Object Recognition Via Attributed Graph Matching

Irving Hofman and Ray Jarvis
Intelligent Robotics Research Centre
Department of Electrical & Computer Systems Engineering
Monash University
Wellington Road, Clayton, Victoria, 3168, Australia

Abstract

A complete machine vision system implemented at Monash University is described, covering all the stages beginning with data acquisition, segmentation, modelling, through to matching. The process provides a near complete scene description, including the identity, location and pose of objects in it. This information is clearly suitable for the support of subsequent intelligent robotic manipulation in that domain. In this paper particular emphasis is made on the segmentation, modelling and matching stages. Attributed graphs are used to describe both objects in the scene and the model database. A sub-graph isomorphic approach is used to generate matching hypotheses. The hypotheses are confirmed or rejected by examining the residual of an affine transformation between the objects in the scene and proposed models. The system is capable of handling multiple objects with occlusion. Preliminary results are presented.

1 Introduction

Machine vision involves "the construction of explicit, meaningful descriptions of physical objects from images" [Bal82]. In order to achieve this, it is necessary to determine "what is present in the world, and where it is" [Mar82]. Consequently, machine vision entails a great deal more than simply acquiring some images into a computer.

This paper will describe a machine vision system which has been implemented at Monash University. Section 2 will present a framework for a complete vision system, describing all the necessary stages. Section 3 will describe the implementation of this framework, including some preliminary results. Finally, section 4 will make some concluding remarks.

2 Machine Vision Framework

The implementation of a complete machine vision system involves a sequence of processes, starting with a scene to be analysed and ending with a complete description of the scene being viewed. The processes are data acquisition, segmentation, modelling, and matching. These processes will be briefly described in the following subsections.

2.1 Data Acquisition

Data acquisition is the process of using sensors to gather information about the scene being viewed. Cameras and range finders are commonly used to acquire a sampled three-dimensional point representation of the surfaces of objects, often called a range image [Lev73, Yak78, Shi71, Hal82, Ale87].

If more than one view of the scene is acquired, then the data from the various views must be brought into a common frame of reference. This process is known as registration [Bes92].

2.2 Segmentation

Segmentation involves extracting geometric primitives relevant to higher-level cognitive processes. The range data is partitioned into clusters, each with a common geometric property. There are two general approaches – edge based and region based.

Edged based approaches detect significant surface discontinuities, and based on knowledge of where the edges are, form surface patches [Lan84, Mit83, Fan87]. Region based approaches estimate some property at each range pixel and cluster range pixels with homogeneous properties to form surface patches [Har83, Bes88, Hof87]. There are also numerous hybrid approaches.

2.3 Modelling

Modelling involves representing the segmented data in a higher-level form than pixels. Common representations used are wire-frame [Low87], constructive solid geometry (CSG) [Fol90], geometric icons [Bie85], aspect graph [Fan87, Che89, Fly91, Tan92], extended Gaussian image [Hor84], generalised cylinders [Agi76], super-quadric [Pen86], and octree [Che88].

2.4 Matching

Having obtained a higher-level description of the scene, the final step is matching this description to a database of models. Matching involves two steps – recognition and localisation. Recognition is the process of identifying each object in the scene. Localisation involves determining precisely where the objects are located in space, both position and orientation (pose).

There are two general approaches to matching. The first is recognition via localisation. Invariant geometric features are extracted from the scene and matched against the features in a model database. Matches are checked for local consistency. The global solution is found from the set of local consistency matches. The interpretation tree search [Gri86] is a popular mechanism for implementation of this paradigm.

The second approach is recognition followed by localisation. A graph representation of the scene is matched against graphs of the model database using techniques such as sub-graph isomorphism or maximal clique detection [Fan89]. Then, global shape descriptors or object specific features are used to compute the pose. A combination of both these approaches is used in this research and explained in section 3.4.

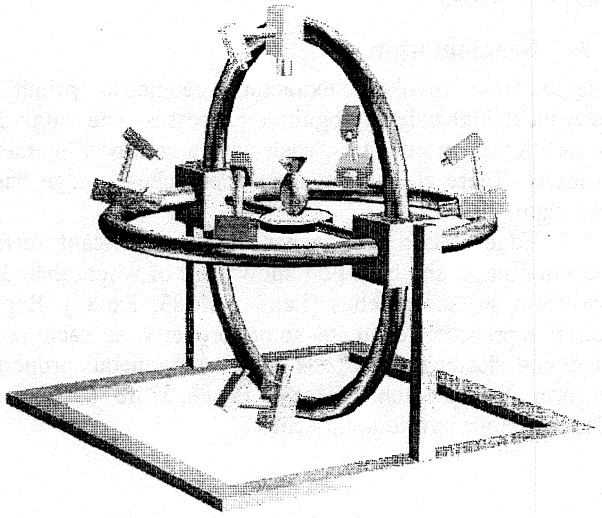


Figure 1: Experimental Rig

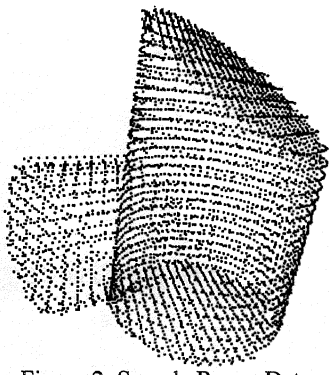


Figure 2: Sample Range Data

3 Machine Vision Implementation

3.1 Data Acquisition

This research uses six active striped light range finders to implement a multiple view system. The range finders are mounted onto a custom built chassis in six different fixed viewing positions – four side views, a top view, and a bottom view. Objects in the scene are placed onto a more or less transparent metallic mesh located at the centre of the rig. A diagram of the rig is shown in Figure 1. Colour data is also obtained by fusing CCD intensity images with the range data.

Data is returned in the form of (x, y, z) triplets, rather than the more conventional regularly sampled parametric range map format $z = f(x, y)$. Range point adjacency information is also returned by the system. Inter-point spacing varies from 0.5 to 1.5 centimetres, depending on the viewpoint and underlying surface topology. Objects used have 300 to 2000 sampled points per view, depending on the size of the object.

Views are merged together using a unique registration technique which utilises a custom built registration frame. Registration is only performed once since the sensors and scene workspace are fixed. Registration accuracy is approximately 3mm, which is less than the sampling density of the range finders. Sample range data of a pipe object taken from the six views and then registered is shown in Figure 2.

3.2 Segmentation

The outward pointing surface normal vectors for each range point are computed by fitting a plane to a 3×3 and 5×5 window neighbourhood. If a hard edge is present near a range point – which would corrupt the normal vector – the plane equation will vary significantly between the two window sizes. The exact location of the edge cannot be determined due to the sparse and irregular density of the range points. Nevertheless, this approach ensures that data points are only assigned normal vectors if it is known that the vector is correct. The above process and all of the

segmentation stages are performed on individual views.

The eigenvectors from each plane equation form a Darboux frame for each point. A bi-quadric polynomial is then fitted to a 5×5 window around each range point, ensuring that hard edge boundaries are excluded from the calculation. The two principal curvatures at each point, κ_1 and κ_2, are calculated directly from the bi-quadric equation. These values are then converted into a shape index and curvedness measure, using the formulas developed by Dorai [Dor97]. Shape index represents the shape of the underlying surface, which ranges from 0 to 1 (equation 1). Planar surfaces cannot be represented on the shape index scale, so are arbitrarily assigned a value of 2. Curvedness represents how highly or gently curved that surface is (equation 2). This allows the underlying surface types at range points, p, to be uniquely identified by two real numbers.

$$SI(p) = \frac{1}{2} - \frac{1}{\pi}\tan^{-1}\frac{\kappa_1(p)+\kappa_2(p)}{\kappa_1(p)-\kappa_2(p)} \quad (1)$$

$$C(p) = \sqrt{\frac{\kappa_1^2(p)+\kappa_2^2(p)}{2}} \quad (2)$$

An iterative region growing approach is then used which merges points together that have similar shape index [Dor97]. Each point starts out as its own patch. During each iteration, it merges the two adjacent patches that would result in the least merged shape index spread. This process continues until the shape index spread of a region would become larger than a predefined threshold. This region growing approach automatically minimises the number of regions or surface patches.

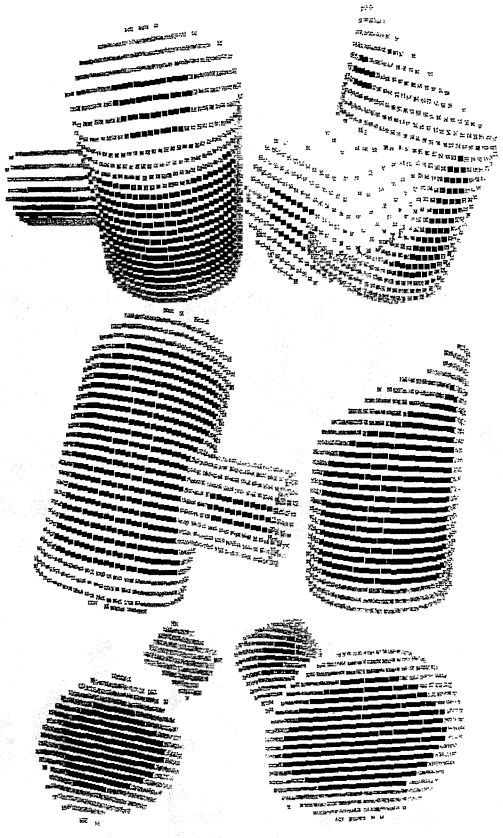


Figure 3: Segmented Pipe Object

Segmentation of the sample pipe object is shown in Figure 3, with extracted surface patches in black. Each view is shown separately for clarity. Notice how regions near hard edges are not included in the patches, due to the inability to reliably detect their exact location. Regions near the extremes of objects are not included because there are insufficient points to form a 5×5 window neighbourhood. Use of higher density range finders will minimise the amount of unsegmented surface parts, and will permit the reliable detection of edges.

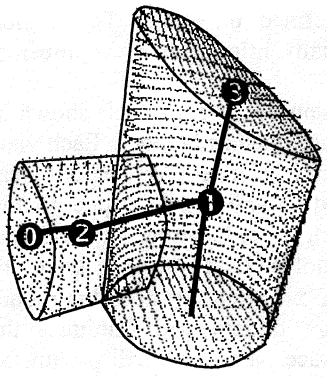
3.3 Modelling

Attributed graphs are used to model the objects. These are standard graphs in the mathematical sense consisting of vertices and edges, but with the addition of certain attributes assigned to the vertices and edges. Each view is represented by it's own attributed graph. The surface patches extracted during the segmentation stage form the vertices (nodes) of the graph. Each node contains geometric and other information about the corresponding surface patch. Nodal properties include the underlying point set, shape index distribution, curvedness distribution, colour distribution, and quadric polynomial. These are all unary properties.

The quadric equation is determined by a least squares fitting approach, and uses all the 10 quadric parameters. This permits a quadric to be fitted regardless of the point distribution in space. Note that quadrics are generally unbounded (except the sphere and ellipsoid) and extend to infinity. More intricate surface topologies can be modelled using alternative representations, such as higher order polynomial surfaces. Properties such as surface area and centroid, which are very useful and commonly used in other works, have not been used here because they are not occlusion tolerant.

The edges (links/arcs) of the graph represent the attributes and relationship between two adjacent surface patches. An edge exists between nodes if they are connected by at least one boundary. Edge information is recovered by using a distance threshold from the point sets of all node pair combinations. If the distance is small, then the patches are considered adjacent. If two patches partially overlap, then they are merged into a single node. The properties of the merged node are derived from the properties of the individual nodes. It is at this point, that the graphs from the six views become merged into a single graph. The only edge property used is the angle between planar patches. This is a binary property.

If denser and more uniform range data were available, it would be possible to extract other edge attributes such as the type of adjacency (e.g. jump, limb, convex crease, concave crease). It is also theoretically possible to extract edge information such as the equation of the space curve formed by the by intersection of adjacent



Node 2 properties
Number of points: 93 points
Quadric equation:
$0.485x^2+0.137y^2+0.702z^2+0.502xy+0.028xz-0.007yz-0.017x-0.008y-0.087z+0.002=0$
Shape index spread: 0.686 to 0.809
Median shape index: 0.750
Curvedness spread: 23.894 to 28.651
Median curvedness: 26.333
Colour: red=0.3, blue=0.9, green=0.7

Figure 4: Pipe Model

quadric equations. However, there are some difficult mathematical issues which the authors have not been able to resolve.

Each object in the model database is represented by its own attributed graph. Objects in the scene to be analysed are also represented by attributed graphs. Scenes containing multiple objects will have separate graphs for each object. If two objects are touching and the surface properties of the touching surfaces are identical, the two objects will be represented by a single graph.

Figure 4 shows the graph used to represent the pipe object overlaid over the range data. A node represents each semantically distinct surface region, which have been outlined for clarity. The edges represent connectivity of surface patches. The properties of node 1 are also shown.

3.4 Matching

The goal is to recognise objects in scenes with occlusions. Each scene may contain multiple objects with self and mutual occlusion. Determining a match consists of finding the largest sub-graph in a model object for which every node maps onto a node of the scene object graph, according to the geometric operations which transform the model onto the scene. The transformation is initially unknown. This process is a variant of sub-graph isomorphism.

Rather than performing an exhaustive search on all possible sets of pairs and transformations, a two-stage process is used. The first stage finds the likely model object candidates for each scene object in the scene. It implements a fast search involving the computation of coarse differences in properties of the nodes of the graphs. The output is a short list of candidates. The second stage confirms or rejects each matching hypothesis generated in the first stage by examining the residual of the affine transformation that maps the model object onto the scene object. If the residual is lower than a specified threshold, then a match has been found.

During the first stage, vertex compatibility matrices (VCMs) are constructed for each model-scene combination. Matrix rows represent model object nodes, while columns represent scene object nodes. A '1' entry denotes a potential match between that model and object node. A '0' entry denotes a definite mismatch. A sample VCM is shown in Figure 5.

Each column of the VCM is examined. If all the entries are '0', then the object cannot possibly match the model because that object node doesn't match any model nodes. If there are multiple entries of '1' (like object node zero in the example), then there is a conflict that needs to be resolved by examining the edge properties and the node properties of adjacent nodes. The degree of a node is another useful property. For a potential match, the degree of a scene node must be less than or equal to the degree of a model node. This is because a scene graph can't be larger than a model graph.

At each stage, a VCM is assigned a mismatch value that denotes how good or bad the match is. The lower the value, the better the match, with a value of zero representing a perfect match. This information can be used to order the matching hypothesis and verification process.

Not all conflicts can be directly resolved, so it is necessary to resort to affine transformations to resolve these types of conflicts. A matching permutation matrix (MPM) is constructed which contains a list of hypothesised matches. Each row of the MPM is a potential model and scene object match. The MPM is partially self-culling because any rows which contain two duplicate numbers cannot possibly be a match. Powell's direction set method is applied to every valid hypothesised match to determine the affine transformation which best aligns the model and object pair. By examining the residual of substituting the point sets from

MODEL B \ *OBJECT A*	0	1	2	3
0	1	0	0	0
1	0	1	0	0
2	1	0	0	1
3	0	0	1	0
4	0	0	0	0

Figure 5: Sample VCM

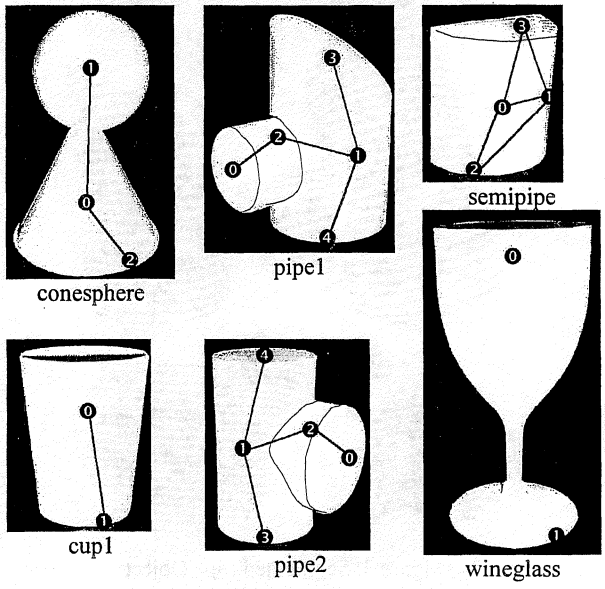


Figure 6: Model Database with Superimposed Graphs

a scene object into the proposed corresponding quadric equations of a model object, the presence of a match can be determined without conflict. Of course, it is always possible that an object in the scene doesn't match any of the objects in the database.

Even if a match is found without using the second affine transformation stage, an affine transformation is still required to recover the position and orientation of the scene object (localisation). Object identification is only part of the matching process.

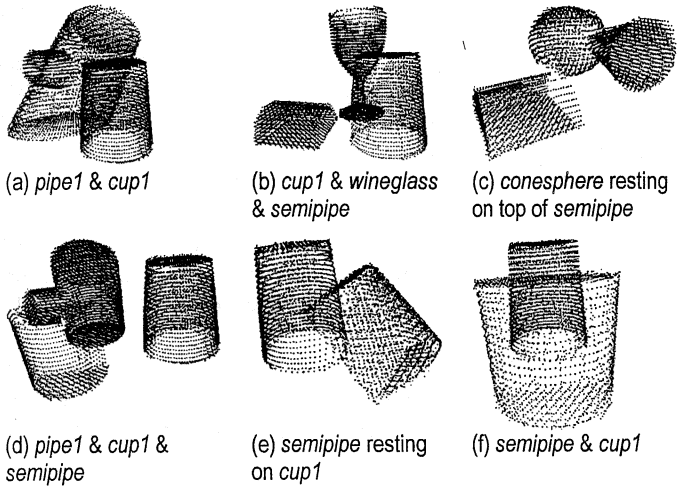
The database of objects used to obtain preliminary matching results is shown in Figure 6. The basic graph structure is superimposed over the models. The models were carefully chosen to test various aspects of the matching system. For example, *pipe1* and *pipe* are very similar; the radius of node '0' in *semipipe* is the same as node '1' in *pipe1*; the model of *wineglass* is incomplete because part of the model is too small compared to the range data density.

Scenes containing one or more of the objects in the database were presented to the system. Objects were placed in different positions and orientations to those of the corresponding model. In some scenes objects were abutting each other. Occlusion is present in scenes with multiple objects. Figure 7 shows range data of the scenes tested. Correct object identification and localisation occurred in all the test cases. The use of multiple views ensures that most object parts are visible in at least one view.

Figure 8 shows an example of the matching process, using the *pipe1* object from the scene in Figure 7a. After step 1, the only model which might match the scene is *pipe1*. All the other models contain a column of zeros. The various permutations in step 2 arise due to the planar nodes. There are insufficient attributes in the graph to determine the correct matching. In step 3, affine transformations are used to determine which valid permutation is correct.

4 Conclusions

A complete machine vision system developed at Monash University which uses range data from six fixed views was described. The stages of data acquisition, segmentation,



(a) *pipe1* & *cup1*

(b) *cup1* & *wineglass* & *semipipe*

(c) *conesphere* resting on top of *semipipe*

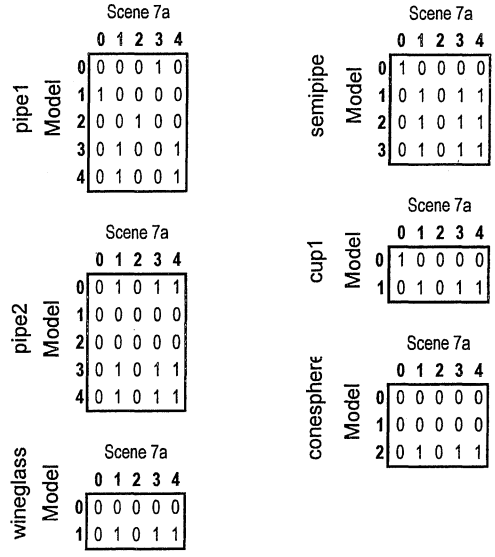
(d) *pipe1* & *cup1* & *semipipe*

(e) *semipipe* resting on *cup1*

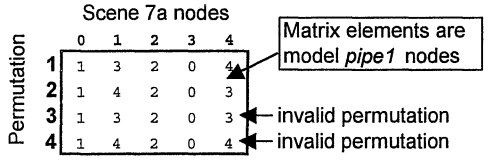
(f) *semipipe* & *cup1*

Figure 7: Test Scenes

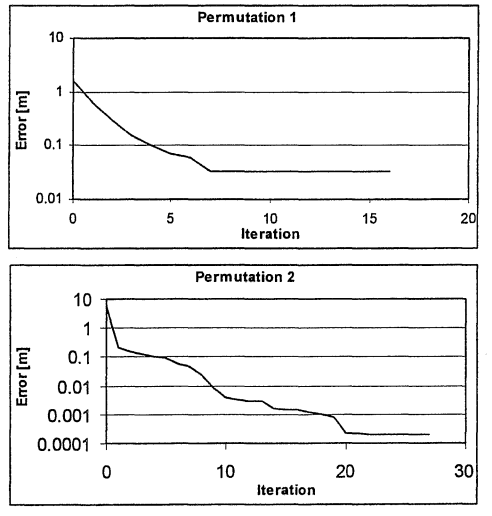
Step 1: Compute vertex compatibility matrices (VCMs). The graph topology and attributes are used to determine if a scene node potentially matches a model node. Note: only VCMs for *pipe1* object are shown.



Step 2: Compute matching permutation matrix (MPM). This is extracted from the VCM and lists all the matching permutations.



Step 3: Compute affine transformations for valid permutations. This is done using Powell's direction set method. The permutation with the lowest error is the correct one, if it is less than a predefined threshold.



Permutation 2 results in an error of 0.2mm. The affine parameters that align the scene with the model are:

$X_{rotation}$ = 2.788 rad, $Y_{rotation}$ = -0.463 rad, $Z_{rotation}$ = -3.123 rad

$X_{translation}$ = 0.014 m, $Y_{translation}$ = -0.014 m, $Z_{translation}$ = 0.109 m

Figure 8: Example of Matching Process

modelling, and matching were described. Favourable results have been obtained on scenes containing multiple objects and occlusion. The objects used are only moderately complex due to limitations of the range finder data. Nevertheless, the techniques used are scalable to more complicated objects. The use of attributed graphs to model objects, rather than other schemes such as CSG, is highly suitable for machine vision systems because it provides a unique qualitative and quantitative description capable of recognising arbitrarily curved surfaces in the presence of partial occlusion.

The methodology adopted in this research is deliberately the antithesis of the 'active' vision approach, which seeks to combine intelligent selective exploration with data acquisition. Active vision uses a repeated cycle of analysis followed by action to arrive at conclusions. Unless significant care is taken, a single error in the chain of decisions of the 'active' approach can lead to an irrecoverable breakdown of correct analysis, thus rendering this procedure fragile outside strictly defined domains. Our approach avoids fragility by delaying the binding of domain specific information and providing semantic-free structural descriptions before that binding is activated.

For future work, it would be easy to envisage the implementation of a robotic manipulation process to handle all the objects in the scene as described by this vision system.

References

[Agi76] Agin, G. J. and Binford, T. O., "Computer description of curved objects", *IEEE Transactions on Computers*, Vol. 25, No. 4, 1976, pp. 439-449

[Ale87] Alexander, B. F. and Ng, K.C., "3D shape measurement by active triangulation using an array of coded light stripes", SPIE, Vol. 850, 1987, pp. 199-209

[Bal82] Ballard, D. H. and Brown, C. M., *Computer Vision*, Prentice-Hall, New Jersey, 1982

[Bes88] Besl, P. J. and Jain, R. C., "Segmentation through variable order surface fitting", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 2, 1988, pp. 167-192

[Bes92] Besl, P. J. and McKay, N. D., "A method of registration of 3-D shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, 1992, pp. 239-256

[Bie85] Bierderman, I., "Human image understanding: recent research and a theory", *Computer Vision, Graphics and Image Processing*, Vol. 32, 1985, pp. 29-73

[Che88] Chen, H. H. and Huang, T. S., "A survey of construction and manipulation of octrees", *Computer Vision Graphics Image Processing*, Vol. 43, 1988, pp. 409-431

[Che89] Chen, C. H. and Kak, A. C., "A robot vision system for recognizing 3-D objects in low-order polynomial time", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 6, 1989, pp. 1535-1563

[Fan87] Fan, T. J., Medioni, G. and Nevatia, R., "Segmented descriptions of 3D surfaces", *IEEE Journal of Robotics and Automation*, Vol. 3, No. 6, 1987, pp. 527-538

[Fan89] Fan, T. J., Mediono, G. and Nevatia, R., "Recognising 3-D objects using surface descriptions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 11, 1989, pp. 1140-1157

[Fly91] Flynn, P. J. and Jain, A. K., "Cad-based computer vision: From cad models to relational graphs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 10, 1991, pp. 1066-1075

[Fol90] Foley, J. D., Van Dam, A., Feiner, S. K. and Hughes, J. F., *Computer Graphics: Principles and Practice*, Second Edition, Addison-Wesley, New York, 1990

[Gri86] Grimson, W. E. L., "The combinatorics of local constratins in model-based recognition and localization from sparse data", *Journal of the ACM*, Vol. 33, No. 4, 1986, pp. 658-686

[Hal82] Hall E. L. et al., "Measuring curved surfaces for robot vision", *Computer*, Vol. 15, No. 12, 1982, pp. 42-54

[Har83] Haralick, R. M., Watson, L. T. and Laffey, T. J., "The topographic primal sketch", *International Journal of Robotics Research*, Vol. 2, No. 1, 1983, pp. 50-72

[Hof87] Hoffman, R. and Jain, A. K., "Segmentation and classification of range images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 5, 1987, pp. 608-620

[Hor84] Horn, B. K. P., "Extended Gaussian images", *Proceedings IEEE*, Vol. 72, No. 12, 1984, pp. 1671-1686

[Lan84] Langridge, D. J., "Detection of discontinuities in the first derivatives of surfaces", *Computer Vision Graphics and Image Processing*, Vol. 27, 1984, pp. 291-308

[Lev73] Levine, M. D., O'Hundley, D. A., an Yagi, D. A., "Computer determination of depth maps", *Computer Graphics and Image Processing*, Vol. 2, 1973, pp. 134-150

[Low87] Lowe, D. G., "Three-dimensional object recognition from single two-dimensional images", *Artificial Intelligence*, Vol. 31, 1987, pp. 355-395

[Mar82] Marr, D., *Vision*, Freeman, San Francisco, 1982

[Mit83] Mitchie, A. and Aggarwal, J. K., "Detection of edges using range information", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 5, No. 2, 1983, pp. 174-178

[Pem86] Pentland, A. P., "Perceptual organization and representation of natural form", *Artificial Intelligence*, Vol. 28, No. 2, 1986, pp. 293-331

[Shi71] Shirai, Y. and Suwa, M., "Recognition of polyhedrons with a range finder", Proceedings 2nd *International Joint Conference on Artificial Intelligence*, 1971, pp. 80-87

[Tan92] Tang, Y. C. and Lee, C. S. G., "A geometric feature relation graph formulation for consistent sensor fusion", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 22, No. 1, 1992, pp. 115-129

[Yak78] Yakimovsky, Y. and Cunningham, R., "A system for extracting three-dimensional measurements from a stereo pair of TV cameras", *Computer Graphics and Image Processing*, Vol. 7, 1978, pp. 195-210