\| \|\mathbf{w}_j\|} \right), \forall i, j \in P, i \neq j \quad (7)$$

Where $\Delta\theta_{i,j}$ is the orientation difference between the weight vectors \mathbf{w}_i and \mathbf{w}_j of neurons i and j , respectively. If the orientation difference is smaller than the threshold value δ_θ then the number of clusters must be reduced. That is:

$$\text{if } \Delta\theta_{i,j} < \delta_\theta \text{ then } c_a = c_a + 1 \quad (8)$$

Where c_a is the number of similar colors. The final number of clusters c_t is:

$$c_t = c - c_a \quad (9)$$

Finally, the image is segmented using the FCM with c_t clusters. It is important to mention the image is processed with the colors represented in the RGB space.

3 Experiments and Results

Lately the Berkeley Segmentation Database¹ (BSD) is becoming the benchmark for testing segmentation algorithms of color images. We select randomly a set of 12 images, shown in Fig. 4, from the 300 color images the BSD contains. For experiments we employ the following threshold values: $\delta_H = 0.001$ and $\delta_\theta = \pi/6$, for both NNs.

Figure 5 shows the images obtained using the CNN. Table 1 shows the number of clusters computed and the final number of clusters obtained for segmentation for each image processed by the CNN.

The final number of clusters is computed with Eq. (9); i.e., the total of clusters minus the number of similar colors.

Figure 6 shows the images obtained using the SOM with 4×4 neurons. Table 2 shows the number of clusters computed and the final number of clusters obtained for

¹ <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>.



Fig. 4. Images employed for experiments; original images taken from the Berkeley image database (Color figure online)



Fig. 5. Images obtained after processing the images of Fig. 4 using the competitive neural network (Color figure online)

Table 1. Number of clusters computed and final number of clusters obtained for segmentation using the competitive neural network

Image	c	c_t	Image	c	c_t
1	7	6	7	8	6
2	8	6	8	8	6
3	7	5	9	7	5
4	6	4	10	9	7
5	8	6	11	7	5
6	6	4	12	7	5

segmentation for each image processed by the SOM. The final number of clusters is computed as Eq. (9).

By observing Tables 1 and 2, the SOM recognizes more colors than the CNN; but also, there are reduced more similar colors using the SOM. It is notable the appearance difference between the images of Figs. 5 and 6, where there are more colors or sections in the images obtained using the SOM. In the following section there are discussed and analyzed the results obtained.



Fig. 6. Images obtained after processing the images of Fig. 4 using the self-organizing map (Color figure online)

Table 2. Number of clusters computed and final number of clusters obtained for segmentation using the self-organizing map

Image	c	c_t	Image	c	c_t
1	8	7	7	11	6
2	10	7	8	12	8
3	8	5	9	9	6
4	7	6	10	13	9
5	11	6	11	10	6
6	7	4	12	10	7

4 Discussion

We claim, from the images shown in Sect. 3, it is possible to estimate the number of colors within an image by processing it using a NN trained with chromaticity samples, without training it every time an image is processed. The segmentation of the images using both NNs is, to some extent, similar because there are computed almost the same number of clusters; but the SOM recognizes more colors than the CNN. We perform a discussion about it in Sect. 4.1

On the other hand, the colors of the images, processed by the FCM, are represented in the RGB space, but in this space the chromaticity is sensitive to the intensity, we discuss in Sect. 4.2 how the segmentation is affected if the vectors are normalized and then processed by the FCM.

4.1 Comparison Between Both Neural Networks

Despite both NNs are trained with the same training set and with the same number of neurons, the number of colors they recognize is different although they process the same images. The segmentation of the images 1, 2, 3, 5, 6, 7, 9, 10, 11 and 12 obtained using both NNs is very similar because the final numbers of clusters are the same or almost the same. For instance, in image 1 it is computed 6 clusters using the CNN and 7 employing the SOM; in image 10 it is computed 7 clusters employing the CNN and 9 using the SOM; in image 12 it is computed 5 clusters using the CNN and 7 employing the SOM. In other images it is computed the same final number of clusters using both NNs; specifically images 3, 5, 6 and 7.

The images where the appearance difference is more notable, depending on the NN employed, are the images 4 and 8. In the image 4 processed by the CNN, the color corresponding to the grass area has the same green hue; but in the same section of the image, processed by the SOM, is segmented with two different kinds of green hue. In image 8, the grass and the house are segmented with the same color using the CNN; while using the SOM the grass and the house are segmented with different colors.

The SOM recognizes more colors, but when the number of clusters is adjusted, there are obtained values similar to the number of clusters obtained using the CNN. The plausible explanation is the architecture of the SOM emulates the geometric shape of the HSV about the chromaticity, that is, the base of the cone, see Fig. 3.

4.2 Normalizing the RGB Color Vectors

When the FCM are applied to the images, the colors of the pixels are represented in the RGB space. Despite several colors with the same chromaticity can be grouped in different clusters if their intensities vary. For instance, the image 8 of Figs. 5 and 6, the area corresponding to the water is segmented in three parts because of the intensity differences, despite the chromaticity is almost the same. Also, in image 12 of Fig. 6 and image 9 obtained with both NNs, the sky area is segmented with two intensity levels.

If only the chromaticity data is processed the segmentation changes significantly; that is, the color vectors are normalized and then they are processed by the FCM. In this way the color vectors have the same magnitude, therefore the same intensity; but the orientation does not change, so, the chromaticity does not change. Figure 7 shows the images obtained employing the CNN and normalizing the color vectors before they are processed by the FCM.

The colors of several sections are defined better, they are more homogeneous and it is easier to appreciate them. For example, in image 7 the section corresponding to the stone wall has only one kind of hue; in image 9 the sky is segmented homogeneously with the same blue hue. The segmentation of image 8 resembles the image segmentation obtained with the SOM; in the same image, using the CNN, the hue of the grass area is alike to the hue of the house, but in the same image of the Fig. 7 the same area is segmented in green hue, similar to the image obtained using the SOM.

In image 12 there are two areas segmented with different kinds of green hue, which it is not appreciated in the same image shown in Figs. 5 and 6, besides the hue of the sections are more homogeneous.

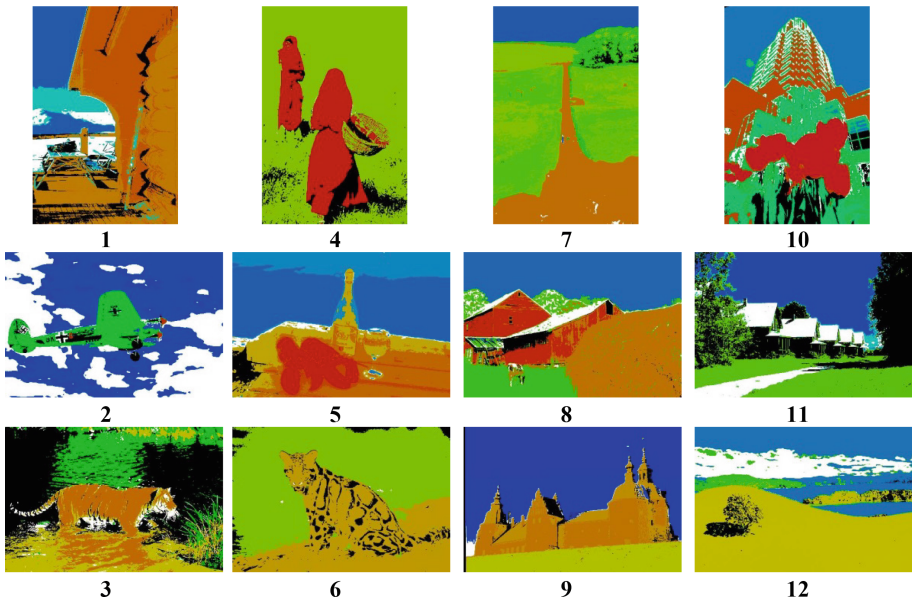


Fig. 7. Images obtained using the competitive neural network by normalizing the color vectors of the pixels before being processed by the FCM (Color figure online)

5 Conclusions and Future Work

We have introduced a proposal to compute the number of clusters a color image must be segmented using fuzzy *c*-means. We employ a competitive neural network and a self-organizing map, trained with chromaticity samples of several colors. The chromaticity data of each image's pixel is feeded to the neural networks, in a histogram there are collected the activation occurrences of each neuron. The number of the most activated neurons represents the amount of colors within the image; therefore, the number of sections or clusters. This number is adjusted by eliminating similar colors.

The performances of both neural networks are alike, although the self-organizing map recognizes more colors, when the number of clusters is adjusted, there are obtained almost the same values for both neural networks. The number of colors within the image is computed by employing only the chromaticity data; the image segmentation success without training the neural networks every time a new image is given.

As future work it is necessary to perform experiments with more images and with different threshold values δ_H and δ_θ , and sizes of the neural network, so as to find optimal values for these parameters that influence the segmentation; with and without normalizing the color vectors, as shown in images of Fig. 7, the segmentation seems to be more accurate if the color vectors are normalizing before they are processed with the fuzzy *c*-means. The quantitative evaluation of our segmentation proposal can be computed using the probabilistic rand index and variation of information metrics along with the Berkeley segmentation database, which are becoming the standard metrics and benchmark, respectively, to compute the performance of segmentation algorithms of color images [11].

References

1. Lepistö, L., Kuntuu, I., Visa, A.: Rock image classification using color features in Gabor space. *J. Electron. Imaging* **14**(4), 1–3 (2005)
2. Pathare, P., Linus, U., Al-Said, F.: Colour measurement and analysis in fresh and processed foods: a review. *Food Bioprocess Technol.* **6**(1), 36–60 (2013)
3. Santos, J., Rodrigues, F.: Applications of computer vision techniques in the agriculture and food industry: a review. *Eur. Food Res. Technol.* **235**(6), 989–1000 (2012)
4. Abbas, A.A., Guo, X., Tan, W.H., Jalab, H.A.: Combined spline and B-spline for an improved automatic skin lesion segmentation in dermoscopic images using optimal color channel. *J. Med. Syst.* **38**, 80 (2014)
5. Goffredo, M., Schmid, M., Conforto, S., Amosori, B., D'Alessio, T., Palma, C.: Quantitative color analysis for capillaroscopy image segmentation. *Med. Biol. Eng. Comput.* **50**(6), 567–574 (2012)
6. Guan, T., Zhou, D., Xu, C., Liu, Y.: A novel RGB Fourier transform-based color space for optical microscopic image processing. *Robot. Biomimetics* **1**, 16 (2014)
7. Ozturk, O., Aksac, A., Ozyer, T., Alhaji, R.: Boosting real-time recognition of hand posture and gesture for virtual mouse operations with segmentation. *Appl. Intell.* **43**(4), 786–801 (2015)

8. Kim, J.Y.: Segmentation of lip region in color images by fuzzy clustering. *Int. J. Control Autom. Syst.* **12**(3), 652–661 (2014)
9. Guo, Y., Sengur, A.: A novel color image segmentation approach based on neutrosophic and modified fuzzy c-means. *Circuits Syst. Sig. Process.* **32**(4), 1699–1723 (2014)
10. Balasubramaniam, P., Ananthi, V.P.: Segmentation of nutrient deficiency in incomplete crop images using intuitionistic fuzzy c-means clustering. *Nonlinear Dyn.* **83**(1), 849–866 (2016)
11. Mujica-Vargas, S., Gallegos-Funes, F.J., Rosales-Silva, A.J.: A fuzzy clustering algorithm with spatial robust estimation constraint for noisy color image segmentation. *Pattern Recogn. Lett.* **34**(4), 400–413 (2013)
12. Nadernejad, E., Sharifzadeh, S.: A new method for image segmentation based on fuzzy c-means algorithm on pixonal images formed by bilateral filtering. *Sig. Image Video Process.* **7**(5), 855–863 (2013)
13. Khan, A., Ullah, J., Jaffar, M.A., Choi, T.S.: Color image segmentation: a novel spatial fuzzy genetic algorithm. *Sig. Image Video Process.* **8**(7), 1233–1243 (2014)
14. Khan, A., Jaffar, M.A., Choi, T.S.: SOM and fuzzy based color image segmentation. *Multimedia Tools Appl.* **64**(2), 331–344 (2013)
15. Omran, M., Salman, A., Engelbrecht, A.P.: Dynamic clustering using particle swarm optimization with application in image segmentation. *Pattern Anal. Appl.* **8**(4), 332–344 (2006)
16. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn. Prentice Hall, Upper Saddle River (2002)
17. Liu, Z., Song, Y.Q., Chen, J.M., Xie, C.H., Zhu, F.: Color image segmentation using nonparametric mixture models with multivariate orthogonal polynomials. *Neural Comput. Appl.* **21**(4), 801–811 (2012)
18. Ito, S., Yoshioka, M., Omatu, S., Kita, K., Kugo, K.: An image segmentation method using histograms and the human characteristics of HSI color space for a scene image. *Artif. Life Robot.* **10**(1), 6–10 (2006)
19. Kohonen, T.: The self-organizing map. *Proc. IEEE* **78**(9), 1464–1480 (1990)