Recognising Objects by their Silhouette*

Thorsten Bandlow, Alexa Hauck, Tobias Einsele, Georg Färber
Lab. for Process Control and Real-Time Systems
Technische Universität München, Germany
email: hauck@lpr.e-technik.tu-muenchen.de

ABSTRACT

We present an object recognition system that identifies objects by their silhouette from single views of a greyscale camera. The centroidal profile *describing the object boundary is matched with boundaries from a model base using a* dynamic programming *technique. Objects are modelled by a multi-view representation which can either be learned from a set of images or generated from a geometric model. Experiments with non-rigid and polyhedral objects show the validity of this approach.*

1. INTRODUCTION

Vision-based object recognition is a field of research that has brought forth many systems that are able to recognise objects from single views of greyscale cameras, yet a fast recognition of a large number of free-form, perhaps even generic object classes in complex scenes with occlusion still seems to be a goal impossible to realise. The reason why we want to present another system that is clearly limited to a certain field of application is that we believe that the synthesis of various systems that are each specialised on a different kind of recognition task might proffer a way to attain the larger goal.

Existing recognition systems can be classified by the way objects are represented in the model data base and by the type of features used. Two orthogonal approaches for object representation have evolved (discussed in detail in [9, 17]): *Geometric* representations maintain a 3D model of the entire object with descriptions that vary from the very simple such as triangulated surfaces to the more complex, as superquadrics [19], algebraic surfaces [10] or generalized cylinders [23]. Geometric models permit the construction of large databases, enable part–based descriptions and, therefore, can be used to describe generalised objects and object classes. Geometric models also assist the segmentation process in a top–down manner by predicting views of the object.

In contrast, *appearance–based* representations are "learned" from a set of images of an object, taken from different poses and with varying lighting conditions, implicitly taking into account surface properties like texture or reflectance. Here, too, a wide variety of approaches exist, differing in which image information is used and how data is stored. These approaches range from aspect graphs based on geometrical features and their topological relations [18] to an eigenspace representation on pixel value level [14]. Appearance–based approaches facilitate the matching process, since the data compared is very similar from the start; however, they rely heavily on robust segmentation, which is problematic in the case of cluttered scenes.

A second classification criterion is whether global or local features are used. *Global features*, such as area or compactness, summarise information about the entire visible part of an object. The identification process is thus reduced to comparing the detected image features with those from the model data base and using a measure of difference for classification, which makes such methods very fast. Unfortunately, global features are very sensitive to occlusion and require almost perfect segmentation. Most appearance-based systems use global features, [18] being a well known exception from this rule.

Local features, such as line segments or junctions are often associated with geometric systems. They permit recognition even in cluttered scenes, but require additional stages in the identification process, such as perceptual grouping, establishing of correspondences between image and model features and verification of hypotheses.

Since our object recognition system is part of a project on hand-eye coordination (for details see [7] or [6]), we are dealing with objects that are graspable and thus cannot be heavily occluded. Therefore we decided to use a global feature, the "silhouette", which is the entire region corresponding to the projected object. Published recognition systems using silhouettes differ in the way this global feature is represented as well as in the method of image-model comparison. In [15], the object boundary is represented by the centroidal profile and matched point-wise with reference profiles using a neural net. This approach has the advantage that, in contrast to methods using Fourier descriptors [20] or size functions [22], the information about corresponding boundary points is made explicit and thus can be used to refine the pose estimate.

We propose to modify the approach described in [15] by using a matching algorithm based on the *dynamic programming* technique, a classical pattern recognition algorithm, instead of a neural net. The resulting system is fast and does not require training. Object models can either be learned from images or generated from a geometric model.

The paper is organised as follows: Section 2 describes the segmentation and the extraction of silhouettes. The object model base and the identification process are topic of section 3. Section 4 describes first experiments and results, followed by a short conclusion.

*The work presented in this paper is supported by the *Deutsche Forschungsgemeinschaft* as part of the Special Research Program "Sensorimotor – Analysis of biological systems, modelling and medical-technical application" (SFB 462).

2. FEATURE DETECTION

The process of feature detection can be divided into two phases: *Segmentation*, in which the image part corresponding to the object is determined, and *feature extraction*, in which the segmented image region is processed to extract the information by which the object is to be recognised.

The methods described in this section were developed and implemented using the image analysis system **Halcon**, an extensive domain-independent software library providing low-level and medium-level image processing operators [5].

Segmentation

As the presented object recognition method is to be part of a system for visually guided grasping, some assumptions can be made due to the fact that the objects to be recognised have to be graspable. Typical assumptions in this framework are that objects are compact and that objects to be grasped are placed one at a time in front of the robot-camera system.

Therefore, we can use a simple mechanism to remove the background from each image of the scene. First, we take a reference image G_b of the stationary scene. Then we subtract this reference image from each subsequent image G_o to obtain the tha part G_s that corresponds to the object inserted into the scene. The transformation

$$g_s(x,y) = (g_o(x,y) - g_b(x,y)) + 127, \qquad \forall\, x,y \in D$$

assigns the pixels corresponding to the background a value near 127 (with intensity values in the range of 0-255), where D denotes the domain of an image. The relevant domain R_i is determined by applying a thresholding operator, yielding a binary image. A subsequent dilation with a circular element of radius 5.5 pixels ensures that border regions of low contrast will not be assigned to the background domain.

The segmented region has to be processed further as it may still contain shadows caused by the object. Furthermore, the silhouette extracted from the image should fit the object boundary as closely as possible, as it is to be reused for pose refinement. Therefore, a gradient filtering stage is added to refine the segmentation of the boundary of the object. The surrounding edge of the object is extracted using a modified variation of the recursive Deriche edge detector [11] followed by a non-maximum suppression algorithm to get a skeleton representation. Applying hysteresis thresholding and an edge closing algorithm ensures a coherent contour, which is then filled to yield the silhouette of the object. Note that the filling applies to the entire enclosed area, which means that holes, as in the case of a cup with a handle, disappear.

Feature extraction

Because of the constraints on the complexity of objects and scene due to the projected application of the recognition system, the extracted silhouette can be converted into one global feature instead of many local ones. Possible feature types include Fourier descriptors [20], moments [21], centroidal profile [15], cumulative angular and curvature representations. We chose the centroidal profile as it performs well the presence of noise and distortion [1].

The *centroidal profile* is a sequence of the distances between the centroid of the object region and the points on the boundary. The centroid $\mathbf{m} = (m_x, m_y)^T$ of an region R is determined by the ratio of the first-order moments to the enclosed area:

$$m_x = \frac{\sum_x \sum_y f(x,y)x}{\sum_x \sum_y f(x,y)}, \quad m_y = \frac{\sum_x \sum_y f(x,y)y}{\sum_x \sum_y f(x,y)}$$

$$\text{with} \quad f(x,y) = \begin{cases} 1 & \text{if } x,y \in R \\ 0 & \text{else} \end{cases}$$

Defining N' as the number sample points $\mathbf{s}_{k'} = (x_{k'}, y_{k'})^T$ along the boundary $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_{N'})^T$, the squared Euclidean distances $d_{CP}(k')$ are computed as follows:

$$d_{CP}(k') = |\mathbf{s}_{k'} - \mathbf{m}|^2 = (x_{k'} - m_x)^2 + (y_{k'} - m_y)^2, \quad k' = 1..N'$$

The length of the sequence is determined by the number of samples along the boundary. The boundary can be sampled either at equi-distant or equi-angular intervals. An equi-angular method that samples the contour at equal angular steps is applied in Peli's shape signature system [16]. The angular step size can be derived by applying the sample theorem. Thus, it is assured that the original boundary can be reconstructed from the profile pattern. Unfortunately, two serious problems arise: First, boundaries of highly convex and concave regions are sampled irregularly, secondly more than one sample point may correspond to a single angle. Solving these problems requires additional processing cost and further approximations [4].

An alternative is sampling the contour at equal distances. Using a fixed spacing between the sample points, the total number of samples can vary, depending on the length of the contour. In contrast to methods using a fixed number of sample points, this technique can handle small occlusions, as they cause only local changes. However, the comparison algorithm must process centroidal profiles of different length. The *dynamic programming* technique described in section fulfils this requirement.

To enable pattern matching, the extracted pattern has to be made invariant with respect to translation, rotation and scaling. Since the Euclidean distance is a centered measure, the centroidal profile is automatically normalised with respect to translation. Because objects positioned at different distances from the camera produce pattern profiles of different amplitude, the scale must be normalised. This is achieved by dividing the function $d_{CP}(k')$ by the squared maximum distance, resulting in a range of values $0 \leq d_{CP_0}(k') \leq 1$. In order to extract the angle pose ξ of the segmented region and thus achieve invariance in rotation, a definite starting boundary point $\mathbf{s}_s$ must be specified. To do this, we determine the boundary points closest to the intersection of the boundary with the principal axis of inertia $\mathcal{A}(x,y)$ and select the one with the largest distance from the centroid:

$$|\mathbf{s}_s - \mathbf{m}| = \max \qquad \forall\, \mathbf{s}_s \in \{\mathcal{A}(x,y) \cap \mathbf{S}\}$$

where the angle ξ derives from the moment M_{ij}:

$$\xi = -0.5 arctan(M_{11}, M_{02} - M_{20})$$
$$M_{ij} = \sum_x \sum_y f(x,y)(x - m_x)^i (y - m_y)^j$$

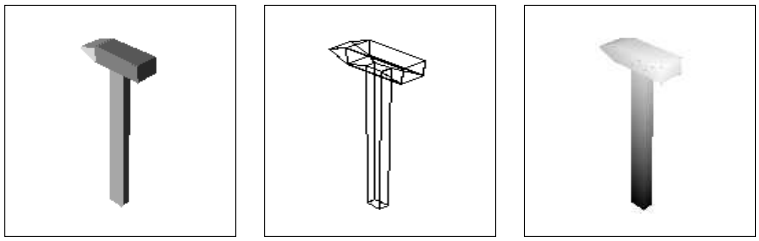

Figure 1: Model of a hammer: a) faces, b) wire frame, c) z-buffer view

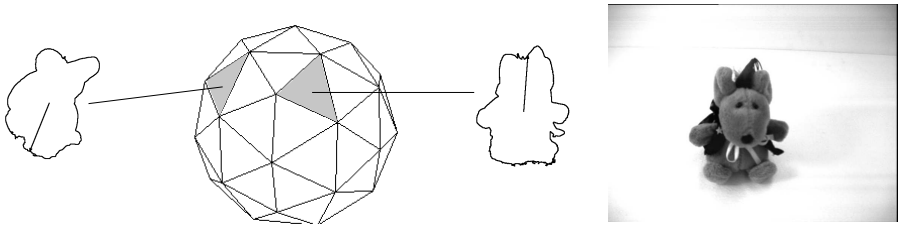

Figure 2: Discretisation of views using a triangulated Gaussian sphere

The principal axis $\mathcal{A}$ is described in the following equation:

$$\mathcal{A}(x,y): \qquad (y-m_y) = (x-m_x)\tan\xi, \qquad x,y \in Z.$$

This yields the normalised centroidal profile function $d_0(k)$ which is then used for classification. Note that this method for determining the starting point does not guarantee a correct result. In the case of unsymmetric, compact objects, a slight change of the viewpoint or the presence of occlusion might cause a significant change in $\mathcal{A}$.

3. OBJECT IDENTIFICATION

The unknown object represented by the extracted normalised centroidal profile $d_0(k)$ is identified by comparing $d_0(k)$ with reference profiles stored in a model database. Therefore, three subjects must be addressed: the creation of a data base of object models, the selection of candidates from this database, and the comparison itself, which yields a measure of the difference between image and model feature.

Model database

As mentioned in section 1, model data bases can be classified according to whether objects are represented by a full 3D model (representation of *shape*) or by a set of views learned from images (representation of *appearance*). The former is impractical in the case of free-form objects but offers easy integration with other image interpretation modules. This is important for the development of complex systems, such as the hand-eye system our project focusses at. With appearance-based approaches, on the other hand, one can learn the model of virtually any object; such models are very sensor- and task-specific, though, and may not be usable for other tasks.

We have chosen a hybrid approach. For polyhedral objects or objects that can be approximated by polyhedrals, a geometric environmental modelling system is used that was originally developed for the use on autonomous mobile robots [8]. In this system, objects are represented by a CAD-like boundary description (see fig. 1a,b). Views of an object for a given camera pose are computed using a *z-buffer* algorithm (see [8] for details). From such a view (Fig. 1c), the silhouette is extracted using the methods described above.

Models of objects that can not be easily approximated by polyhedrals are "learned" by taking a set of images from known viewpoints, extracting the silhouette and storing it in the model database.

For both cases, a *multi-view representation* is used. The viewpoints are determined based on the triangulated Gaussian sphere which guarantees an approximately homogeneous distribution of viewpoints around the object [13]. Fig. 2 illustrates this at the example of a non-polyhedral object. In the silhouettes, a line is drawn from the centroid to the starting point.

Following the method described in [13], the number of stored views can be reduced by determining *characteristic views* and by using the well-known *aspect* approach [3].

Indexing

Another way to reduce the computational complexity and the time requirements of pattern matching is to minimise the number of reference profiles to test by indexing the data base with other features that can be extracted with little effort. Because the segmentation process described above already provides information about the region corresponding to the object, the *compactness* $c = \frac{l^2}{4\pi a}$ with the embedded area a and length l of a contour C can be used as a computationally inexpensive global feature that nevertheless provides a first classification of regions.

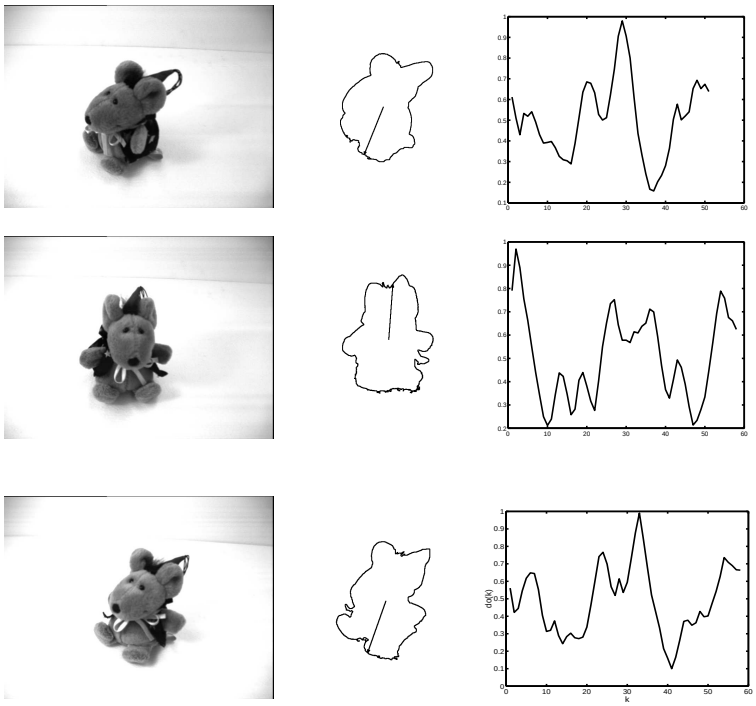

Figure 3: Two exemplary learned views $\langle\ 35cm, 0°, 55°\ \rangle$ (top), $\langle\ 35cm, 45°, 55°\ \rangle$ (middle), and the tested view $\langle\ 45cm, 35°, 50°\ \rangle$ (bottom) of the mouse: a) images, b) silhouettes, c) centroidal profiles

Matching

Dynamic programming (DP) [2] is a classical pattern matching algorithm which establishes and evaluates correspondences between a reference pattern and a test pattern. The actual matching of two profiles is achieved by finding the optimal path through a matrix of grid points that is spanned by the distance measure $d_{DP}(m,n), m \in [1 \ldots M], n \in [1 \ldots N]$ between the normalised centroidal profiles of the reference feature $\mathbf{d_{CP_0,r}}$ and the image feature $\mathbf{d_{CP_0,i}}$, where M is the length of $\mathbf{d_{CP_0,r}}$ and N is the length of $\mathbf{d_{CP_0,i}}$. The complexity of the DP algorithm is of the order $\mathcal{O}(mn)$ in time and space, being the same as in the case of correlation and least squares approaches. However, DP can be massively accelerated, of course not by decreasing its order [12]. In addition, DP performs very well under noisy conditions. For the object recognition system, a global reduced distance matrix is used to further accelerate pattern matching. The global distance between model and image feature is computed by summing the distance values along the optimal path. The unknown object is then identified by sorting the global distance values.

4. EXPERIMENTS

The object recognition system has been implemented in C++. Experiments were run on a P133 PC; here, one matching step takes about $5ms$. In the first experiment, a non-rigid toy mouse was learned by taking images from a fixed elevation angle at a radial distance of about 35 centimeters, while the azimuth angle was varied with a step size of 45 degrees, yielding eight patterns. The silhouette boundaries were sampled at a constant distance of 20 pixels and compared with a test pattern of a similar view, as depicted in fig. 3. The closest adjacent view was detected with a wide margin (see table).

Exp. 1: comparison with view $\langle\ 36cm, 35°, 55°\ \rangle$		
azimuth [deg]	*elevation* [deg]	*distance*
45	55	0.2805
315	55	0.4011
270	55	0.4493
225	55	0.5038
135	55	0.5170
0	55	0.5786
90	55	0.6776
180	55	0.7851

In the second experiment, the robustness of the classification was tested by presenting similar objects in comparable views (fig. 4). The results show the ability of the system to generalise: The two hammers, though of different shape, are classified as being more similar than another object class.

Exp. 2: comparison with hammer1	
object	*distance*
hammer2	0.5237
screw driver	1.1119

5. CONCLUSION

We have presented a system that recognises objects by their silhouette. Silhouettes are represented by the centroidal profile of

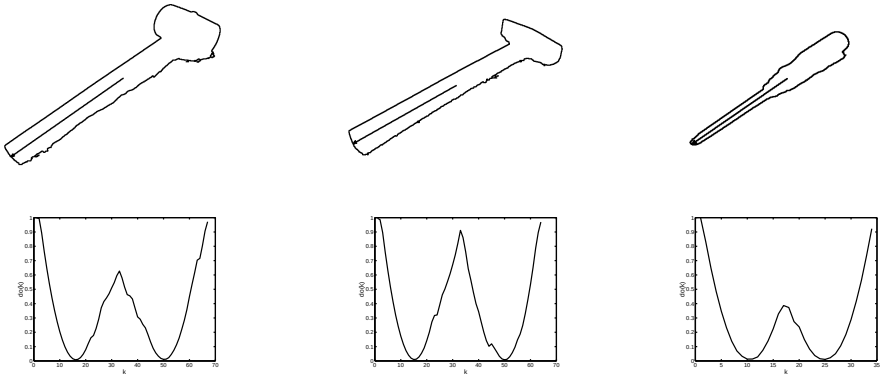

Figure 4: Silhouettes (top) and centroidal profile (bottom) of similar objects from the same viewpoint: a) hammer1 (test pattern), b) hammer2 (reference pattern), c) screw driver (reference pattern)

the region corresponding to the image of the object and matched using a *dynamic programming* technique. The underlying multi-view model database can be learned from images or be generated from a geometric model. Experiments with free-form as well as polyhedral objects have shown promising results concerning performance and speed.

The object recognition module is part of a hand-eye system. On the one hand, this permits the formulation of constraints on objects and scene, thereby reducing the complexity of the problem. On the other hand, the need to integrate it with subsequent image interpretation modules introduces additional requirements. For example, the matching process has to establish point-to-point contour correspondences as input for the localization module.

The proposed object recognition method still must be tested extensively, with the main emphasis lying on the separability of object classes and the sensitivity to occlusion. Additionally, the segmentation process will be developed further, perhaps by combining the implemented algorithm with colour segmentation or contour tracers such as active contours (snakes). To reduce the number of model silhouettes that have to be compared with the extracted one, the system is to be extended by a method that automatically determines the *characteristic views* of an object from a set of views on the triangulated Gaussian sphere.

Concerning the integration of the object recognition module into a hand-eye system, the approximate 3D pose of the recognised object has to be estimated, to serve as a starting hypothesis for a subsequent localisation algorithm.

References

[1] G.N. Bebis and G.M. Papadourakis. Object Recognition using Invariant Object Boundary Representations and Neural Network Modells. *Pattern Recognition*, 25(1):25–44, 1992.

[2] R.E. Bellman and S.E. Dreyfus. *Applied Dynamic Programming*. Princeton University Press, 1962.

[3] I. Chakravarty and H. Freeman. The Use of Characteristic-View Classes in the Recognition of Three-Dimensional Objects. In E. Gelsema and L. Kanal, editors, *Pattern Recognition in Practice*, Amsterdam, 1980. North Holland Publishing Company.

[4] S. Dubois and F. Glanz. An autogressive model approach to two-dimensional shape classification. *IEEE Trans. Pattern Anal. Mach. Intell 8*, pages 55–66, 1986.

[5] Wolfgang Eckstein and Carsten Steger. Architecture for Computer Vision Application Development within the HORUS System. *Journal of Electronic Imaging*, 6(2):244–261, April 1997.

[6] Thomas Fink, Alexa Hauck, and Georg Färber. Towards an Anthropomorphic Robotical Hand-Eye Coordination. In *IMACS Conf. on Comp. Eng. in Systems Appl. (CESA'98)*, April 1998.

[7] A. Hauck and G. Färber. Hybrid Hand-Eye Coordination with a Single Stationary Camera. In *4th Int. Conf. on Control, Automation, Robotics and Computer Vision (ICARCV'96)*, pages 1715–1719, 1996.

[8] A. Hauck and N. O. Stöffler. A Hierarchical World Model with Sensor- and Task-Specific Features. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'96)*, pages 1614–1621, 1996.

[9] M. Hebert, J. Ponce, T. Boult, and A. Gross, editors. *International NFS–ARPA Workshop on Object Representation in Computer Vision, New York City, USA*, volume 994 of *Lecture Notes in Computer Science*. Springer-Verlag, December 1994.

[10] D. J. Kriegman and J. Ponce. Representation for Recognizing Complex Curved 3D Objects. In *Proc. NFS–ARPA Workshop on Object Representation in Computer Vision*, pages 125–138. Springer-Verlag, 1994.

[11] S. Lanser and W. Eckstein. A Modification of Deriche's Approach to Edge Detection. In *11th International Conference on Pattern Recognition (ICPR)*, volume III, pages 633–637. IEEE, 1992.

[12] H. Ney D. Mergel, A. Noll, and A. Paeseler. A Data-Driven Organisation of the Dynamic Programming Beam Search for Continuous Speech Recognition. In *Proc. IEEE Int. Conf. of Acoustics, Speech, and Signal Processing*, pages 833–836. IEEE, 1987.

[13] O. Munkelt. Aspect-Trees: Generation and Interpretation. *Computer Vision and Image Understanding*, 61(3):365–386, May 1995.

[14] H. Murase and S.K. Nayar. Visual Learning and Recognition of 3D Objects from Appearance. *Int. J. Computer Vision*, 14(1):5–24, January 1995.

[15] Kang Park and David J. Cannon. Recognition and Localization of a 3D Polyhydral Object using a Neural Network. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, pages 3613–3618, April 1996.

[16] T. Peli. An algorithm for recognition and localisation of rotated and scaled objects. In *Proc. IEEE 69*, pages 483–485. IEEE, 1981.

[17] J. Ponce, A. Zisserman, and M. Hebert, editors. *International Workshop on Object Representation in Computer Vision II, Cambridge, U.K.*, volume 1144 of *Lecture Notes in Computer Science*. Springer-Verlag, April 1996.

[18] A.R. Pope. *Learning to Recognize Objects in Images: Acquiring and Using Probabilistic Models of Appearance*. PhD thesis, University of British Columbia, Canada, 1995.

[19] E. Rivlin, S. J. Dickinson, and A. Rosenfeld. Recognition by Functional Parts. In *Computer Vision and Pattern Recognition (CVPR)*, pages 267–272. IEEE Computer Society Press, 1994.

[20] K. Arbter W.E. Snyder, H. Burkhardt, and G. Hirzinger. Application of Affine-Invariant Fourier Descriptors to Recognition of 3D Objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):640–647, July 1990.

[21] C. Teh and R. Chin. On image analysis by the method of moments. *IEEE Trans. Inf. Pattern Anal. Mach. Intell. 10*, 1988.

[22] A. Verri, C. Uras, P. Frosini, and M. Ferri. On the use of size functions for shape analysis. *Biol. Cybern.*, 70:99–107, 1993.

[23] M. Zerroug and G. Medioni. The Challenge of Generic Object Recognition. In *Proc. NFS–ARPA Workshop on Object Representation in Computer Vision*, pages 217–232. Springer-Verlag, 1994.