

Color Image Segmentation by Means of a Similarity Function

Rodolfo Alvarado-Cervantes, Edgardo M. Felipe-Riveron*,
and Luis P. Sanchez-Fernandez

Center for Computing Research, National Polytechnic Institute, Juan de Dios Batiz w/n,
Col. Nueva Industrial Vallejo, P.O. 07738, Mexico
Tel.: (52)-55-5729 6000; Ext. 56515
ateramex@gmail.com, edgardo@cic.ipn.mx, lsanchez@cic.ipn.mx

Abstract. An interactive, semiautomatic image segmentation method is presented which, unlike most of the existing methods in the published literature, processes the color information of each pixel as a unit, thus avoiding color information scattering. The process has two steps: 1) The manual selection of few sample pixels of the color to be segmented, 2) The automatic generation of the so called *Color Similarity Image* (CSI), which is a gray level image with all the tonalities of the selected color. The color information of every pixel is integrated by a similarity function for direct color comparisons. The color integrating technique is direct, simple, and computationally inexpensive. It is shown that the improvement in quality of our proposed segmentation technique and its quick result is significant with respect to other solutions found in the literature.

Keywords: Color image segmentation; Adaptive color similarity function; HSI parameter distances; Morphology in color images.

1 Introduction

Image segmentation consists of partitioning an entire image into different regions, which are similar in some predefined manner. Segmentation is an important feature of human visual perception, which manifests itself spontaneously and naturally. It is also one of the most important and difficult tasks in image analysis and processing [2] [6] [8] [9] [10]. All subsequent steps, such as feature extraction and objects recognition depend on the quality of segmentation. Without a good segmentation algorithm, objects of interest in a complex image are difficult (often impossible) to recognize using automated techniques [1] [2] [7] [8] [10]. At present, several segmentation techniques are available for color images, but most of them are just monochromatic methods applied on the individual planes in different color spaces where the results are combined later in different ways [5]. Their common problem is that when the color components of a particular pixel are processed separately the color information is so scattered in its components that most of the color information is lost [2] [5] [9].

In this work, an interactive, semiautomatic image segmentation method is presented which, in contrast with most of previously published algorithms, uses the color

* Corresponding author.

information for each pixel as a whole, thus avoiding color information scattering. In our method, the three color components (RGB) of every pixel transformed to the HSI color model are integrated in two steps: in the definitions of distances in hue, saturation and intensity planes $[\Delta_h, \Delta_s, \Delta_i]$ and in the construction of an adaptive color similarity function that combines these three distances assuming normal probability distributions.

To obtain a consistent color model for direct color comparisons, some simple but important modifications to the classical HSI color space were necessary. These modifications eliminated the discontinuities occurring in the red hue (in 0 and 360 degrees) and all the problems associated with them.

The segmentation method proposed basically relies on the calculation of a color similarity function for every pixel in a RGB 24-bit true color image, its automatic thresholding and finally the possible application of some simple morphological filters to introduce geometric characteristics in some cases where it is needed.

2 Previous Works

There has been a considerable amount of research dedicated to the problem of color image segmentation due to its importance and potential, and because color is an effective and robust visual cue for differentiating between objects in an image. The current available techniques and approaches vary widely from extensions of classical monochromatic techniques to mathematical morphology [2], clustering schemes [4] [12], wavelets [3] and quaternions [11], among others. Until recently, the majority of published approaches were based on monochromatic techniques applied to each color component image in different color spaces, and in different ways to produce a color composite [5].

Some color similarity measures and distances are presented in [10]. All these measures compare color pixels as units. They are all based in three dimensional vector representations of color in which each vector component corresponds to the RGB color channels components.

A technique that combines geometrical and color features for segmentation extending concepts of mathematical morphology (for gray images) is developed in [2] to process color images. The final segmentation is obtained by fusing a hierarchical partition image and a text/graphic finely detailed image.

In [15] the authors present a mathematic and physic solid framework for the local measure of texture in color images. They present a physic based color model using as a starting point three dimensional energy density functions $E(x, y, \lambda)$. From these energy density functions they derive color texture measures in the wavelength – Fourier domain using Gaussian derivative apertures integrating in this way texture and color information. In their implementation they start with RGB images transforming them to an opponent Gaussian color space $(E, E_\lambda, E_{\lambda\lambda})$ by a linear transform where they process with each channel separately with a set of Gabor filters and integrate later the results.

3 Description of the Method

The segmentation method proposed in this paper basically relies on the calculation of a color similarity function for every pixel in a RGB 24-bit true color image to form what we call a *Color Similarity Image* (CSI), which is a gray level image. A true color image usually contains millions of colors and many thousands of them represent the same perceived color of a single object due to the presence of additive noise, lack of definition between color borders and regions, shadows in the scene, etc., [1] [8] [10]. The color similarity function proposed allows the clustering of the many thousands colors representing the same perceived color in a single gray output image. This CSI image is then automatically thresholded and the output can be used as segmentation layer, or it can be used with morphological operators to introduce geometric enhancements if they are needed.

Firstly, we compute the color centroid and color standard deviation of a small sample consisting of few pixels. The computed centroid represents the desired color to be segmented using the technique we designed for that purpose.

Then, our color similarity function uses the color standard deviation calculated from the pixel sample to adapt the level of color scattering in the comparisons. The result of a particular similarity function calculation for every pixel and the color centroid (meaning the similarity measure between the pixel and the color representative value) generates the CSI. The generation of this image is the basis of our method and preserves the information of the color selected from the original color image. This CSI is a digital representation of a normalized function [0 - 1] extended to the range of [0 - 255].

The CSI can be thresholded with any automatic method like Otsu's [13] which was our selection for the results presented in this work. In those cases where color is a discriminating characteristic of objects of interest in a source image, only thresholding the CSI could be necessary to complete the segmentation.

To generate a CSI we need: 1. A color image in RGB 24-bit true color format; and 2. A small set of arbitrarily located pixels forming a sample of the color desired to be segmented. From this sample of pixels we calculate the statistical indicators according to our HSI modified color model (see Section 3.1). This information is necessary to adapt the color similarity function in order to obtain good results. To obtain the CSI we calculate for every pixel (i, j) in the image the following color similarity function S :

$$S_{i,j} = e^{\left(\frac{-\Delta_h^2}{2\sigma_h^2}\right)} * e^{\left(\frac{-\Delta_s^2}{2\sigma_s^2}\right)} * e^{\left(\frac{-\Delta_i^2}{2\sigma_i^2}\right)} \quad (1)$$

where Δ_h is the hue Euclidean distance between $hue(i, j)$ and the *average_hue* ; Δ_s is the saturation Euclidean distance between $saturation(i, j)$ and the *average_saturation* ; Δ_i is the intensity Euclidean distance between $intensity(i, j)$ and the *average_intensity* ; σ_h is the hue standard deviation of the sample; σ_s is the Saturation standard deviation of the sample; σ_i is the Intensity standard deviation of the sample. In Eq. (1) the color information is integrated giving high importance to

perceptual small changes in hue, as well as giving wide or narrow tolerance to the intensity and saturation values depending on the initial sample, which is representative to the desired color to be segmented.

The common disadvantages attributed to the cylindrical color spaces such as the irremovable singularities of hue in very low saturations or the periodical nature of hue [5] (which is lost in its standard representation as an $angle \in [0^\circ, 360^\circ]$) are overcome in our technique using vector representation in \mathfrak{R}^2 in the separation of chromatic and achromatic regions, and in the definition of the Δ_h, Δ_s and Δ_i distances.

Two modifications on standard HSI color space were necessary in order to create a consistent model to represent colors and color centroids:

1. *Representation of hue.* Instead of standard representation of hue as an angle in the range $[0^\circ-360^\circ]$, hue is represented here as a normalized vector in \mathfrak{R}^2 (with magnitude 1 or 0). This representation has at least three advantages compared to an angle in the range $[0^\circ-360^\circ]$ a) the existing discontinuity in 360 and 0 degrees is eliminated; b) the average hue of a group of pixels can be understood as the resulting angle of a vector addition of the color pixels in the chromatic region of the sample, giving a simple manner to calculate the average hue; c) setting magnitude to 0 or 1 works as a flag intended for distinction between chromatic or achromatic regions.

2. *Separation of chromatic and achromatic regions.* We use a separation of the region as described in [10] in order to calculate the average hue and Δ_h . Once calculated Δ_h, Δ_s and Δ_i this distinction is no longer necessary because in the formulation of $S_{i,j}$ (Eq. 1) all the cases of color comparison between zones are accounted for and it is a simple matter to maintain consistency. The use of Gaussians in the definition of $S_{i,j}$ (Eq. 1) reflects our belief that the color model modifications proposed in this paper allows normal distributions of the color characteristics in this modified HSI space according to the visual experience of color similarity.

The pixel sample is a representation of the desired color(s) to be segmented from a color image. From this pixel sample we obtain two necessary values to feed our segmentation algorithm: the color centroid and a measure of the dispersion from this centroid, in our case the standard deviation. These two values are represented accordingly to our modified HSI model.

The achromatic zone G is the region in the HSI color space where no hue is perceived by humans. This means that color is perceived only as a gray level because the color saturation is very low or intensity is either too low (near to black) or too high (near to white).

Given the three-dimensional HSI color space, we define the achromatic zone G as the union of the points inside the cylinder defined by *Saturation* $< 10\%$ of MAX and the two cones *Intensity* $< 10\%$ of MAX and *Intensity* $> 90\%$ of MAX, were MAX is the maximum possible value as presented in [10]. Pixels inside this region are perceived as gray levels.

3.1 Calculation of Average Hue

In order to obtain the average of the hue (H_m) of several pixels from a sample, we take advantage of the vector representation in \mathfrak{R}^2 . Vectors that represent the hue values of individual pixels are combined using vector addition. From the resulting vector we obtain the average hue corresponding to the angle of this vector respected to the red axis. Thus H_m is calculated in the following manner:

1. For every pixel $P(x, y)$ in the sample the following \mathfrak{R}^3 to \mathfrak{R}^2 transformation is applied:

$$V_1(P) = \begin{bmatrix} 1 & -\cos(\pi/3) & -\cos(\pi/3) \\ 0 & \sin(\pi/3) & -\sin(\pi/3) \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \quad \text{If } P \notin G \quad (2)$$

and $V(P) = V_1(P) / |V_1(P)|$;

In other case:

$$V(P) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{If } P \in G$$

where $V(P)$ is the normalized projection of the RGB coordinates of the pixel P to the perpendicular plane to the Intensity axis of the RGB cube when the x axis is collinear to the Red axis of the chromatic circle. On the other hand G (see Section 3) represents the achromatic zone in the HSI space and $[RGB]^t$ is a vector with the color components of the pixel in the RGB color space.

To carry out this, the following code is executed:

```
Vector.x = 0;
Vector.y = 0; // initialize vectors
For (i = 1; i <= n; i++) // for every pixel in
                        // the sample do
    {Vector.x = Vector.x + V(i).x; // x-component of the
      Vector.y = Vector.y + V(i).y;} // y-component of the
                                     accumulated vector
Vs = [Vector.x Vector.y]; // Accumulated vector
```

In this code we have a vector in \mathfrak{R}^2 , which accumulates the vector additions as index i increments. Each of the vectors being added corresponds to the previous \mathfrak{R}^3 -to- \mathfrak{R}^2 transformation for every pixel in the sample made in step 1.

2. The angle of the accumulated vector (V_s) with respect to the X-axis is the average hue:

$$H_m = \text{angle}(V_s, 0)$$

where 0 represents the Red axis.

Using the vector representation of Hue obtained by the \mathfrak{R}^3 -to- \mathfrak{R}^2 transformation of RGB space points expressed in Eq. (2), we can calculate the hue distance Δ_h between two colors pixels or color centroids C_1 and C_2 , as follows:

$$\begin{aligned} \Delta_h(C_1, C_2) &= |V_1 - V_2| && \text{If } C_1 \text{ and } C_2 \notin G \\ &= 0 && \text{If } C_1 \text{ or } C_2 \in G \end{aligned}$$

where G is the achromatic region; V_1 and V_2 are the vectors in \mathfrak{R}^2 calculated with the transformation on C_1 and C_2 given in Eq. (2).

Using the standard conversions for saturation and intensity from RGB space [10], normalized in the range [0, 1]:

$$saturation(P) = 1 - \left[\frac{3}{R + G + B} \min(R, G, B) \right]. \tag{3}$$

$$intensity(P) = \frac{1}{3}(R + G + B).$$

we define saturation distance Δ_s and intensity distance Δ_i between two pixels or color centroids as:

$$\begin{aligned} \Delta_s &= abs[saturation(C_1) - saturation(C_2)], \text{ and} \\ \Delta_i &= abs[intensity(C_1) - intensity(C_2)], \end{aligned}$$

where C_1 and C_2 are color pixels or color centroids, respectively, in RGB space.

In Eq. (3) we defined the saturation equal zero in case of the black color.

The statistical values needed in Eq. (1) are calculated as follows [14]:

$$Saturation_average = S_c = \frac{1}{n} \sum_{i=1}^n saturation(i).$$

$$Intensity_average = I_c = \frac{1}{n} \sum_{i=0}^n intensity(i).$$

$$Hue_standard_deviation = \sigma_h = \sqrt{\frac{\sum_{i=1}^n \Delta_h^2(i)}{n}}.$$

$$Saturation_standard_deviation = \sigma_s = \sqrt{\frac{\sum_{i=1}^n \Delta_s^2(i)}{n}}.$$

$$Intensity_standard_deviation = \sigma_i = \sqrt{\frac{\sum_{i=1}^n \Delta_i^2(i)}{n}}.$$

where n is the number of pixels in the sample; Δ_h is the hue distance between $hue(i, j)$ and $hue_average$; Δ_s is the saturation distance between $saturation(i, j)$ and $saturation_average$; Δ_i is the intensity distance between $intensity(i, j)$ and $intensity_average$.

4 Results and Discussion

In this section we present the results of our segmentation method applied to two classical color images in RGB 24-bit true color format that are representative of many image processing and analysis applications. These experiments consisted of the segmentation of color regions according to the following three steps:

1) Selection of the pixel sample. In order to have a helping direction for this task the following considerations may be useful to select the number of pixels of the sample: If the color of the desired area to segment is solid (without additive noise) it is only necessary to have one pixel sample from the desired area. However, if we want to take in account the color lack of definition happening in the borders, we have to take a sample of the new colors that appear in that area due to the above condition. The pixels of the samples from the original images can be selected arbitrarily, that is, in any order, in any number and physically adjacent or not.

2) CSI calculation. This step is automatic; its output is a gray level image showing the similarity of each pixel of the RGB true color image to the color centroid formed with the chosen pixel sample taken from of the region of interest to be segmented, being white for 100% of similarity and black for 0%.

3) Thresholding and application of mathematic morphology. The user can threshold now the CSI and could be arranged as an automatic step by using, for example, the non-supervised Otsu's thresholding method [13]. After this step we can apply optionally any desired morphological tool if the thresholding results are not solid enough or geometric characteristics are needed to correctly separate the objects of interest.

Figure 1 shows a RGB color image (sized 200 x 200 pixels and with 33753 different colors) of the popular image of the baboon. In this image we can see four main hues of colors despite the many thousands of actual RGB values to represent them: The red part of the baboon's nose, the blue part of the nose, the orange eyes and the yellow-orange part of the fur.

Different pixel tonalities in the image depend on their particular saturation and on the unavoidable presence of additive noise. The proposed segmentation method is

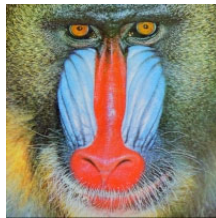


Fig. 1. Baboon

practically immune to these conditions, although obviously there are some solutions to improve the quality of the segmented regions, as for example, preprocessing the image for smoothing noises of different types, applying some morphological operator to reduce objects with given characteristics, and so on.

In this experiment we took pixel samples for the blue color belonging to the edge of the perceived blue color. They are selected from an enlarged 21 x 21 pixels region as shown in Fig. 2. From this sample we calculated the color centroid and the standard deviation in our modified HSI space; with these two values we use the Eq. 1 to calculate for every pixel the pixel values of the CSI shown in Fig. 3. After applying Otsu's thresholding method and an area closing with a box of 3x3 to eliminate small holes, we obtain the final segmentation shown in Fig. 4.

For the blue part of the nose we repeated part of the process. Figure 5 shows the pixels sample, its corresponding CSI is shown in Fig. 6 and after applying Otsu's thresholding, an opening with a box of 3x3, a closing with the same box and eliminating remaining small areas, the final segmentation is shown in Fig. 7.

In Fig. 8 we show the pixel sample and in Fig. 9 the CSI for the orange color of the eyes. After thresholding the CSI, applying an opening with a disk 5x5 and eliminating the big area of the fur we obtain the final segmentation of the eyes shown in Fig. 10. The yellow-orange part of the fur shown in Fig. 11 was obtained as residue from the thresholding of the CSI and shown together in the composite of the segmentations of Fig. 4, 7 and 10.



Fig. 2. Sample composed by 5 pixels located in two zones with red color

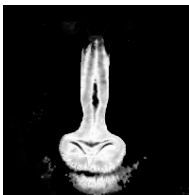


Fig. 3. The Color Similarity Image (CSI) of red

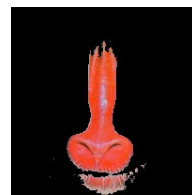


Fig. 4. Final segmentation of the red nose



Fig. 5. Pixel sample for the blue nose

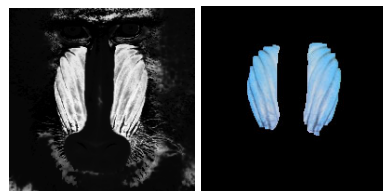


Fig. 6. CSI Fig. 7. Final segmentation



Fig. 8. Pixel sample for the eyes

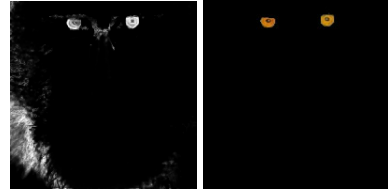


Fig. 9. CSI Fig. 10. Final segmentation

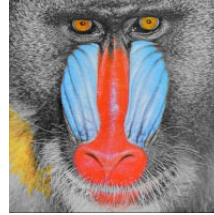
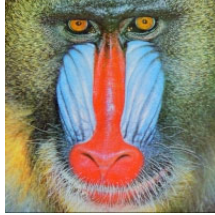


Fig. 11. Original image and a composite image achieved from the four previously segmented regions

5 Conclusions

The results in the previous section, demonstrate that the color segmentation method presented in this paper offers a useful and efficient alternative for the segmentation of objects with different colors in relatively complex color images with good performance in the presence of the unavoidable additive noise. The steps required to obtain a good segmentation of regions with different colors by using the proposed methodology are usually straightforward, simple and repetitive. If color is a discriminative characteristic in the layer of interest, only the selection of a given threshold to the color similarity function CSI is needed to obtain a good segmentation result. From many experiments we have observed that a good percentage of colors were obtained in a straightforward way only by thresholding the so called Color Similarity Image. In our method, the three RGB color components of every pixel transformed to the HSI color model are integrated in two steps: in the definitions of the Euclidean distances $[\Delta_h, \Delta_s, \Delta_i]$ in hue, saturation and intensity planes and in the construction of an adaptive color similarity function that combines these three distances assuming normal probability distributions. Thus the complexity is linear ($O[n]$) with respect to the number of pixels n of the source image. The method discriminates whichever type of different color objects independently on their shapes and tonalities in a very straightforward way.

Acknowledgments. The authors of this paper wish to thank the Computing Research Center (CIC), Mexico; Research and Postgraduate Secretary (SIP), Mexico, and National Polytechnic Institute (IPN), Mexico, for their support.

References

1. Alvarado-Cervantes, R.: Segmentación de patrones lineales topológicamente diferentes, mediante agrupamientos en el espacio de color HSI, M. Sc. Thesis, Center for Computing Research, National Polytechnic Institute, Mexico 1 (2006)
2. Angulo, J., Serra, J.: Mathematical morphology in color spaces applied to the analysis of cartographic images. In: Proceedings of International Congress GEOPRO, México (2003)
3. Bourbakis, N., Yuan, P., Makrogiannis, S.: Object recognition using wavelets, L-G graphs and synthesis of regions. *Pattern Recognition* 40, 2077–2096 (2007)
4. Chang, H., Yeung, D.Y.: Robust path-based spectral clustering. *Pattern Recognition* 41, 191–203 (2008)
5. Cheng, H., Jiang, X., Sun, Y., Wang, J.: Color image segmentation: Advances and prospects. *Pattern Recognition* 34(12), 2259–2281 (2001)
6. Felipe-Riverón, E.M., García-Ramos, M.E., Levachkine, S.P.: Problemas potenciales en la digitalización automática de los mapas cartográficos en colores. In: Proceedings of International Congress on Computation CIC IPN, Mexico City, Mexico (2000)
7. Felipe-Riverón, E.M., García-Ramos, M.E.: Enhancement of digital color halftoning printed images. In: Proceedings of International Congress GEOPRO, México (2002) ISBN: 970-18-8521-X
8. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 3rd edn. Prentice Hall, USA (2008)
9. Hanbury, A., Serra, J.A.: 3D-polar coordinate colour representation suitable for image analysis, Technical Report PRIP-TR-77, Austria (2002)
10. Plataniotis, K.N., Venetsanopoulos, A.N.: *Color Image Processing and Applications*, 1st edn. Springer, Germany (2000)
11. Shi, L., Funt, B.: Quaternion color texture segmentation. *Computer Vision and Image Understanding* 107, 88–96 (2007)
12. Van den Broek, E.L., Schouten, T.E., Kisters, P.M.F.: Modeling human color categorization. *Pattern Recognition Letters* (2007)
13. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging* 13(1), 146–165 (2003), doi:10.1117/1.1631315
14. Dodge, Y.: *The Concise Encyclopaedia of Statistics*, 1st edn. Springer, Germany (2008)
15. Hoang, M.A., Geusebroek, J.M., Smeulders, A.W.: Color texture measurement and segmentation. *Signal Processing* 85(2), 265–275 (2005)