SEMANTIC OBJECT SEGMENTATION OF 3D SCENES USING COLOR AND SHAPE COMPATIBILITY

Mehran Yazdi and André Zaccarin
CVSL, Dept. of Electrical and Computer Engineering, Laval University
Ste-Foy, Québec G1K 7P4, Canada
yazdi,zaccarin@gel.ulaval.ca

ABSTRACT

A key problem in object recognition is the problem of determining which regions in the image are likely to come from a single object. In this paper, we present an approach that uses color as a cue to create initial segmentation of images containing multicolored piece-wise uniforms objects and then perform two measures of shape compatibility to merge the ones which belong to same object. Specially, the paper introduces a segmentation approach that separately uses hue and intensity to generate a semantic segmentation. More specifically, we use a region segmentation of images based on chromatic components of HSV space, and then use the intensity component to test the shape compatibility in border and inside regions to finally achieve a semantic object segmentation of images. Experiments on scenes containing 3D objects demonstrate that the proposed approach gives intuitively reasonable results.

Keywords: Image Segmentation, Line Profile Modeling, Reflectance Ratio, Semantic Object Segmentation, Surface Shape Compatibility.

1. INTRODUCTION

The preliminary problem in object recognition is the problem of identifying regions in an image by which to start the recognition process. Ideally, we want these regions in the image to come from a single object.

Model-based object recognition methods usually use geometric features such as points or edges and try to identify an object model in the database of images for which the model is present in the scene. This leads to the problem of object matching and searching, which is combinatorially explosive. It has been shown that this search process, when using an isolated stage where subsets of data features (likely to come from a single object) can be separated, is considerably reduced [1]. Thus, the isolated stage allows model-based object recognition methods to be focused on those matches that are more likely to lead to a correct solution.

Even though isolated stages can help recognition process, it has largely remained unsolved. In the ideal case, if the appearance of the desired object in the scene were known and objects in the scene were nicely separated and distinguishable from the background, and the illumination conditions were known, then methods that rely on intensity measurements would work well to extract groups of features. But in reality, not only the appearance of the objects is not known, but also illumination conditions and surface geometries of objects present in a scene can cause problems of occlusion, shadowing, specularities, and inter-reflections in the images which make it difficult to separate objects in the scene.

The previous approaches to object separation or namely object segmentation, have mostly focused on incorporating the knowledge of the object model and grouping data features such as edges, lines, points based on constraints such as parallelism, co-linearity, distance, and orientation [3] [5]. Recognition based on these grouping methods depends on the reliability of the groups produced and how many of them really come from a single object. Some others have used features solely from image data such as color and intensity and grouped them based on physical properties of the scene such as reflectance [6]. Our approach can be classified in this type of approaches.

In this paper, we propose an approach of object segmentation based on first analyzing the chromatic components of hue-saturation-value (HSV) space to segment an image into piece-wise uniform color regions, and then using its intensity component to test the compatibility in the shape of adjacent color regions. With this set, images containing multi-colored uniform dielectric objects can be segmented so that a semantic interpretation of 3D scene can be reached.

Firstly, we present an image segmentation algorithm based on color. Results of this algorithm produce adjacent color regions in the image which are likely to belong to an object in the scene. Then, two measures of intensity for shape compatibilities of adjacent regions will be proposed to merge regions. We present finally the current results on different type of scenes.

2. COLOR SEGMENTATION

Color is known to be a strong cue in attracting an observer's attention. Actually, the motivation for using color for segmentation comes from the fact that it provides region information and that when specified appropriately, it can be relatively insensitive to variation in illumination conditions and appearances of objects [8]. The standard red-green-blue (RGB) color space is not very useful for color processing, as distances in RGB space have little meaning and there is no simple mapping from RGB coordinates to human color names. A hue-based space such as HSV is superior to RGB in these respects [4]. Figure 1 shows the HSV space.

For a given point (h,s,v), h and s are the angular and radial coordinates of the point on a disk of radius v at height v. All coordinates range from 0 to 1. Points with small v are black, regardless of their h and s values. The chromatic hs and intensity v components of HSV space are well separated and this allows us to take advantage of the fact that chromatic components have less variation in a colored surface than intensity component. Actually we propose to use solely the chromatic components of HSV space to segment the images into different colored regions and then to perform a set of compatible measurements for the adjacent regions based on the variation of intensity component within these regions.

To obtain a segmentation based on chromatic components (or *HS* space), we should partition this space into subspaces where the color remains perceptually the same and it is distinctly different from that of neighboring subspaces. Such subspaces can be called color categories. Most of the categorization techniques create categories based on color space where they are independent of the image. We propose to use the histogram of each component of *HS* space in the image to perform a categorization. Figure 2 shows these histograms for a test image containing multi-colored objects.

The component h has a continuous values in HS space and thus must be treated in 3D space. In contrary, the component s has a variation from 0 to 1 and then must be treated in 2D space. In each histogram, we first apply a relaxing filter to smooth the peaks and then group the bins around the important peaks and give each group different ID (see Figure 2). A threshold is used to eliminate the local peaks which are not important in the histogram.

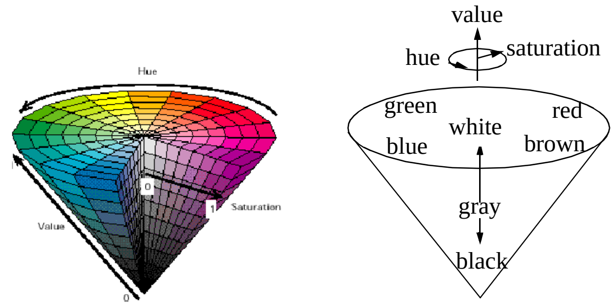


Figure 1: The HSV color space.

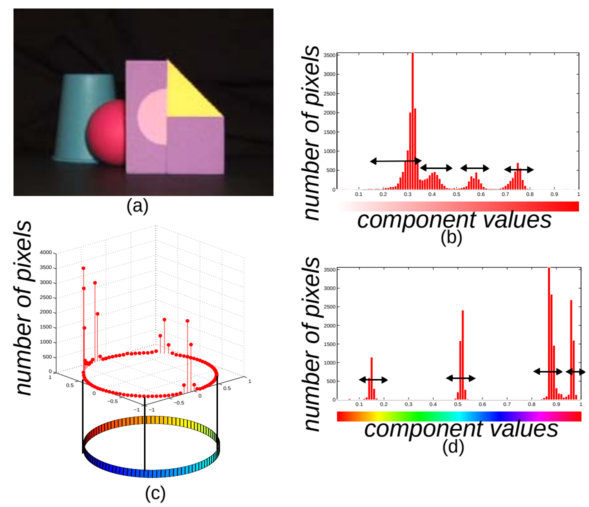


Figure 2: a) Test image, b) s-component histogram, c) 3D h-component histogram, d) 2D h-component histogram.

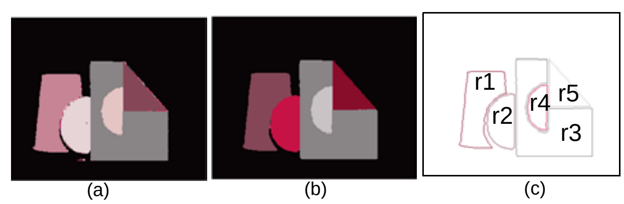


Figure 3: a) Initial segmentation, b) final segmentation by median filter, c) region borders.

After categorizing the HS space, we perform the image segmentation using a region growing method. The algorithm traverses the image in scanline order looking for seed regions where the current pixel and 4-connected neighboring have similar HS category and none of these pixels already belong to another region. When it finds a seed region, it puts the current pixel on a stack and begins a region growing process searching the pixels of same HS category. When a region has finished growing, the search for another seed region continues until all pixels in the image have been checked. In the end, all pixels in the image that are part of a region are marked with their region ID in the region map. Some pixels in the image may have a black intensity. We do not treat these pixels since their chromatic components are not reliable (see Figure 1). Some isolated pixels may be produced by this algorithm. We can classify these pixels using a relaxation filter such as the median filter.

The objective of this segmentation algorithm is to find regions that can be considered part of the same surface or object. Figure 3 shows the segmentation of the test image in Figure 2. Once the segmentation is complete, the merging process of adjacent region, using two measures of shape compatibility, begins.

3. SHAPE COMPATIBILITY IN REGION BORDERS

After identifying the image regions, we ask the following question: which region pairs could be part of the same surface? Shape of surfaces is a strong cue to test the compatibility of regions and the variation in region intensity values can be used to measure this cue.

[7] proposed a photometric invariant called reflectance ratio that can be computed from the intensity values of nearby pixels to test shape compatibility at the border of adjacent regions. Consider two adjacent colored regions r_1 and r_2. If r_1 and r_2 are part of the same piece-wise uniform object and have a different color, then the discontinuity at the border must be due to a change in albedo, and this change must be constant along the border between the two regions. Furthermore, along the border, the two regions must share similar shape and illumination. If r_1 and r_2 belong to different objects, then the shape do not have to be the same. If the shape and illumination of two pixels p_1 and p_2 are similar, then the reflectance ratio, defined in Eq. (1), where I_1 and I_2 are the intensity values of pixels p_1 and p_2, reflects the change in albedo between the two pixels [7].

$$R = \left(\frac{I_1 - I_2}{I_1 + I_2}\right) \tag{1}$$

For each border pixel p_{1i} in r_1 that borders on r_2, we find the nearest pixel p_{2i} in r_2. If the regions belong to the same object, the reflectance ratio should be the same for all pixel pairs (p_{1i}, p_{2i}) along the r_1 and r_2 border. A simple measure of constancy is the variance of the reflectance ratio. If r_1 and r_2 are part of the same object, this variance should be small (some small changes must be tolerated due to noise in the image and small-scale texture in the scene). If, however, r_1 and r_2 are not part of the same object, then the illumination and shape are not guaranteed to be similar for each pixel pair, violating the specified conditions for the characteristic. Differing shape and illumination should result in a larger variance in the reflectance ratio. Table 1 shows the variances in the border reflectance ratios of the regions pairs for the test image.

Region 1	Region 2	R	σ^2	Compatibility
r1	r2	0.2895	0.0103	no
r2	r3	0.2168	0.0032	yes
r3	r4	0.1022	0.0018	yes
r3	r5	0.1367	0.0005	yes
r4	r5	0.0020	<<0.0001	yes

Table 1: Results of shape compatibility in region borders for Figure 2. $T_{\sigma 2}$=0.0050.

Although this test has a strong ability to measure the compatibility of region shape, it only specifies which regions are definitely not compatible. Thus, all regions pairs that can potentially be merged must undergo further analysis.

4. SHAPE COMPATIBILITY INSIDE REGIONS

So far, we have discussed the shape compatibility between two adjacent regions using a measure based on the intensity values of border pixels. Now, we concentrate on the compatibility of the shape of adjacent regions by analyzing the intensity values within the two adjacent regions. Actually, if two regions are part of the same object, then the surface form of two regions must have a continuous profile. Thus we should represent the surface profile of two regions and compare them in the matter of compatibility of their form. The intensity value of pixels within the regions gives a good indication of the form of region surfaces. As a matter of fact, we have allowed all variations of pixel intensity values within the regions by solely using the chromatic components of HSV space to perform the segmentation. These variations (or intensity profiles) represent the shape of regions in the image. In general, the intensity profile of regions in the image form 3D patches and their analysis and modeling which are a challenging task are out of the subject of this work [1] [2].

Rather than observe the intensity profile in 3D case, we abstract the problem to a simpler domain by analyzing it along the lines crossing both regions. In other words, we convert the pixels to a line profile that records the pixel intensity as a function of position. To obtain the line profiles for a region pair, we take into account all pixel pairs (p_{1i}, p_{2i}) along the adjacent region borders r_1 and r_2. For each pixel pair, we then fit a line passing this pair and crossing both regions. Figure 4a shows an example of line profiles for two adjacent regions of the test image in Figure 2.

Since the path may not touch some pixels, we use a nearest neighboring interpolation to determine the intensity value of equally pixels along the path. As a result, we can calculate the intensity profile along the segment of the path correspond to each adjacent region. Figs. 4b-4c show the intensity profile calculated for a line in two adjacent regions of Figure 2.

For a complex scene containing non-uniform 3D objects, intensity profiles may have any degree of complexity and their modeling is an elaborate task to do. However, for piece-wise uniform objects, we should be able to effectively represent the intensity profiles by simple models. In this work, we present an approximate parametric approach for modeling the intensity profiles which are either straight-line segments or circular arcs. Our goal is to differentiate between these two cases and we are not searching for a precise modeling of each case (which are needed to consider highly order polynomials). Table 2 summarizes this parametric modeling in details; x_i is the position of pixel i related to the region border in the line profile, and y_i is the intensity value of pixel i in the line profile.

First, we calculate the parameters of each model for a given line profile and we then determine which model is a better match to the line profile using the minimum of mean absolute error between each model and the line profile. It is important to note that clearly a curved line can be arbitrarily close to a straight line and thus this distinction must be made by using a selected threshold. After our experiments, we found that a general threshold $T_{straight/curved}$=0.015 is appropriate for this distinction if we use the mean absolute error as a measure of distinction. Furthermore, we do not process lines that are too small or long since they cannot be reliably modeled.

Once the best model matches are determined for line profiles, we can examine the compatibility of models using some rules. Table 3 shows the compatibility of these models. While these rules may be restrictive, they guarantee that we do not create an interpretation which is not realistic. As a matter of fact, one can change these rules and reach other scene interpretations which may be also logical. Table 4 shows the results obtained on the test image of Figure 2. A strong percentage match encourage a merger of two regions. The test of shape compatibility is performed to many line profiles of a region and as a result, the test is less sensitive to the noise and is more robust. One drawback of this test tool could be that it cannot, in general, be used on small regions of an image because it violates basic assumptions necessary for the tool to function properly.

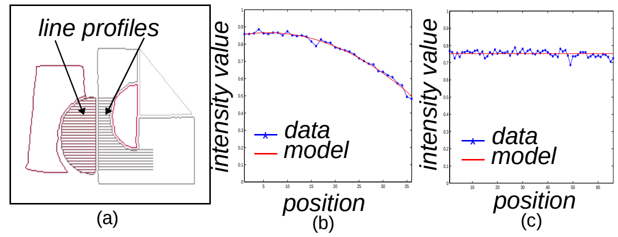
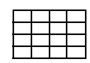
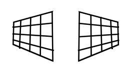
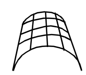
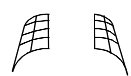


Figure 4: a) Line profiles of r2 and r3. b) example of a line profile in r2, c) example of a line profile in r3. T_{short}=10. T_{long}=70.

equivalent 3D surface model	etiquette	2D model	computing the parameters
	planar	$y = c$	$c = mean(Y)$
	planar+ planar- (*)	$y = bx + c$	$\begin{bmatrix} a \\ b \end{bmatrix} = \frac{Y}{X1}$
	curve	$y = ax^2 + bx + c$	$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \frac{Y}{X2}$
	curve+ curve- (+)	$y = ax^2 + bx + c$	$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \frac{Y}{X2}$

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad X1 = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \quad X2 = \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{bmatrix}$$

(*) planar+: the slope is going up, planar-: the slope is going down.
(+) curve+: the curve is going up, curve-: the curve is going down.

Table 2: Computing the appropriate models of straight lines and circular arcs.

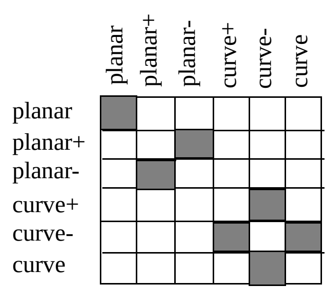


Table 3: Shaded boxes indicate potential merges. Merges are desired if the shapes match.

Region1	Region2	Dominant models	Match percentage	Compatibility
r2	r3	curve+/planar	15%	no
r3	r4	planar/planar	87%	yes
r3	r5	planar/planar	78%	yes
r4	r5	planar/planar	91%	yes

Table 4: Results of shape compatibility inside regions for Figure 2. T_m=50%.

5. RESULTS AND DISCUSION

We have tested our approach to many image scenes containing complex objects and multiple backgrounds (see Figure 5). When we deal with the background, the color segmentation based on HS space shows two interesting characteristics. First, it allows intensity variations inside the regions, such like non-intense object shadows are considered as a part of background regions. Second, the segmentation based on chromatic *s* component allows to separate the object reflections on background from the object itself. The shape compatibility tests then use these characteristics to merge the object reflection regions as a part of the background (see the third and fourth test images in Figure 5). For a complex object like the fifth test image in Figure 5, the algorithm is able to semantically merge/separate different regions of the scene. A multi background scene like in the sixth test image can be handled very well as a part of one region in the final object segmentation. The last test image in Figure 5 shows that the algorithm does not falsely merge adjacent regions belonging to different objects. Results show the performance of this approach to segment complex multiple colored objects and to handle some scene effects such as shadow, inter-reflection, and occlusion.

6. CONCLUSION AND FUTURE WORKS

We have successfully implemented an approach of object segmentation based upon an intelligently color segmenting of images and only two tests of shape compatibility of regions. Using this approach, we are able not only to segment images containing complex objects but also to handle shadow and reflection effects of objects on the background. Future works include combining this approach with object tracking methods to achieve a complete multiview representation of 3D objects using video sequences taken from the scenes containing these objects.

7. REFERENCES

[1] W. E. L. Grimson, *Object recognition by computer: the role of geometric constraints*, MIT Press, Cambridge 1990.

[2] R. M. Haralick, L. G. Shapiro, *Computer and robot vision*, Addison-Wesley, vol. 2, 1993.

[3] D. W. Jacobs, "Grouping for recognition," *Technical Report, Artificial Intelligence Lab*, MIT, AI-Memo-1177, 1989.

[4] A. K. Jain, *Fundamentals of digital image processing*, Prentice Hall, 1989.

[5] D. G. Lowe, *Perceptual organization and visual recognition*, Kluwer Academic: Boston, 1985.

[6] B. A. Maxwell and S. A. Shafer, "Physics-based segmentation: moving beyond color," *IEEE Conference on Computer Vision and Pattern Recognition*, june 1996.

[7] S. K. Nayar and R. M. Bolle, "Reflectance based object recognition," *Inter. Journal of Computer Vision*, vol. 17, no. 3, pp. 219-240, 1996.

[8] M. J. Swain and D. Ballard, "Color indexing," *Inter. Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.

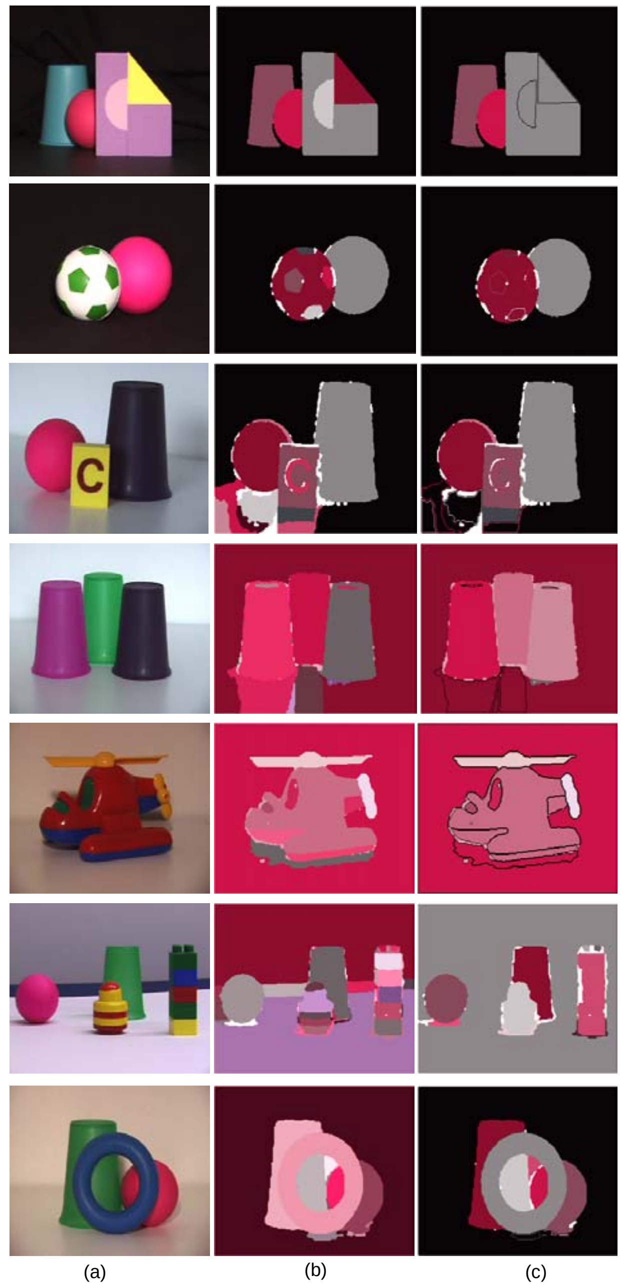


Figure 5: Results of the object segmentation algorithm for some different 3D scenes.a) original image, b) initial region segmentation, c) final object segmentation.