Palette-Based Image Segmentation using HSL Space

Lala Z. Krikor
Applied Science University
Faculty CSIT, Compter Science Department
Amman. Jordan
L_krekor@asu.edu.jo

Sami E. I. Baba
Applied Science University
Faculty CSIT, Compter Science Department
Amman. Jordan
samie@asu.edu.jo

Maro Z. Krikor
Applied Science University
Faculty CSIT, Compter Science Department
Bahrain
marozareh@yahoo.com

Journal of Digital Information Management

ABSTRACT: *This paper describes a palette-based colored image segmentation system. In this paper we analyzed the properties of Hue, Saturation, and Luminance (HSL) color space with emphasis on the visual perception of the variation in Hue, Saturation and Luminance values of palette colors. The proposed algorithm based on the concept of re-coloring the palette colors of 256-color images using the properties of HSL color space and segmenting true type images by taking part of the image, generating a palette to it then apply the method to segment the selected region. After re-coloring of the palette colors the colors in the image will be re-colored according to its new palette, and as a result the image will be segmented to different segments.*

Categories and Subject Descriptors
I.4.6[Image Processing and Computer Vision];Segmentation **I.4.8 [Scene Analysis];** Colour

General Terms
Image Segmentation System, Palette colour

Keywords: Image Segmentation, HSL space

Received 02 Feb. 2006; Revised and accepted 17 Aug. 2006

1. Introduction

Segmentation is the process that partitions an image into regions. It has been the subject of considerable research activity over the last three decades. Many algorithms have been elaborated for gray scale images. However, the problem of segmentation for color images, which convey much more information about objects in scenes, has received much less attention of scientific community.

Many segmentation techniques are available in the literature, and some of them have been widely used in different application problems. These techniques mainly differ in the criterion used to measure the similarity of two regions and in the strategy applied to guide the segmentation process. Most of these techniques – if not all of them – concentrate on content-based segmentation [1].

There are several approaches in image segmentation, some of these are [2]:

1. Histogram thresholding: assumes that images are composed of regions with different gray (or color) ranges, and separates it into a number of peaks, each corresponding to one region.
2. Edge-based approaches: use edge detection operators such as Sobel, Laplacian for example. Resulting regions may not be connected, hence edges need to be joined.
3. Region-based approaches: based on similarity of regional image data. Some of the more widely used approaches in this category are: Thresholding, Clustering, Region growing, Splitting and merging. Most popular methods deal with region basedalgorithms. Some of these are [3 – 6].
4. Hybrid: consider both edges and regions.

Some segmentation algorithms deal with color images are based on RGB color representation. However, RGB representation does not coincide with the vision psychology of human eyes and there is high correlation among its three components, though its convenient for display devices [7].

One of the most used color segmentation method is the segmentation in RGB Vector space [8] where this approach is straightforward. The objective of this method is to segment objects of a specific color range in an RGB image, in which a set of sample color points representative of the colors of interest, is obtain an estimate of the average color that will be segmented. The average color be denoted by the RGB is saved in a vector a. The objective of segmentation is to classify each RGB pixel in a given image as having a color in the specified range or not. In order to perform this composition, it is necessary to have a measure of similarity. One of the simplest measures is the Euclidean distance, and if z denote an arbitrary point in RGB space. We say that **z** is similar to **a** if the distance between them is less than a specified threshold D_0. Other color spaces have been considered in the literature [7-9].

In this paper, we propose a new algorithm for color image segmentation in the (Hue Saturation Luminance) HSL space, since the HSL color representation is compatible with the vision psychology of human eyes. The main thought is sorting the image palette according to HSL space.

2. HSL Color Representation

The HSL colour space has three coordinates: hue, saturation, and lightness (sometimes luminance) respectively, it is sometimes referred to as HLS. The hue is an angle from 0 to 360 degrees, typically 0 is red, 60 degrees yellow, 120 degrees green, 180 degrees cyan, 240 degrees blue, and 300 degrees magenta. Saturation typically ranges from 0 to 1 (sometimes 0 to 100%) and defines how grey the colour is, 0 indicates grey and 1 is the pure primary colour. Lightness is intuitively what it's name indicates, varying the lightness reduces the values of the primary colours while keeping them in the same ratio [10].

Hue indicates the color sensation of the light, in other words if the color is red, yellow, green, cyan, blue, magenta, ... This representation looks almost the same as the visible spectrum of light, except on the right is now the color magenta(the combination of red and blue), instead of violet (light with a frequency higher than blue):

Hue works circular, so it can be represented on a circle instead as shown in fig. 1. A hue of 360° looks the same again as a hue of 0°.

Saturation indicates the degree to which the hue differs from a neutral gray. The values run from 0%, which is no color, to 100%, which is the fullest saturation of a given hue at a given percentage.

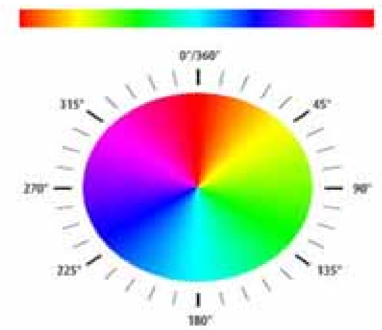


Figure 1. Color Circle of HSL Space

f illumination. The more the spectrum of the light is concentrated around one wavelength, the more saturated the color will be.

Lightness indicates the illumination of the color, at 0% the color is completely black, at 50% the color is pure, and at 100% it becomes white. In HSL color, a color with maximum lightness (L=255) is always white, no matter what the hue or saturation components are. Lightness is defined as (maxColor+minColor)/2 where maxColor is R, G or B component with the maximum value, and minColor the one with the minimum value.

2.1 Converting RGB to HSL

The conversion algorithms for these color spaces are originally from the book *Fundamentals of Interactive Computer Graphics* by Foley and Van Dam [11]. The algorithm of converting RGB to HSL is shown below:

Algorithm: RGB-to-HSL

1. Convert the RBG values to the range 0-1
2. Find min and max values of R, B, G
3. L = (*maxcolor* + *mincolor*)/2

If the max and min colors are the same (*ie* the color is some kind of grey), S is defined to be 0, and H is undefined but in programs usually written as 0

4. Otherwise, test L.

If L < 0.5, S=*(maxcolor-mincolor)/(maxcolor+mincolor)*
If L >=0.5, S=*(maxcolor-mincolor)*/(2.0-*maxcolor-mincolor*)
If R=*maxcolor*, H = (G-B)/(*maxcolor-mincolor*)
If G=*maxcolor*, H = 2.0 + (B-R)/(*maxcolor-mincolor*)
If B=*maxcolor*, H = 4.0 + (R-G)/(*maxcolor-mincolor*)

5. To use the scaling shown in the video color page, convert L and S back to percentages, and H into an angle in degrees (*ie* scale it from 0-360). From the computation in step 6, H will range from 0-6. RGB space is a cube, and HSL space is a double hexacone, where L is the principal diagonal of the RGB cube. Thus corners of the RGB cube; red, yellow, green, cyan, blue, and magenta, become the vertices of the HSL hexagon. Then the value 0-6 for H tells you which section of the hexgon you are in. H is most commonly given as in degrees, so to convert H = H*60.0

If H is negative, add 360 to complete the conversion.

3. Proposed Algorithm

If we wish to segment an image based on color, and, in addition, we want to carry out the process on individual plans, it is natural to think first of the HSL space because color is conveniently represented in the hue image. Typically, saturation is used as a masking mage in order to isolate further regions of interest in the hug image. The luminance image is used less frequently for segmentation of color images because it carries no color information.

The algorithm accepts two kind of images the paletted and un-paletted images. For the second kind the original image with the colors of interest should be enclosed by a rectangle. Then a pre step of generating a palette of that region is done to perform the segmentation on it.

The proposed segmentation algorithm depends on rearranging the palette of the image according to the hue, saturation and luminance of the colors in the palette. Divide the colors in the palette into groups (several colors will be in the same group that have the same properties); this will be illustrated in the next section. After that, a calculation of the histogram of image contents will be done; this will help to indicate which color will be the dominant in each group of the palette. Unifying the colors in each group by giving them the same color, the color is given depending on the highest histogram in the group. Display the image after the modification on the palette the image will have segments depending on the modified palette.

One can give the level of segmentation depending on the application. The proposed algorithm allows three levels of segmentation, since by extending the range of each group according to the Hue, the level of segmentation will be decreased. The user can select a color and convert the image into a binary image by applying the white color to the selected color and black to the rest of the image. The proposed segmentation method divided into three parts as shown in the following sections:

3.1 The generation palette step

In this step the user will select a palette or non-palette images, if the image is non-paletted image then the program will ask the user to select a part of that image to perform the segmentation on it, in this step the program will generate a palette for the selected part, by reading the RGB value of each pixel and generate a palette for each different combination of RGB, and then recreate that image of that region by giving the palette number instead of pixel values in the region.

3.2 Step Decision making Part:

This part is an optional part where the user can or cannot perform it; this part will return the step of the hue (which is 10, 20 or 30). This part is done by calculating the histogram of the palette colors according to their hues.

Note that the flat histogram gives an indication that the image is full of colors/details, and vice versa. So, the algorithm will give an estimation of the Hue_step used in the segmentation part. The estimation of the Hue_step is decided from the following algorithm:

```
Algorithm Step Decision Making
Input: The Original Color Palette
Output: Hue_step
   For i = 0 to 360
        HistH[Hue[i]]++
   End for
   check =0
   For i=0  to 360
        If Hist[i]<> 0 then check = check +1
   End for
```

If check>270 then step=30
If check >150 and check <270 then step =20
If check >=0 and check <=150 then step =10
End algorithm

3.3 Histogram Calculation Part

In this part the calculation of image content histogram will done according to the following algorithm:

Algorithm: Image Histogram
Input: The content of the image
Output: The Histogram of the image

Define an array His[0..255]
Initialize the value of His to zero
X = get the length of the image
Y= get the width of the image
For i= 1 to x * y
P = read a pixel from the body of the file
Increment the His[p] by one
End For
End Algorithm

3.4 Palette Re-Coloring Part:

Re-Coloring the palette depends on HSL components, namely, Hue, saturation and luminance. Colors having the same Hue may have different saturation and luminance values. For example, if we consider the red color, when the values of saturation and luminance are varied, while the hue is constant the range of red color will be as shown in the figure (2) below:

Figure 2. Where the value of Hue= 0

The basic concept applied in this paper is to group colors which have the same property. The property of the group is defined according to the Hue, saturation and luminance. The re-coloring of the palette colors is performed using the following two algorithms:

Firstly, the determination and re-coloring of (Black and dark) colors and (White and light) colors in the color palette will be done according to the luminance of the colors. This procedure is explained be the following algorithm.

Algorithm Determination and Re-Coloring of Blacks and Whites

Input: Original Color Palette List

Output: The Rest List

For each item in the palette do the following:
If the luminance value of the item is between 0 and 10 then replace its RGB values by (0, 0, 0)
Else If the luminance value of the item is between 90 and 100 then replace its RGB values by (255, 255, 255)
Else build a new list that contains the rest colors of the palette and their according positions.

End Algorithm

Secondly, separate the colors of the Rest List according to the hue as shown in the following algorithm, where the segmentation level will depend on the decision step. The step could be 10, 20 or 30, and it is determined according to the Step Decision Making algorithm, or manually by user, which will be discussed later. Also the histogram of the image contents is needed in this step and could be obtained from the Histogram calculation algorithm.

Algorithm: Separate_Rest_List

Input: The Rest List, Hue_Step

Output: Modified Original Palette

For Each item in the rest list of the color palette do the following:

Sort the items in the rest color palette according to the Hue

Separate the sorted list into groups according to the value of Hue_step.

Note: These groups may differentiate in length.

For each group do the following

Separate the items in each group into two lists according to saturation value. In the first list put the items were their saturation value is less than 50, and the others in the second list.

According to the histogram of the image obtain the dominant color in the first list, which has the highest value in histogram among other items of the list.

Substitute the RGB components of the dominant color with the RGB components of the original palette according to the positions of the other list items in the original palette.

Separate the second list according to the luminance value into two sublists. The first sublist contains the items that have luminance value less than 75. The second sublist contains the other items.

According to the histogram of the image obtain the dominant color in the first sublist.

Substitute the RGB components of the dominant color with the RGB components of the original palette according to the positions of the other first sublist items in the original palette.

According to the histogram of the image obtain the dominant color in the second sublist.

Substitute the RGB components of the dominant color with the RGB components of the original palette according to the positions of the other second sublist items in the original palette.

End Algorithm

3.5 Converting the image to a black regions and white regions

This step is optional the user can pick a color from the segmented image, then the program can convert the image into a black and white images where the colors from the same group will convert to white and the other pixels will be in black color.

4. The Results

In this section we will apply the proposed algorithm on two different images. The first one contains few image colors and/or details. The second one, have more colors/details. The user needs to select a specific region by converting the image into a black and white image.

5. Conclusion

The proposed algorithm depends on the power of color sorting, it will reduce the time of segmentation, since it will change the colorsof the palette and according to these changes the image will be segmented and only the time it takes, is the time of palette sorting and color modification.

As compared with the existing method of RGB color segmentation, it needs fewer calculations since the existing method needs to calculate the distance between the existing point and the selected color region, which is computationally expensive for images of the same size.

The size of the image is not a factor since the new method depends on sorting the color palette rather than comparing the points with a specified selected color vector. It's noted that the image of low color numbers and/or details gives a good segmentation results. So, if the image with true type or high color numbers, it is preferred to select a region to perform the segmentation on it, rather than selecting the whole image, to give the user better results. The image with high number of colors and/or details requires the largest segmentation step (i.e. Segmentation Step = 30), and vice versa.

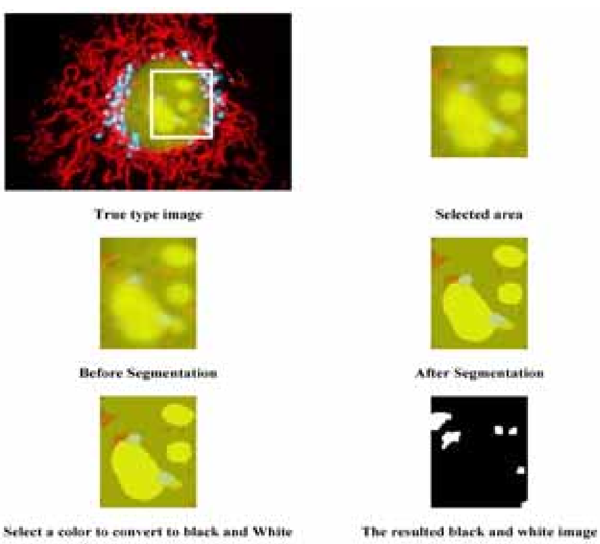
Example 1:
Segmentation step = 20 due to the low colors/details in the image

Figure 3. First example of implementing the proposed image segmentation algorithm

Example 2:
Segmentation step =30 due to high colors/details

Figure 4. Second example of implementing the proposed image segmentation algorithm

Example 3:
The segmentation of an non-palette image by selecting a part of that image:



References

1. Hearn, Donald., Baker, M. Pauline (2004). *Computer Graphics with OpenGL*, Pearson Hall.

2. http://iris.usc.edu/Vision-Notes/bibliography/contents.html

3. Tang, Yinggan., Guan, Xinping (2004). Color image segmentation using competitive and cooperative learning approach, *Int. J. of Signal Processing*, 1 (4) 312.

4. Mena, J. B., Malpica, J.A (2003). Color image segmentation using the Dempster-shafer theory of evidence for the fusion of texture, ISPRS Archives, 34, Part 3/W8, Munich, September.

5. Orlando J. Hernandez (2004). Color image segmentation using multispectral random field texture model and color content features, *JCS&T* 4 (3) 2.

6. Ety NAvon, et al. (2003). Color Image segmentation based on automatic derivation of local thresholds, Proc. 7th Digital Image Computing: Techniques and Applications, Sydney.

7. Zhang, Chi., Wnag, P (2000). *A new method of color image segmentation based on intensity and hue clustering*, 15th Int. Conf. of Pattern Recognition (ICPR'00), vol. 3., p. 3617, 2000.

8. Gonzalez, Rafael C., Woods, Richard E (2002). Digital Image Processing, Prentice Hall.

9. Sural, Shamik., Qian, Gang., Pramanik, Sakti (2002). *Segmentation and histogram generation using HSV color space for image retrieva*l, *In*: IEEE Int. Conf. on Image Processing, p. 589.

10. Vandevenne, Lode (2004). *Light and Color*, Lode's Computer Graphics Tutorial.

11. Foley and Van Dam (1982). *Fundamentals of Interactive Computer Graphic*, Addison Wesley.

Lala Z. Krikor received her BSc. Degree 1992 from Mansour University College, and her MSc. and PhD. From University of Technology (Baghdad-Iraq) in 1996 and 2000 respectively. She is now an Asst. Prof. at Computer Science Dept., Faculty of Computer Science and Information Technology in the Applied Science University, Amman Jordan. Her research interests are Image Processing and Information Hiding.

Got his BSc. Degree 1992 from Mansour University College, and his MSc. and PhD. From University of Technology (Baghdad-Iraq) in 1996 and 2000 respectively. He is now an Asst. Prof. at Computer Science Dept., Faculty of Computer Science and Information Technology in the Applied Science University, Amman Jordan. His research interests are Grammatical Inference, Adaptive Systems, and Data Hiding.