Object Recognition Using Appearance-Based Parts and Relations*

Chien-Yuan Huang[1], Octavia I. Camps[1,2]
[1]Department of Electrical Engineering
[2]Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA, 16802
{huang,camps}@whale.ece.psu.edu

Tapas Kanungo
Caere Corporation
100 Cooper Court
Los Gatos, CA 95030
tapas@caere.com

Abstract

The recognition of general three-dimensional objects in cluttered scenes is a challenging problem. In particular, the design of a good representation suitable to model large numbers of generic objects that is also robust to occlusion has been an stumbling block in achieving success. In this paper, we propose a representation using *appearance-based parts* and *relations* to overcome these problems. Appearance-based parts and relations are defined in terms of closed regions and the union of these regions, respectively. The regions are segmented using the MDL principle, and their appearance is obtained from collection of images and compactly represented by parametric manifolds in the two eigenspaces spanned by the parts and the relations.

1 Introduction

The recognition of general three-dimensional objects in cluttered scenes from 2D images is a challenging problem. In particular, the design of a good representation suitable to model large numbers of generic objects that is also robust to occlusion has been an stumbling block in achieving success.

A major difficulty in recognizing three dimensional objects from 2D images is that their appearances change significantly depending on the viewpoint. Common approaches to overcome this problem are to use viewer-centered representations to describe the objects in terms of their appearances, or to use object-centered representations and image invariants.

Viewer-centered approaches can be as structured as features grouped into relational models within aspect views [2, 3], or as loose as appearance-based representations [10] constructed from collections of images. A major limitation of the appearance-based approach is that it requires isolating the complete object of interest from the background, and thus it is sensitive to occlusion. In spite of the increased interest in this approach [8, 9], no satisfactory solution has been found, until now, to handle object occlusion *without sacrificing scaling.*

Approaches using object-centered representations such as part decomposition [1, 13, 7], have the potential to cope with both occlusion and large object databases. However, the definition of parts from generic objects and their image extraction remains a difficult problem[5].

Dickinson et al [4] proposed a hybrid approach where objects are described as combinations of geometric primitives that are represented using aspect graphs. This approach handles occlusion and can potentially describe a large set of objects in terms of a few primitives. However, it requires a fairly good image segmentation and it is limited to objects that can be described by the primitives in the system.

In this paper, we propose a representation using *appearance-based parts* and *relations.* Appearance-based parts and relations are defined in terms of closed regions and the union of these regions, respectively. The regions are segmented using the MDL principle, and their appearance is obtained from collection of images and compactly represented by parametric manifolds in the two eigenspaces spanned by the parts and the relations.

2 Object Representation

2.1 Parts from Images

It is commonly accepted that complex objects can be decomposed into simple parts. However, there is no much agreement on how to define what a *part* is. Several definitions have been proposed in the past, includ-

*This work was supported in part by NSF grant IRI9309100. This work was done while Kanungo was at the University of Washington.

1063-6919/97 $10.00 © 1997 IEEE

ing operational definitions (parts are what a part detector finds), view based definitions (parts are defined by local image properties), and geometric definitions (parts are defined by 3D events) [5].

We believe that a definition of a part must take into account the segmentation algorithms that will be used to extract parts from the images. In particular, we believe that a part definition should be used in the same way at the learning *and* the recognition stages. Thus, we have opted for the following definition:

> **Parts** are polynomial surfaces approximating *closed, non-overlapping* image regions that *optimally partition* the image in a *minimum description length* (MDL) sense.

We have chosen an MDL based definition for the following reasons:

1. The MDL principle has a strong theoretical grounding;
2. Using MDL does not require arbitrary parameters, and thus parts can be extracted in a consistent manner;
3. The MDL objective function can be formulated such that i) segmentations with small number of regions with smooth boundaries are favored and ii) the obtained regions are *homogeneous* in a statistical sense; and
4. Finally, algorithms implemented using fast incremental computations are available [6].

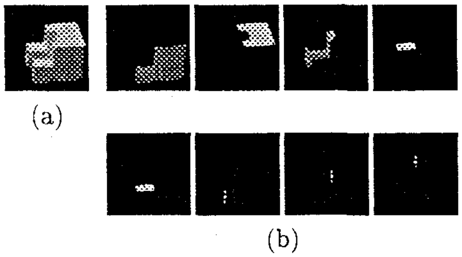


Figure 1: (a) Object "C-cube". (b) Parts obtained using an MDL-based segmentation algorithm.

The MDL objective function that we use is the one proposed in [6] encoding the region boundaries and the statistical parameters describing the data in the regions. Let $\Omega = \{\omega_j\}$ denote the image segmentation into regions $\{\omega_j\}$ and let Y represent the image data. Assuming that the image comes from a stochastic process that can be characterized as a polynomial gray scale surface plus Gaussian noise described by a vector of parameters β, then the MDL objective function to optimize is given by:

$$L(Y, \Omega, \beta) = L(\Omega) + L(\beta|\Omega) + L(Y|\Omega, \beta). \qquad (1)$$

where the first term is the length of encoding the region boundaries, the second term is the length of encoding the parameters and the last term is the length of encoding the residuals.

Figure 1(a) shows an image where the object "C-Cube" has been thresholded from the background, and Figure 1(b) shows the parts obtained using the MDL-based segmentation algorithm described in [6]. Each of the eight parts is shown in a separate image where the remaining of the object has been omitted.

2.2 Appearances of Parts

Obviously, parts obtained using the definition given above are sensor and illumination dependent. Thus, in order to completely characterize an object for different sensors and light sources, we introduce the concept of "appearances" of a part:

> Two parts segmented from two images of the same object obtained with similar sensor and illumination configurations, are said to be *appearances* of the same part if they are judged to have similar polynomial approximations in similar image locations.

This concept can be formalized as follows. Let ω_i be a part obtained from an image. Let Y_i be an $n_i \times 1$ column vector with the gray scale pixel values in part ω_i. Let d be the order of the polynomial used to fit the parts, and $m = (d+1)(d+2)/2$ be the number of polynomial coefficients. Let Φ_i be an $n_i \times m$ matrix of m basis functions for each of the n_i pixels – i.e. products of powers of pixel coordinates. Finally, let Θ_i be an $m \times 1$ column vector with the *optimal* regression coefficients for ω_i. Using these definitions, we have [6]

$$Y_i = \Phi_i \Theta_i + \Psi_i$$

where Ψ_i is a vector of zero mean Gaussian noise with covariance $\sigma^2 I$, and Θ_i is estimated by minimizing the fitting error:

$$\epsilon_i = \|Y_i - \Phi_i \Theta_i\|$$

Then, two parts ω_1 and ω_2 obtained from two images of the same object with different, but similar, sensor and illumination configurations, are considered appearances of the same part ω if

$$\epsilon_{1,2} = \frac{1}{n_1}\|Y_1 - \Phi_1 \Theta_2\| + \frac{1}{n_2}\|Y_2 - \Phi_2 \Theta_1\| \leq T_\epsilon$$

and

$$\Delta_{1,2} = \|\mu_1 - \mu_2\| \le T_\Delta$$

where μ_1 and μ_2 are the centroids of the parts and T_ϵ and T_Δ are given thresholds. Note that these thresholds can be set according to the estimated noise covariance matrix $\sigma^2 I$ and the known difference in sensor locations. Furthermore, this criteria can handle both, over and under, segmentation problems by assigning more than one part in one frame to a part in the other frame.

2.3 Collection of Appearances

The effects of the sensor and illumination configurations on the appearance of a part are learned by collecting appearances of the *same* part in sequences of images under all possible configurations. Appearances of a part can be easily tracked through frames by using the matching criteria presented in the previous section. However, a tracking algorithm must also take into consideration that due to self-occlusion, and under and over segmentation problems, a part may disappear, split into several parts or merge with others. Figure 2 represents a sequence of appearances of a part through ten different frames, $f0, f1 \ldots, f9$. The numbers between the arrows in the figure correspond to the part size number in the different frames (the larger the number, the smaller the part), and the arrows link the appearances from one frame to the next. In this example, the part being tracked splits into two parts in frame $f3$, merges back to a single part in frame $f5$, only to split again in frame $f6$ and to merge back in frame $f8$. Thus, it is fair to ask whether this part should be considered one or two parts. We have chosen the criteria that majority rules – i.e. if the number of frames where the tracked part is split is larger than half of the frames, it is decided that these are the appearances of *two* parts and that undersegmentation has occurred in the remaining frames; on the other hand if the number of frames where the part is split is less than 50% of the frames, like in this example, it is decided that it is indeed a *single* part with oversegmentation occurring at the split frames. Note, that whenever it is decided that there is a case of undersegmentation it is assumed that parts are being merged, and hence are sharing appearances in some of the frames.

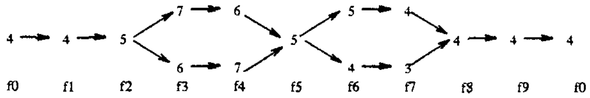


Figure 2: Example of splits and merges of appearances of a part.

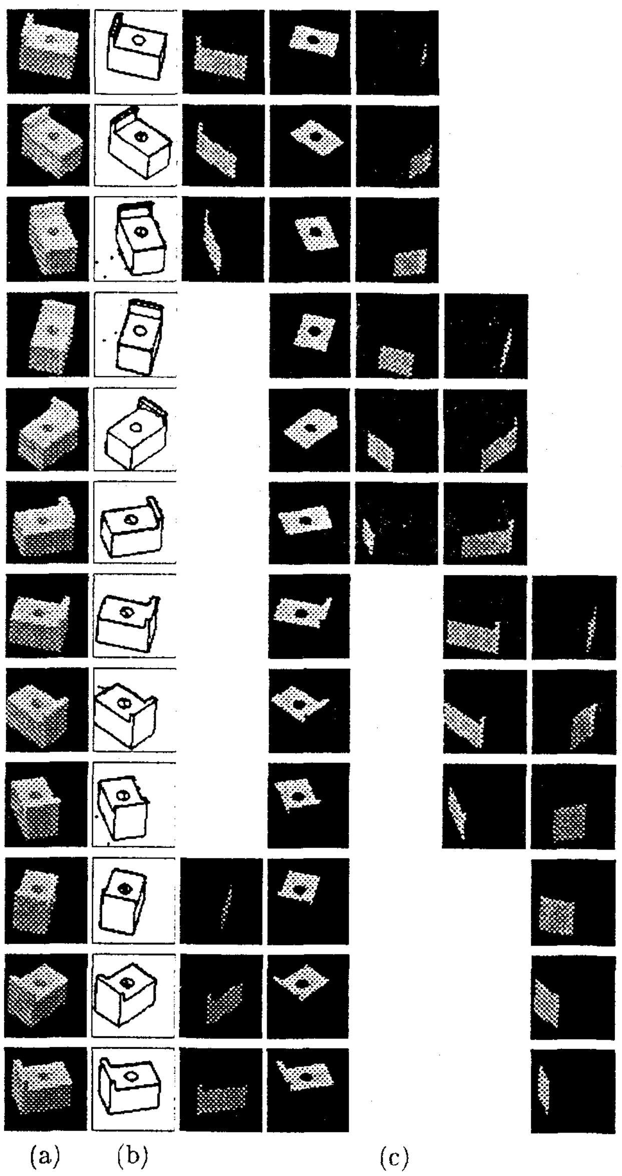


Figure 3: Collection of appearances of parts for "HoleCube". Note that four of the parts disappear for some of the frames. (a) Images of "HoleCube" every 30°. (b) MDL segmentations of the images in (a). (c) Appearances of five parts.

Figures 3 and 4 illustrate the appearances of parts of two objects, "HoleCube" and "Lamp". Figures 3(a) and 4(a) show images of these objects every 30° and Figures 3(b) and 4(b) show their respective MDL segmentations. Figures 3(c) and 4(c) show the appearances of five parts of each object. Note that due to self-occlusion, four of the parts of "HoleCube" disappear for some frames, and that due to segmentation

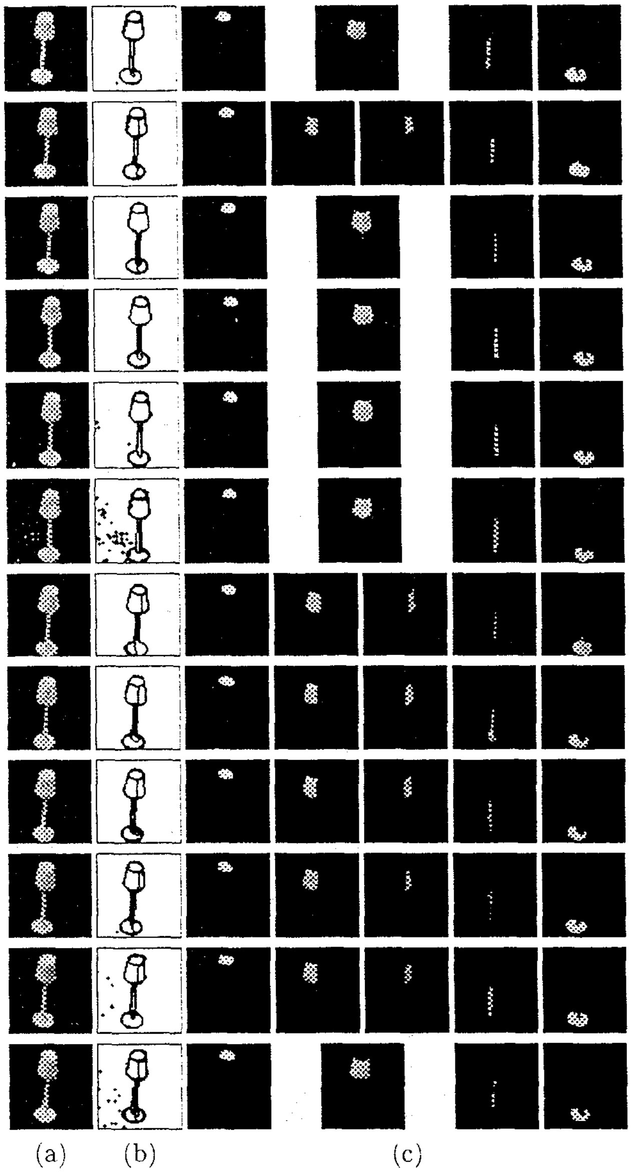


Figure 4: Collection of appearances of parts for "Lamp". Note that the second and third part share appearances in some frames. (a) Images of "Lamp" every 30°. (b) MDL segmentations of the images in (a). (c) Appearances of five parts.

problems the second and third parts of "Lamp" share appearances.

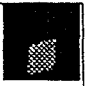
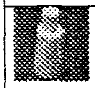
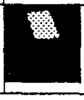
2.4 Appearance-Based Parts

The groups of appearances can be compactly stored and efficiently retrieved by constructing parametrized manifolds interpolating the projections of the individual appearances into eigenspaces obtained by applying the Karhunen-Loeve compression method [12] to a scale and brightness normalized set of appearances. These manifolds are very similar to the ones proposed in [10], which have been shown to be successful when used to recognize and locate *isolated* objects. However, until now they have been used to represent appearances of complete objects and therefore have failed in the presence of occlusion. In this paper, we propose to use this type of representation with *parts*, to take advantage of their good localization properties while addressing the occlusion problem. Formally, we define appearance-based parts:

> An **appearance-based part** (ABP) is a parametrized manifold in a space spanned by a given set of scale and brightness normalized appearances of parts, representing a collection of appearances of a part, obtained by varying the viewing conditions within a given space.

ABPs can be easily constructed with the software package SLAM [11] developed at Columbia University; it only requires to have 1) a set of appearances of parts spanning an eigenspace; and 2) a collection of appearances of parts to obtain the corresponding manifold. The set used to span the eigenspace can be chosen in many ways. It can be, for example, the set of all the collections of appearances of parts for a single or several objects. Table 1 shows representative appearances for the ABPs of three objects.

Table 1: ABPs Database Sample. The ABPs of each object are represented by one of their appearances.

Object	ABPs Representatives				
	a	b	c	d	e
					
					
					

2.5 Appearance-Based Relationships

Although it is possible to identify some objects by recognizing some of their distinctive ABPs, recognizing general objects having several "common" parts requires the use of spatial relationships between the parts being recognized. ABPs, as described earlier, are obtained by utilizing only one 2D primitive in each image to set up an eigenspace. It is also possible to use more than one 2D primitive to establish eigenspaces representing relations between regions. Appearance-based relations (ABRs) are developed by merging adjacent ABPs to create new training sets that also are represented as manifolds in the corresponding spanned eigenspace. Table 2 shows representative appearances for the ABRs of three objects.

3 Object Recognition

The ABPs and ABRs described above are the basis for our object recognition system. Let $\mathcal{ABP}$ and $\mathcal{ABR}$ represent the sets of the union of the ABPs and ABRs, respectively, for *all* the objects in a given database. Then, an object m can be represented using a relational description

$$D_m = \{R_1, R_2\}$$

where $R_1 \subseteq \mathcal{ABP}$ is a *unary* relation and $R_2 \subseteq \mathcal{ABR}$ is a *binary* (adjacency) relation. For example, the relational description of the object "HoleCube" is formed by a relation R_1 comprised of all the ABPs shown in the first row of Table 1 and a relation R_2 comprised of all the ABRs shown in the first row of Table 2.

Similarly, an MDL segmentation of an image can be described using a relational representation

$$D_i = \{S_1, S_2\}$$

where S_1 is a *unary* relation formed by a set of parts or image regions and S_2 is a *binary* relation, formed by a set of pairs of adjacent parts.

The main difference between these representations is that the description of an object is made in terms of ABPs and ABRs - i.e. collections of appearances - while the description of an image segmentation is made of a particular instance of these appearances.

ABP hypotheses are generated by projecting each segmented region into the eigenspace obtained during training, and finding the closest points on the closest manifolds to this projection. While the manifolds provide hypotheses for the part identity, the closest point on each manifold provides a hypothesis for its pose, - i.e. hypothesis for an appearance of the part.

Let f be a mapping from parts in the segmented image to appearances of parts in the model. The mapping f represents a set of ABP hypotheses. Then, if p is the projection of an image part into the ABP eigenspace, and a is the closest point on the closest manifold to p, we have

$$f(p) = a$$

The actual distances between the projections and the manifolds $d_{(p,a)} = \|f(p) - p\|$ are quantitative measures of the goodness of these hypotheses, with the smaller the distance $d_{(p,a)}$, the better the match.

ABP hypotheses with distance $d_{(p,a)} \leq T_1$, where T_1 is a small threshold can be taken as successful hypotheses. Other ABP hypotheses with somewhat larger distances $T_1 \leq d_i \leq T_2$, where T_2 is a second threshold such that $T_2 > T_1$, can be verified or discarded by composing them with the adjacency relationship. The composition of the relation S_2, with f is denoted $S_2 \circ f$ and is given by

$$\begin{aligned} S_2 \quad \circ \quad & f = \{(a_1, a_2) \text{ is a point on an ABR } \in R_2 | \\ & (a_1, a_2) \text{ is the closest point to the projection} \\ & \text{of a pair of image parts} (p_1, p_2) \in S_2, \\ & \text{with } f(p_1) = a_1 \text{ and} f(p_2) = a_2 \text{ and such that} \\ & a_1 \text{ and } a_2 \text{ are points on ABPs } \in R_1\} \end{aligned}$$

This composition takes pairs of adjacent parts in the image and maps them, part by part, into the appearance of a relationship, provided that their object hypotheses are compatible. The distance between the projection of a pair of adjacent image parts and the closest point on the closest ABR manifold $d_{(p_1,p_2)(a_1,a_2)} = \|(p_1, p_2) \circ f - (p_1, p_2)\|$ is also a quantitative measurement of the goodness of the hypothesis. Thus, an ABR hypothesis for a pair of image parts (p_i, p_j) with distance $d_{(p_1,p_2)(a_1,a_2)} \leq T_3$, where T_3 is a threshold, is said to verify the ABP hypotheses for the component parts, p_1 and p_2.

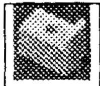
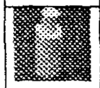
4 Experiments and Results

Figure 5 shows images of the objects in our current database. The ABP database corresponding to these objects has a total of 66 ABPs and the ABR database has a total of 80 ABRs.

Examples of cluttered scenes with busy backgrounds are shown in Figure 6. The first column shows the original image, the second column shows their MDL segmentation, and the following columns show the appearances of the ABPs and ABRs that were hypothesized and verified by the recognition algorithm. It is seen that in spite of the occlusion between the objects and segmentations problems such as the merging of some of the object parts with the background, all the objects and their pose are correctly identified.

Figures 7 (a) and (b) show images of two scenes with three objects from the database, set up on top of a rotating table. In order to study the performance of the recognition algorithm, twelve images of each scene, from different points of view, were taken by rotating

Table 2: ABRs Database Sample. The ABRs of each object are represented by one of their appearances.

Object	ABRs Representatives								
	r1	r2	r3	r4	r5	r6	r7	r8	r9
									
									

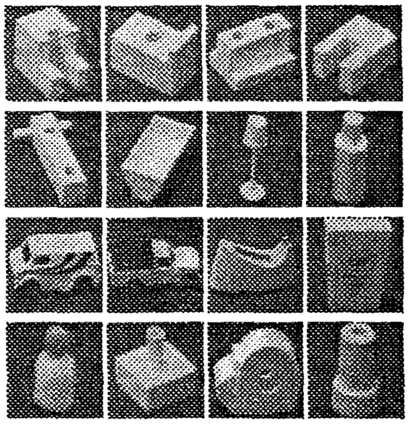


Figure 5: Object Database.

the table in increments of 30 degrees. Figure 7 (c) shows plots of the false alarms vs misdetections for the ABPs, as the threshold T_2 used to hypothesize them is varied from 0.01 to 0.1. The best results for both scenes are obtained when the threshold T_2 is 0.05. The associated probabilities of false alarm are 0.1622 and 0.1421; the probabilities of misdetection are 0.1571 and 0.036. Figure 7(d) shows plots of the false alarms and misdetections for the ABRs, as the threshold T_3 is varied while holding T_2 constant at a value of 0.05. The best threshold value for T_3 is 0.08, with probabilities of false alarm of 0.2424 and 0.2364 and probabilities of misdetection of 0.3454 and 0.3247. Finally, the plots for the probabilities of false alarm versus misdetection for the ABPs as the threshold T_1 is varied while $T_2 = 0.05$ and $T_3 = 0.08$ are shown in Figure 7(e). The best threshold value for T_1 is 0.03 resulting in probabilities of false alarm of 0.2524 and 0.236 and probabilities of misdetection of 0.301 and 0.242.

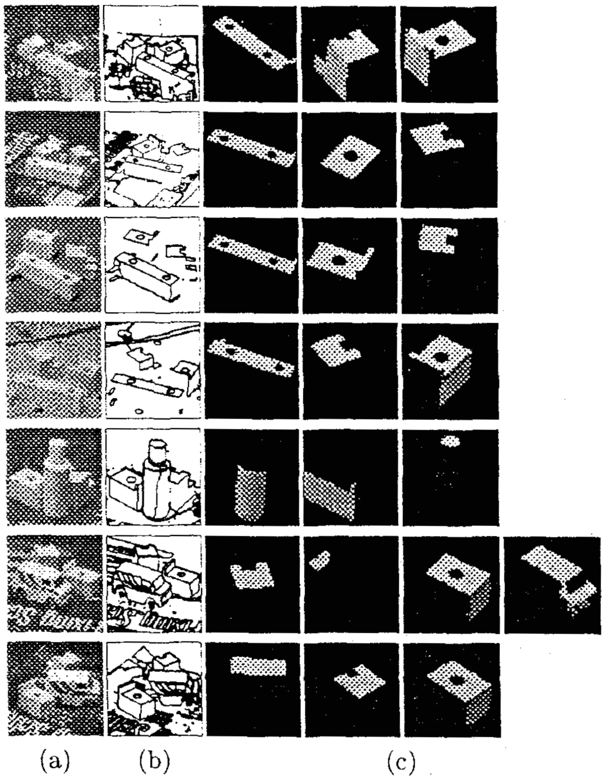
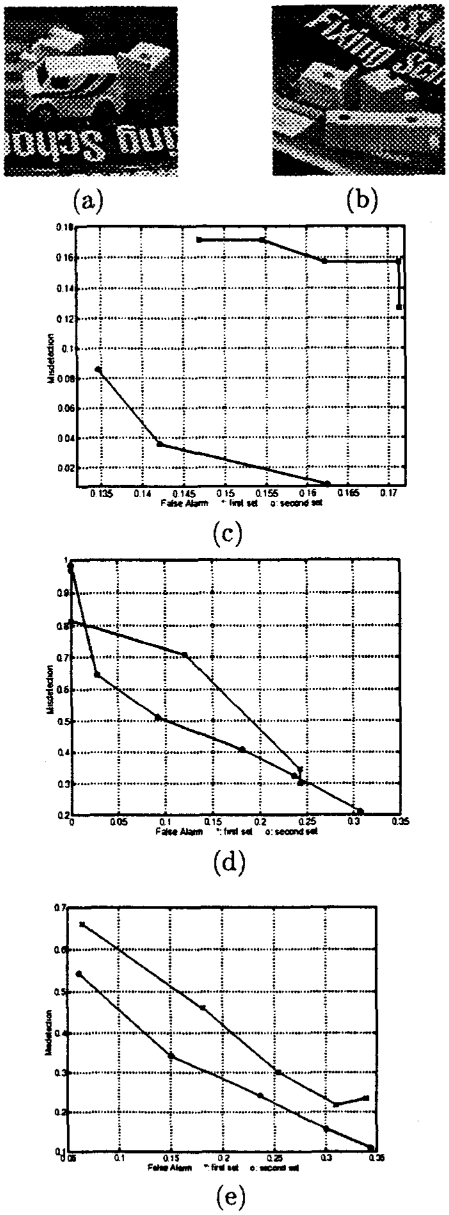


Figure 6: Results for cluttered scenes. (a) Cluttered scenes. (b) MDL segmentations. (c) ABP and ABR hypotheses.

5 Conclusion

In this paper we introduced a new object representation using appearance-based parts and relations. ABPs and ABRs are defined based on the MDL principle and are automatically learned from collections of images without requiring *ad hoc* parameters. They capture



(a) (b)

(c)

(d)

(e)

Figure 7: False alarm versus misdetections for rotating scenes (a) and (b). (c) for ABPs as T_2 varies. (d) ABRs as T_3 varies, $T_2 = 0.05$. (e) for ABPs as T_1 varies, $T_2 = 0.05$ and $T_3 = 0.08$.

not only local shape but also intrinsic reflectance properties, pose in the scene and illumination conditions. Furthermore, ABPs and ABRs are compactly stored using an eigenspace representation parametrized by pose and illumination. Thus, the proposed representation can be used with generic objects and it is robust to occlusion and segmentation variations. Experimental results using images with cluttered backgrounds show that the ABPs and ABRs are useful for object recognition in the presence of occlusion.

6 Acknowledgments

The authors would like to thank Dr. Nayar and Mr. Nene for providing the SLAM software library and their help in using it; and Dr. Kao for some of the test objects.

References

[1] T. O. Binford. Body-centered representation and perception. In *Lecture Notes in Computer Science (994): Object Representation in Computer Vision.* Springer-Verlag, 1995.

[2] O. I. Camps. Towards a robust physics based object recognition system. In *Lecture Notes in Computer Science (994): Object Representation in Computer Vision.* Springer-Verlag, 1995.

[3] M. S. Costa and L. G. Shapiro. Scene analysis using Appearance-Based Models and Relational Indexing. In *International Symposium on Computer Vision*, pages 103–108, Florida, November 1995.

[4] S. J. Dickinson, A. P. Pentland, and A. Rosenfeld. 3-D Shape Recovery using Distributed Aspect Matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):174–198–, February 1992.

[5] M. Hebert, J. Ponce, T. Boult, and A. Gross. Report on the 1995 workshop on 3d object representations in computer vision. In *Lecture Notes in Computer Science (994): Object Representation in Computer Vision.* Springer-Verlag, 1995.

[6] T. Kanungo, B. Dom, W. Niblack, and D. Steele. A Fast Algorithm for MDL-based Multi-band Image Segmentation. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 609–616, Seattle, Washington, June 1994.

[7] D. Kriegman and J. Ponce. Representations for recognizing complex curved 3d objects. In *Lecture Notes in Computer Science (994): Object Representation in Computer Vision.* Springer-Verlag, 1995.

[8] J. Krumm. Eigenfeatures for planar pose measurement of partially occluded objects. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 55–60, San Francisco, California, June 1996.

[9] A. Lenardis and H. Bischof. Dealing with occlusions in the eigenspace approach. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 453–458, San Francisco, California, June 1996.

[10] H. Murase and S. K. Nayar. Visual Learning and Recognition of 3-D Objects from Appearance. *International Journal of Computer Vision*, 14:5–24, January 1995.

[11] S. Nene, S. K. Nayar, and H. Murase. *SLAM: Software Library for Appearance Matching.* Technical Report CUCS-019-94, Department of Computer Scienece, Columbia University, 1994.

[12] E. Oja. *Subspace methods of Pattern Recognition.* Research Studies Press, Hertfordshire, 1983.

[13] M. Zerroug and G. Medioni. The challenge of generic object representation. In *Lecture Notes in Computer Science (994): Object Representation in Computer Vision.* Springer-Verlag, 1995.