MULTIVIEW REPRESENTATION OF 3D OBJECTS OF A SCENE USING VIDEO SEQUENCES

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 framework to obtain a multiview representation of 3D objects in a scene is proposed. The system uses 2D image sequences of a scene from different viewpoints. The system first detects the different continuous shots and classifies them as a function of camera motion. Then, a region segmentation using color features and a region merging algorithm based on surface compatibilities are employed. A group processing is also used to ensure the robustness of the segmentation and to handle new regions. After segmentation, regions are tracked through the video sequence using motion information and extracted local features. Finally, all related object views extracted from all video sequences are used to generate a multiview representation of objects in the scene. With this integrated region/object based analysis, this framework can be used for object recognition applications.

Keywords: Image Segmentation, Multiview Representation, Region Grouping, Semantic Object segmentation, Video Sequence Segmentation.

1. INTRODUCTION

As computer-based video becomes popular with the expansion of transmission, storage, and manipulation capabilities, it offers a rich source of imagery for multimedia and computer graphics applications. However, its continuous capturing ability is not exploited in the field of object recognition and scene understanding.

Object recognition requires the capability to obtain information about many different object aspects and use this knowledge to identify an object [1]. The aspect generation is a complicated task since the real objects are three-dimensional and generally have a different appearance depending from which direction the object is seen. Problem arises when the object is present in a scene containing many objects. Besides, in a scene, occlusion, interreflection and shadow complicate more this problem. In other words, the basic problem we are addressing is how to isolate an object in the scene from other objects and to represent its different aspects related to the scene.

Our approach consists in capturing 2D image sequences of the objects from different viewpoint, and then identify and group the video segments from each object to form a multiview representation that can be used for object recognition. The assumption is made that although the appearance of a 3D object can change dramatically as it viewed from different directions, many aspects of an object can be related over a large range of viewpoints.

For a scene of different objects, if you filmed a video sequence of what you saw, from different view angles, you could subsequently register the rich information about different aspects of the scene. Furthermore, you can isolate different elements of the scene (regions or objects) and track them over different video shots. In this way, you can generate an adequate grouping of the views such that an approximate multiview representation of each object in the scene can be achieved.

This article looks at one way to use video for scene understanding and multiview representation applications. By panning and zooming a camera over a scene containing different 3D multicolored objects and automatically segmenting and classifying the video frames into connected visual contents, this system creates a multiview representation of objects in the scene. This multiview representation could include the views of an occluded and unoccluded object. Given a sequence of views, one can use any aspect graph matching methods [2] [4] [6] to find a desired element in the scene.

2. APPROACH OVERVIEW

Figure 1 presents the acquisition process. The scene containing different objects is fix and only the camera is moving to acquire the video sequences. For different angles, the camera takes a continuous shot of the scene. The result of this process is a set of different shots of a scene from different angle views. We assume that we have no a-priori information about the camera motion and the objects forming the scene. The input data, which is the 2D frame sequences of the scene, is used by the system. The system diagram is introduced in Figure 2.

First, the global motion parameters of the camera are estimated by a motion estimation algorithm. Then, an automatic image segmentation using color features will be applied to the first image of each shot to find the important regions which belong to an object surface. Region merging to obtain semantic objects of the scene is developed based on two surface compatibility tests to achieve an object segmentation. Then, these objects will be tracked over the successive frames of the shot using the motion parameters. This allows us to detect changes in an object appearance from different view points within a shot. Finally, a region matching algorithm based on color features is used to match the regions in different shots.

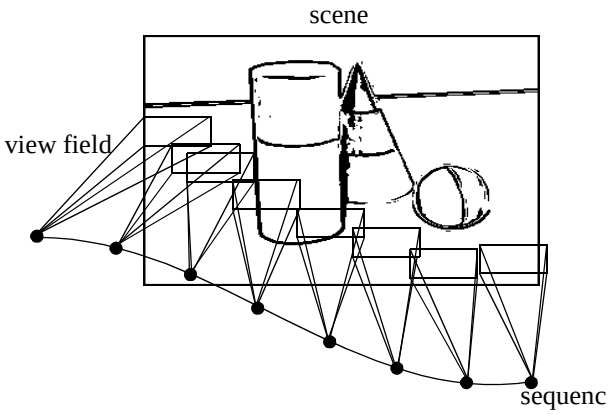


Figure 1: Acquisition system of 2D video sequences.

The result of all this processing will be integrated to an approximate appearance graph of objects extracted from the input 2D image sequences. We discuss the details of each developed algorithm in following sections.

3. VIDEO SEGMENTATION

The key for the tracking step is to detect the continuous frames of the video. It is essential task to relate the changes in the regions regarding changing the views. Our algorithm for motion estimation and classification is based on our previous work [7].

Motion estimation algorithm

In [7], we first compute the global motion of the scene between two consecutive frames. Although many different approaches have been proposed to do so, since the changes between frames is caused solely by camera motion we simply search for a translational motion between two frames. The displacement of a large region is more significant than a smaller one, so we use a block matching algorithm for camera motion compensation which is sufficient and robust. While it is possible to handle affine or projective motions, they do not give additional information and do not necessarily perform better.

Each frame is first divided into blocks and we compute the motion vectors of each block by searching the best similar block in the search area around this block. The similarity is defined as the minimum of mean square error.

Shot detection and motion classification

Motion classification is determined based on dominant vectors which are found by quantification process. Actually, motion vectors are quantified into four regions corresponding to camera displacement directions and three regions corresponding to camera speeds. Dominant motion vectors in one region determine the type of motion (pan and tilt) and (slow, fast, very fast). We also divide motion vectors into positive and negative types which correspond to the direction of motion vectors toward inside and outside the frame respectively. Dominant motion vectors in one type determine zoom in or zoom out effect. The algorithm is robust even in presence of small camera movement.

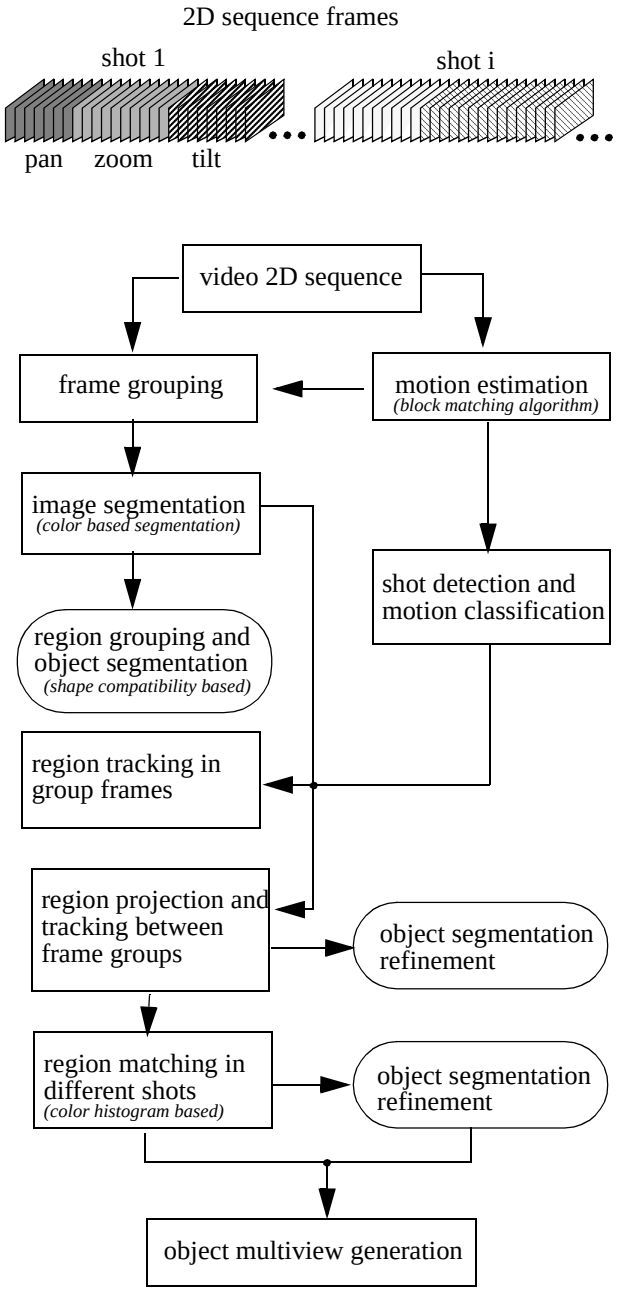


Figure 2: Overall flowchart of multiview generation system.

Once we know the global motion of the scene, we can distinguish motion from shot changes by computing a measure of overall intensity variations during shot changes. Conceptually, cuts are caused by a sudden change of pixel intensities, and thus we use as measure the number of blocks for which the mean intensity changes dramatically. Our algorithm is based on the measure of the number of blocks and their mean intensity instead of the value of pixel intensities themselves. So the algorithm is less sensitive to any little variation of intensity values. Although a cut is the only break

effect considered between shots, all other gradual effects such as fade and dissolve could also be detectable by this algorithm. Figure 3 shows an example of motion classification and shot detection results of a test video sequence.

4. OBJECT SEGMENTATION IN FIRST FRAME

First, we segment each shot into several groups of frames. The number of frames in each group is based on the motion speed calculated by the motion estimation algorithm. When the camera motion is slow, we associate more frames to a group and, when the camera motion is fast, a group has less frames. We assume that during a group of frames there is no new region appearing. In this way, frame grouping in each shot allows us to handle new regions that appear during a shot. Furthermore, we avoid the propagation of possible region segmentation errors from one group to another. We the segment the first frame of each group as described below. More details on this can be found in [8].

Region segmentation

Our algorithm for region segmentation is based on chromatic components of HSV color space, i.e. H and S. The motivation for using color for segmentation comes from the fact that it provides region information, and that it can be relatively insensitive to variation in illumination conditions and appearances of objects [5]. Also, we do not restrict the value of the intensity in segmented regions, as the intensity is used in the next step for semantic region grouping.

To obtain a segmentation based on chromatic components we partition this space into subspaces where the color remains perceptually the same and it is distinctly different from that of neighboring subspaces. To do so, we use the histogram of each component of HS space. We then segment the images 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. 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.

This segmentation finds regions that can be considered part of the same surface or object. The segmented regions include the object regions, inter-reflection regions, and background region. Figure 4 shows the segmentation of a test image in the sequence. Once the segmentation is complete, the merging process of adjacent region using two measures of shape compatibility begins.

Semantic region grouping

Shape of surfaces is a strong clue to test the compatibility of regions and the variation of intensity values of regions can be used to measure this cue.

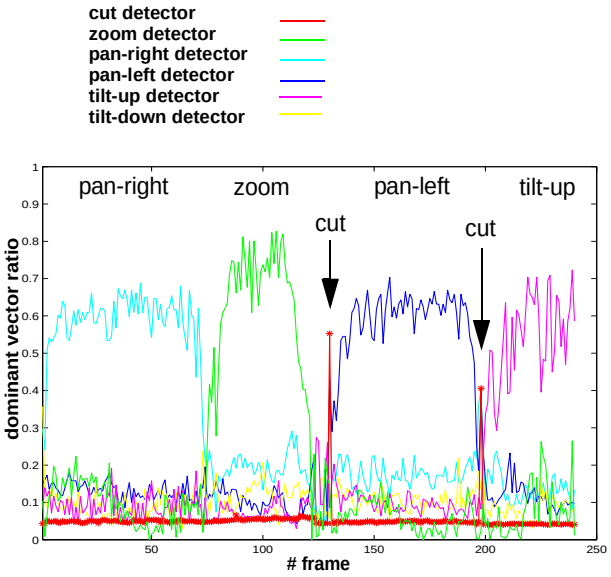


Figure 3: Results of motion classification algorithm.

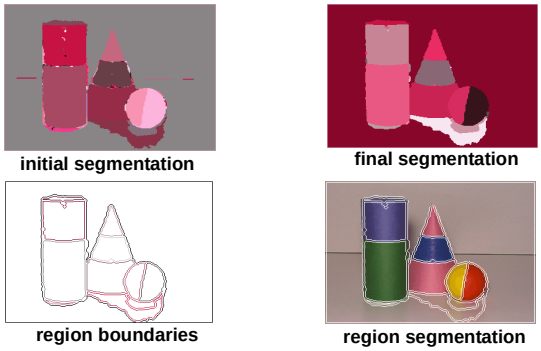


Figure 4: Results of image segmentation algorithm.

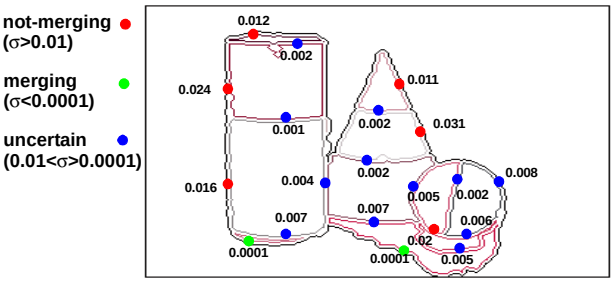


Figure 5: Variance of reflectance ratio for region pairs.

[3] proposed a photometric invariant called reflectance ratio that can be computed from the intensity values of nearby pixels to test shape compatibility in border of adjacent regions. In [3], 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 variance of the reflectance ratio. If r_1 and r_2 are part of the same object, this variance should be small. Differing

shape and illumination should result in a larger variance in the reflectance ratio. Figure 5 shows the result of this test on the image of Figure 4. Although this test has a strong ability to measure the compatibility of region shape for merging regions (σ<0.0001) and not merging regions (σ>0.01), it cannot give a definitive solution for region pairs for which the variance is between these two limits. Thus, all these region pairs must undergo further analysis. We then concentrate on the compatibility of the shape of adjacent regions by analyzing the intensity of their line profiles - if two regions are part of the same object, we assume that their surface must have a continuous profile. We use an approximate parametric approach for modeling the intensity line profiles. We then determine if two regions should be merged based on the compatibility of their respective line profile models. To do so, we use a set of restrictive rules based on our observation of scenes containing curve and flat surfaces. Figure 6 shows the results obtained on the test image. Strong percentage matches encourage a merger of two regions. The test of shape compatibility is performed in many line profiles of a region and as a result, the test is less sensitive to the noise and is more robust. The combination of the reflectance and profile tests can give a good clue of compatible regions in the scene. Figure 7 shows the final result of the semantic object segmentation of the scene.

5. REGION TRACKING IN OTHER FRAMES

Region tracking in one shot

For the first frame of each group, the system uses the intraframe segmentation described in Section 4. For the intermediate frames of a group, existing regions identified in the first frame are still valid since we made the hypothesis that no new region or any dramatic region changes occur in a group. Then, region tracking between groups is done with a projection algorithm as segmented regions and motion information are available between groups. All existing regions in the first frame of group n-1 are projected into the first frame of group n according to global motion calculated between groups. For every region in group n, if it is covered by a projected region (more than 50%) and the difference of mean color between regions (mean square error in HS color space) is below a given threshold, it is labelled as the same region. Other possible situations are as follows: if no region in group n satisfies the condition, the region in group n-1 will be considered as terminated at this point which means that the region no longer exists in other groups. If a region remains unlabeled in group n, the region is labelled as a new region - i.e. it appears in group n. We continue this tracking process for all groups of a shot and for other groups of another shots. All region changes during a shot will be detected and related in this process. Figure 8 shows two examples of region tracking during the continuous shots.

Region tracking in other shots

Tracking regions from one video shot in other shots is based on region color matching. Actually, since the region segmentation process uses chrominance components in HSV space, we also use these components to find the best region match in other shots.

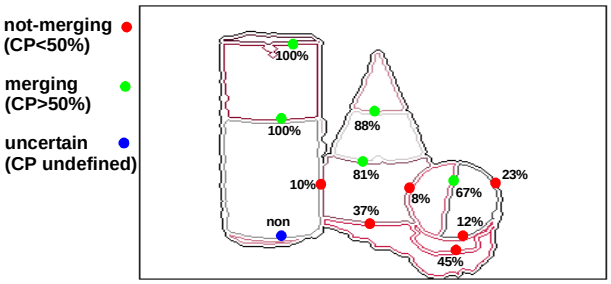


Figure 6: Compatible percentage (CP) of matching the region pairs.

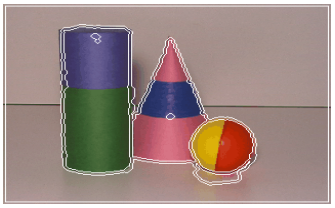


Figure 7: Result of object segmentation based on the combination of two tests.

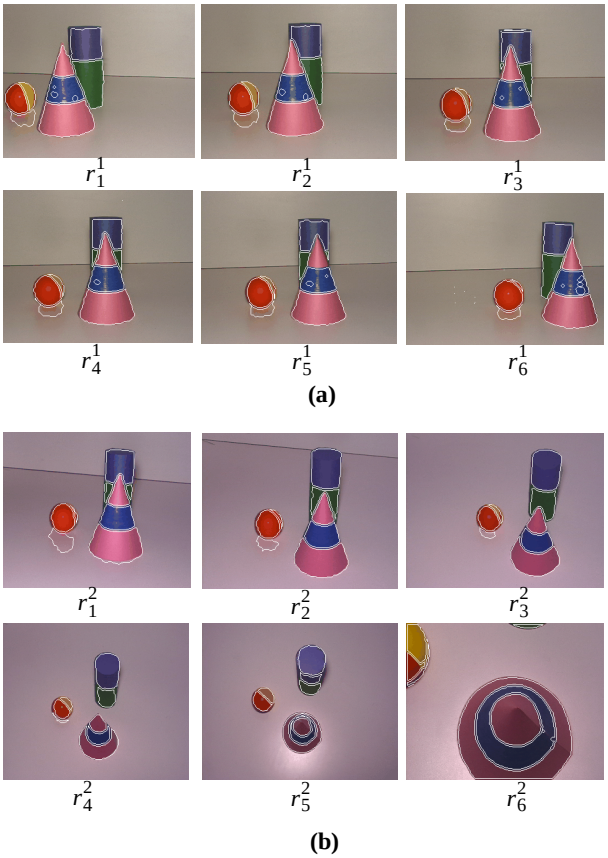


Figure 8: Examples of region tracking during two continuous shots; a) shot 1, b) shot 2.

The histograms of two chrominance components of HS space for each region in the first image of a group are extracted. Then, we compute static parameters such as mean by Eq. (1) and variance by Eq. (2) for these histograms.

$$\mu^R = \frac{1}{N}\sum_{j=1} x_j p_j \quad (1)$$

$$\sigma^R = \sqrt{\frac{1}{N}\sum_{j=1}(x_j - \mu_i)^2 p_j} \quad (2)$$

where x_j is the color value (between 0 and 1) of pixel j, p_j is the color density of pixel j in the region R and N is the total number of pixels in region R. The color difference of two regions is thus given by the Euclidean distance of statistic parameters, i.e.

$$M_{(R1,R2)} = \sqrt{(\mu_h^{R1}-\mu_h^{R2})^2 + (\sigma_h^{R1}-\sigma_h^{R2})^2 + (\mu_s^{R1}-\mu_s^{R2})^2 + (\sigma_s^{R1}-\sigma_s^{R2})^2} \quad (3)$$

where μ_h and σ_h are mean and variance of chrominance component H, μ_s and σ_s are mean and variance of chrominance component S respectively.

The matching process is continued in the following steps: First, for each region detected in a group and all its tracked regions in other groups of a shot, corresponding static parameters are computed and stored in a table. In this manner, we generate a statistic parameter table for all regions detected in video sequences. Then, we find the best match for each region using this table and color difference calculated for each region pair by Eq. (3). Table 1 shows an example of this table for the segmented regions in two images in two different shots in Figure 9. A possible case is that a region in one image has more than one similar color region in match in other image. We propose to use the region neighbor information to avoid this ambiguity. Actually, for a region, its neighboring regions and their mean reflectance ratio introduced by [3] can be used to find the best match in other images. Thus, the best match will be the one which has the minimum of difference between the mean reflectance ratios with the neighboring regions. In Eq. (4), SM is a measure of similarity between two region Ri and Rk in two images which have the same set of neighboring regions.

$$SM_{(Ri,Rk)} = \sum_{j \in G} \left| MRR^t_{(Ri,Rj)} - MRR^f_{(Rk,Rj)} \right| \quad (4)$$

where MRR is the mean reflectance ratio, t and f are two images belonging to different shots, G is the set of neighboring regions for the regions Ri and Rk, and Rj is a region belonging to the neighboring regions G.

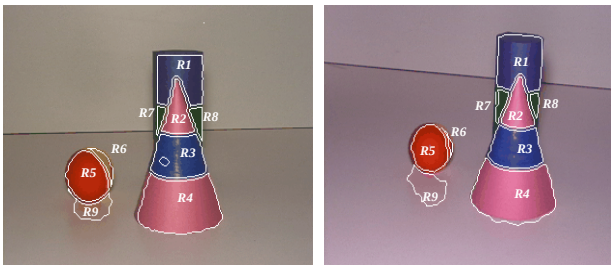
Figure 10 shows the region relationship diagram and the mean reflectance ratio between region pairs for the example images in Figure 9. We successfully find the best match for all regions.

Object segmentation refinement

In each step of region segmentation, we use the region grouping algorithm proposed in Section 4 to group the regions which belong to one object. In other steps of region tracking and matching, we refine the region grouping results to remove any possible semantic ambiguity or wrong regions grouping.

	R1	R2	R3	R4	R5	R6	R7	R8	R9	
μ_h	0.68	0.90	0.63	0.96	0.03	0.12	0.27	0.28	0.04	r_4^1
σ_h	0.01	0.21	0.02	0.06	0.01	0.01	0.09	0.07	0.01	
μ_s	0.37	0.45	0.55	0.48	0.88	0.85	0.38	0.40	0.38	
σ_s	0.04	0.06	0.11	0.04	0.06	0.16	0.08	0.06	0.07	
μ_h	0.69	0.89	0.65	0.93	0.03	0.10	0.30	0.31	0.94	r_1^2
σ_h	0.02	0.16	0.02	0.01	0.05	0.01	0.09	0.09	0.01	
μ_s	0.46	0.46	0.59	0.48	0.88	0.79	0.32	0.35	0.29	
σ_s	0.07	0.08	0.10	0.08	0.06	0.14	0.08	0.08	0.07	

Table 1: Static parameters of all segmented regions of images in Figure 9.



r_4^1 r_1^2

Figure 9: Example of region matching between two images in different shots. r^1_4 is the first image of the fourth group of first shot and r^2_1 is the first image in first group of second shot.

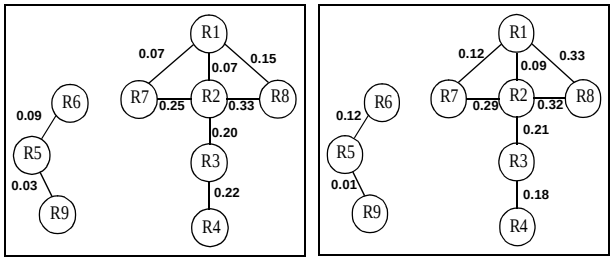


Figure 10: Region relationship diagram for two images of Figure 9.

For the refinement process, we use certain rules as follow: generally, regions belonging to one object satisfy the criteria of region grouping algorithm during all frames of a shot. However due to motion estimation errors and region segmentation errors, a region grouping may not hold in some frames. Thus, we consider that region grouping is valid for all frames of a shot if it was done for more than 75% of them.

6. RESULT OF OBJECT MULTIVIEW REPRESENTATION

Results of the integrated region/object based analysis presented in the previous sections give a set of related object views extracted from different shots. Figure 11 shows the representation of a region seen from different views. As we can

see, although the appearance of the region change from one view to another, we successfully match all the different views of this region in the scene. Figure 12 shows a multiview representation of an object in the scene. As the scene and object complexity increases, the image might contain very small regions which can cause trouble in region extraction by segmentation. However, increasing the image resolution might partially solve this problem. By experience, we found that 10-15 shots is sufficient to identify the different views of an object for simple scenes. Complex scenes need more shots, as the number of views increases due mainly to a larger number of occlusions.

7. CONCLUSION

In this article we have presented a complete framework for creating a multiview representation of 3D objects in a scene. The system uses video sequences taken from a scene containing multi-colored objects. A set of algorithms is used to segment the video sequences into continuous shots, to identify the important regions, and to group the regions belonging to one object. In this approach, a semantic object is modeled as a set of regions with compatible surface features. Finally all the segmented regions are tracked through the video sequences to find their different appearances according to different points of view.

8. REFERENCES

[1] A. K. Jain and P. J. Flynn (Eds.), 3D Object Recognition Systems, Elsevier Science Publishers B. V., Amesterdam, The Netherlands, 1993.

[2] Y. Lamdan, J. T. Schwartz and H. J. Wolfson, "On Recognition of 3-D Objects from 2-D Images", Proceedings IEEE International Conference on Robotics and Automation, pp. 1407-1413, Philadelphia, PA, 1988.

[3] S. K. Nayar and R. M. Bolle, "Reflectance Based Object Recognition," International Journal of Computer Vision, Vol. 17, No. 3, pp. 219-240, 1996.

[4] L. Shapiro, "Relational Matching", Handbook of Pattern Recognition and Image Processing: Computer Vision, T. V. Young, ed., pp. 475-496, Academic Press, 1993.

[5] M. J. Swain and D. Ballard, "Color Indexing," International Journal of Computer Vision, Vol. 7, No. 1, pp. 11-32, 1991.

[6] C. Wang and K. Abe, "Region Correspondence by Inexact Attributed Planar Graph Matching", Proceedings Fifth International Conference on Computer Vision, pp. 440-447, June 1995.

[7] M. Yazdi and A. Zaccarin, "Scene Break Detection and Classification Using a Block-wise Difference Method", IEEE International Conference on Image Processing, Vol. 3, pp. 394-397, Thessaloniki, Greece, Oct. 2001.

[8] M. Yazdi and A. Zaccarin, "Semantic Object Segmentation 3D Scenes Using Color and Shape Compatibility", The Sixth World Multiconference on Systemics, Cybernetics and Informatics, Orlando, Florida, USA, July 2002.

Figure 11: Multiview representation of a region in different points of view.

Figure 12: Multiview representation of an object in different points of view.