Journal of Computing and Information Technology - CIT 2, 1994, 4, 309–318

3D-Modeling for the Analysis of Range Data

J. Duckeck, H. Kirchner

Bavarian Research Center for Knowledge Based Systems, Erlangen, Germany

In this paper a new method of modeling 3D–objects for the interpretation of range images is described.

In range data it is possible to detect surface patches, edges and 3D–vertices by local and therefore fast operations. Recognition of an object is done by associating these structures with one element of a set of possible objects. For that it is necessary to use an adequate description of objects which supports fast matching. In this paper a model is presented, which represents an object by surfaces, edges, vertices and neighborhood relationships. The representation of the objects is restricted on those surfaces, which are easy to detect by the used range sensor.

A further requirement is a fast and easy interactive construction of object models which assists human demands and thus reduces the costs of providing and maintenance of the object database.

Keywords: range data, range image, depth data, surface patch, edge, vertex, neighborhood.

1. Introduction

Today object recognition and pose estimation is an urgent problem in automation. In industrial environments the shape of objects is known a priori. In most cases they are rigid and often they have plane surfaces, bolts or at least drill-holes for the assemblage. But the recognition has to be fast and efficient. Also the objects may be rather similar so it can be impossible to make a distinction based on a single view.

Recognition and pose estimation requires the matching of structures in an image with corresponding structures in the model. As the number of comparisons necessary for this process is very large, it is urgent to optimize the efficiency of this operation. So the goal of this work was to optimize the number of parameters that have to be compared, already in the modeling phase.

Several approaches use geometric primitives like vertices, edges and surfaces for object recognition [8, 10, 15, 16]. Additional informations about objects are topological characteristics [20] which can also be used. To increase the efficiency of the feature extraction process, a restriction on small areas of the range image and local structures of the model is made [6].

In this work we introduce a method of modeling three dimensional objects suitable for fast recognition of objects in industrial environments. The model consists of plane and cylindrical surfaces which are very common. The restriction on geometric structures reduces the number of parameters relevant for the matching. But the modular structure of the model allows the addition of every surface description that may be useful for future applications.

2. Acquisition of Depth

There are several approaches for the acquisition of depth (e.g. the distance between an object point and the focus of the camera) [2, 21, 12]. The photogrammetric computation is based on triangulation. Passive systems use two or more cameras [12]. Active systems use laser scan [2] or patterned [3], coded [21, 18] or structured light combined with a single camera (Figure 1). Recent research deals with active vision by focusing, zooming or moving the camera. The resulting image sequences are used to evaluate depth.

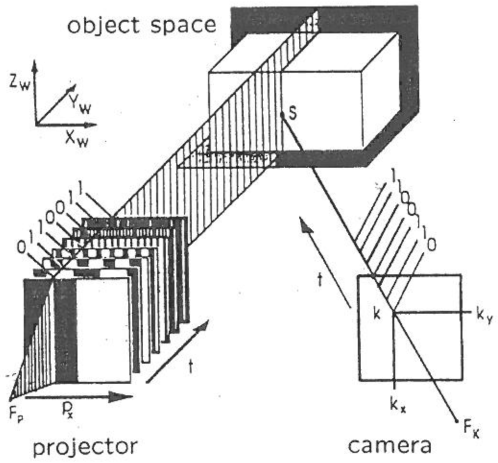


Fig. 1. The Coded Light Approach (CLA)

The result of those methods are very similar. It is always a two dimensional picture containing depth values instead of grey values (Figure 2). This kind of of data is also called $2\frac{1}{2}$ dimensional data.

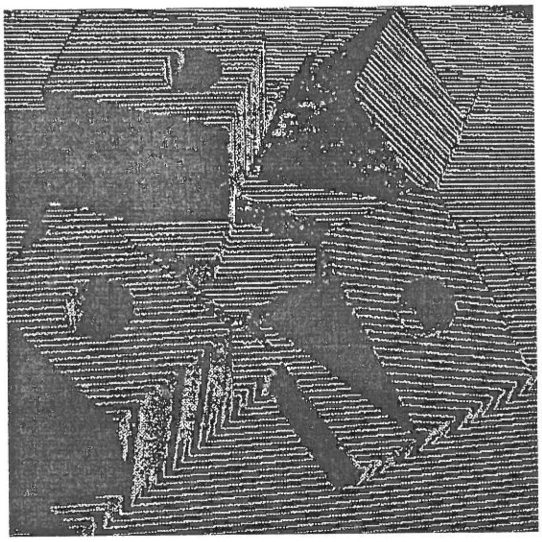


Fig. 2. A typical range image. Grey values are discrete depth values. Black pixels contain no information about the object.

The process of fitting three dimensional models with range data is very time consuming. So a first step is to detect areas of interest by a very easy and therefore fast algorithm. The next step is to detect single geometric features in the neighborhood of those areas of interest (Figure 3).

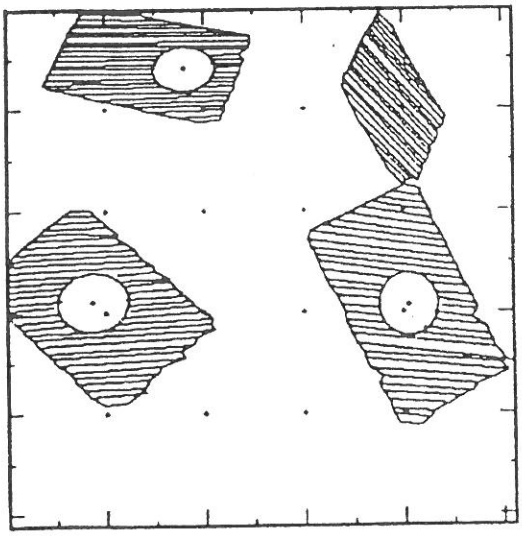


Fig. 3. Four detected patches of plain surfaces

Tools for detecting plain or cylindrical surfaces, edges and vertices exist [9]. In contrast to two dimensional images the range data yields three dimensional coordinates of vertices, also orientation of edges and surfaces and area of visual surface patches [7]. Additionally, it is possible to detect whether a found edge is a ridge or a step. A ridge links two surface patches thus defining a neighborhood. If the edge marks a step, the step is also found in depth data and the neighborhood in the image is just caused by the viewing angle.

3. Geometric Model

The geometric model, the abstract world, is a description of a set of objects.

The structure of a model depends on its intention. In our case it is an abstract description of those parts of the real world objects, which can easily be detected in the range data. The grey–filled boxes in Figure 4 represent the knowledge we got by analyzing the range image. We are able to detect this primitives. Now object recognition is the determination to which object the found structures belong.

The modeled world is a set of objects, which are made of surfaces, edges, vertices and neighborhood relations between surfaces. The basic

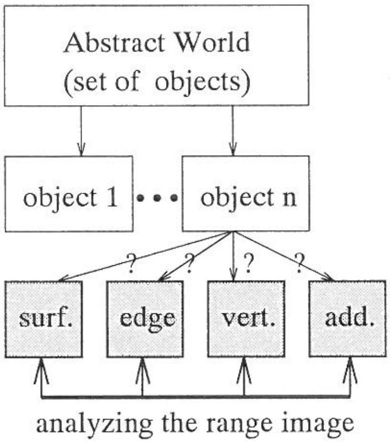


Fig. 4. Motivating the model by detectable primitives

structure of the model is a tree (Figure 5). Figure 5 is very similar to Figure 4 because

- it is necessary to model anything we can detect,
- it is not useful to model anything else, because of runtime, memory, time for modeling and so forth.

The nodes of the tree are geometric elements like edges and surfaces. The edges of the tree represent the relation *"belongs to"*. Vertices belong to edges, edges belong to surfaces and surfaces belong to objects. When an edge belongs to two different surfaces, this fact implicates neighborhood of those surfaces. The tree itself is thus a representation of the structure of an object. The leaves are vertices and consist of three dimensional coordinates. Each object has its own object coordinate system (OCS) and all vertices are defined in this coordinate system (Figure 8).

4. Edges

At the moment four types of edges are used. They are very common and there exist methods to detect them in range data.

1. **Straight Edges** are easy to detect and easy to describe. They are the basic constituents of polyhedrical models. A **Straight Edge** is defined by start and end vertex. Derived parameters are length and orientation.
2. **Virtual Edges** were introduced because of syntactical reasons. They can be used to define edges which are too complex to define. So the meaning is:

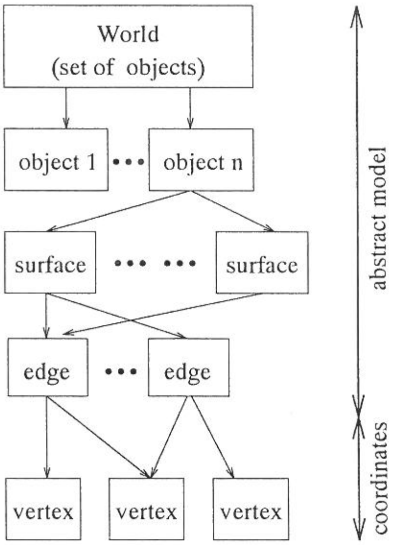


Fig. 5. The basic structure of the modeling conception

- "There is an edge but I do not know how to describe it."
- "If you find something in the image, do not use it."

The second use for them is to add internal contours like drill-holes. Drill-holes are modeled because they are very characteristic and thus useful for the object recognition.

3. **Round Edges** are straight surface boundaries plus information about the neighborhood. They are a way to describe bends or flexures. Such structures are common in industrial environments (Figure 7).

Round Edges are links between two surface patches, where we are not able to describe the interspace. But if we do no want to lose

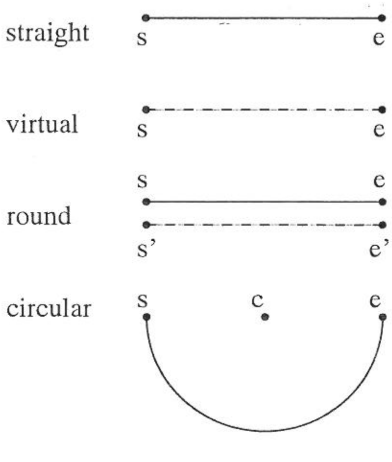
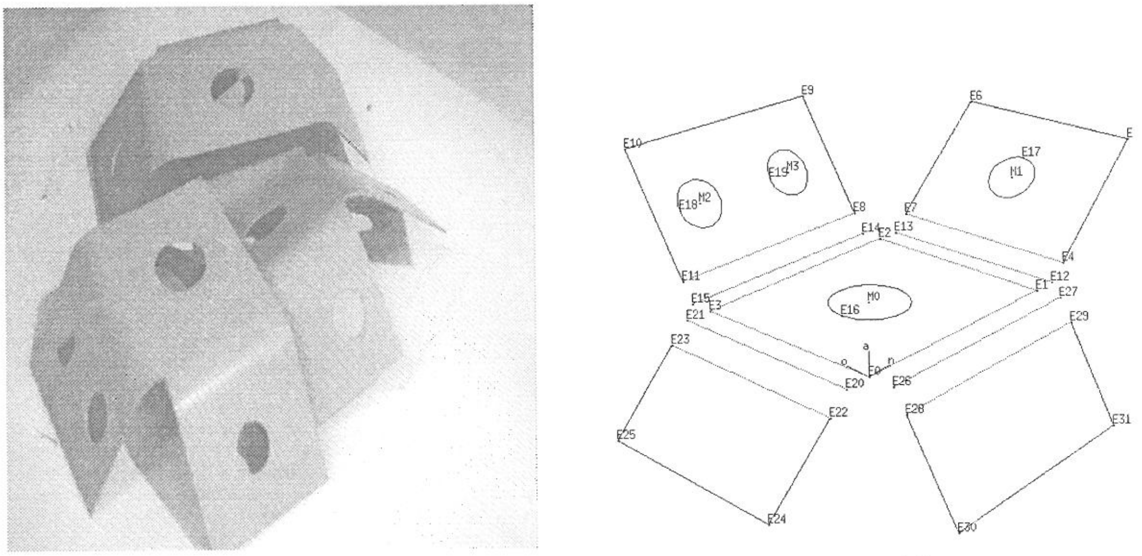


Fig. 6. Four types of edges



Typical industrial objects

model

Fig. 7. Example for the edge type **Round Edge**

the information of those two surfaces being neighbors, we have to define an edge that consists of three parts:

1. the boundary of the surface,
2. the boundary of the neighbor surface
3. and information about how they are linked.

There are only straight **Round Edges**, so the linked plane surfaces normally have a line of intersection.

4. **Circular Edges** are needed to describe cylindrical surfaces or drill-holes in surfaces.

As the used tools are able to compute center, radius and orientation of circles it is useful not to approximate circles by polygons. **Circular Edges** are represented by start and end vertex plus center. Circle parameters like radius and orientation are derived from this data.

An edge has two independent characteristics:

1. geometric structure: straight, circular, spline, ...
2. visibility: not visible, visible, detectable.

As some combinations between geometric structure and visibility make no sense, it is not useful to model all of them. In case of the straight edges the meaning and also the structure of an edge is different for each type of visibility (table 1). Not visible circular edges make no difference to the straight **Virtual Edge**, because the geometry of invisible edges does not matter.

Round edges with circular geometry may be a useful addition, but as there was no example object with this characteristic, this edge type is not defined yet. It is very easy to add this edge type to the model, if necessary. So this is an example for the advantage of the modular modeling concept.

A special problem for the object recognition is the visibility of surface boundaries. The transi-

edge	geometry	visibility
Straight Edge	straight	visible or detectable
Virtual Edge	straight	not visible
Round Edge	straight	detectable
Circular Edge	circular	visible or detectable

Table 1. Geometry and visibility of the four types of edges

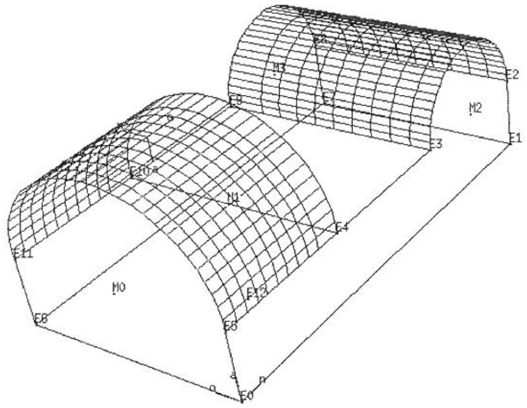


Fig. 8. An object with **Plane Surfaces** and **Cylindrical Surfaces**. The entire surface of this object is modeled. The object coordinate system (OCS) is in the front corner.

tion from a surface patch to the rounded intersection may be very smooth, see Figure 7. A similar edge is E4-E5 in Figure 8 where a cylinder patch turns into a plain patch. Although those edges are not visible in grey images, they are detectable in range data as they mark the border between surfaces of different characteristics. A problem is that they may be very inaccurate. Their position may depend on the point of view.

5. Surfaces

The described tools are able to detect plain and cylindrical surface patches, so these are the two existing kinds of surfaces.

The addition of more surface models, if corresponding detection tools are developed, is intended.

Both surface types are bounded by edges, so they are described by a closed loop of edges. This means that the end of one edge is the start of the next one. The end of the last edge is the start of the first.

Plane Surfaces are finite parts of a plane. An infinite large geometric plane is cut in two parts by a closed loop of edges (Figure 9). The parts with the finite area is the inner side. All bounding edges and their vertices are in this plane. The orientation of **Circular Edges** is identical to the surface normal.

Cylindrical Surfaces (Figure 10) are defined as follows:

- They consist of two **Circular Edges**. If the **Circular Edges** are arcs, we say it is a segment. In this case the arcs are connected by **Straight** or **Round Edges**, otherwise the connections are **Virtual Edges**.
- The centers of the **Circular Edges** are forming the axis and both **Circular Edges** are perpendicular to this axis.
- One of the **Circular Edges** may have radius zero. In this case we say it is a cone.

The number of surface types is not restricted. If a new tool for detecting a certain structure in the range data is developed, a new surface model may be added.

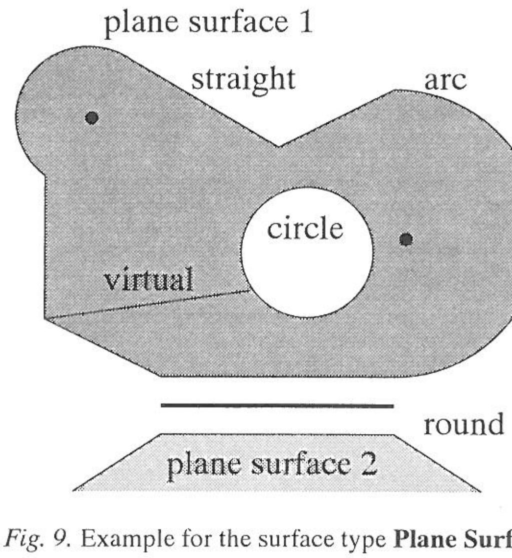


Fig. 9. Example for the surface type **Plane Surface**

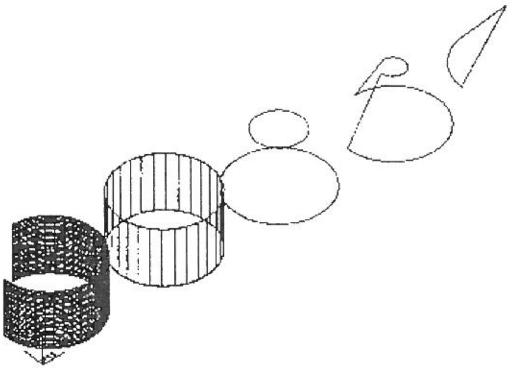


Fig. 10. Examples for the surface type **Cylindrical Surface.** Cone segment, frustum segment, frustum, cylinder, and cylinder segment.

6. Results

The described 3D model is realized by several applications introduced in this section. They were implemented in C under UNIX and work together with the 3D robot sensor introduced in section 7.

- **File format**

The file format is a readable text. The syntax is rather easy and described explicitly in [5]. Figure 11 shows an example.

All vertices are defined by their three dimensional coordinates in a so called object coordinate system. This is just for defining the object and may be chosen freely.

An object is defined by its surfaces, so the definition is just a list of surface definitions. It starts with the number of edges and the surface type, `e` for **Plane Surfaces** and `z` for **cylindrical surfaces**. The actual definition consists of numbers of vertices and parameters for the edges alternating with each other. The following example definitions are surfaces of the object in Figure 11.

```
%F    11
surface with hole:
7 e    4  a[16,17] 7 g 6 g 5 g 4 v
      23 k1 23 v
small triangle:
3 e   32  g  6 g  7 a[2,29]
front:
6 e    4  g  5 g 15 g 12 a[12,0] 0
      g 1 a[1,4]
drill-hole:
4 zi  22 k0 22 v 26 k4 26 v
```

- **Data structure**

The data structure is designed for fast access. The structure is very similar to that in Figure 5, but there are some additional information and derived parameters added.

- **Library**

A library for IO operations (box "IO" in Figure 12) maintains a database of objects, computes the additional parameters, administers files and the data structure.

geometry	circular edges	radii	straight edges
cylinder	circles	equal	virtual
cylinder segment	arcs	equal	straight or rounded
cone	circles	one equals zero	virtual
cone segment	arcs	one equals zero	straight or rounded
frustum	circles	unequal	virtual
frustum segment	arcs	unequal	straight or rounded

Table 2. Special cylindricals and their structure

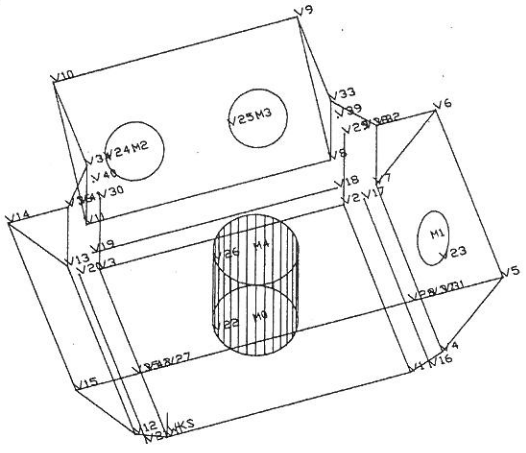


Fig. 11. Example object with vertex numbers

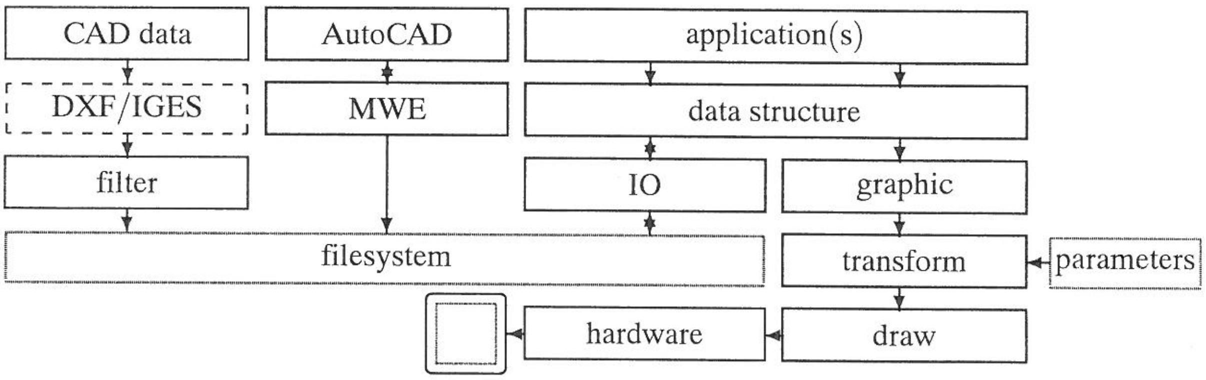


Fig. 12. Overview of the implemented system

- **Visualization**

The visualization is a graphic tool for the display of results. It is also an example for an application (box "application(s)" in Figure 12) using the library.

It simulates the way in which the camera takes pictures of the world. Using the model of the objects and the evaluated object- and pose–hypothesis, it is possible to overlay the results of the pose-estimation to the original (depth or grey) image. This allows interactive verification of the results.

Figure 13 shows a typical result of this visualization tool.

- **Graphic editor**

The interactive construction of objects requires a three dimensional graphic editor with syntactic user guidance and semantic check of the constructed objects. This editor is called MWE (model world editor) and as you see in Figure 12 (box "MWE") it is attached to a CAD-system (box "AutoCAD"[1]). This CAD-system offers the possibility to add applications which use the graphic interface of the system.

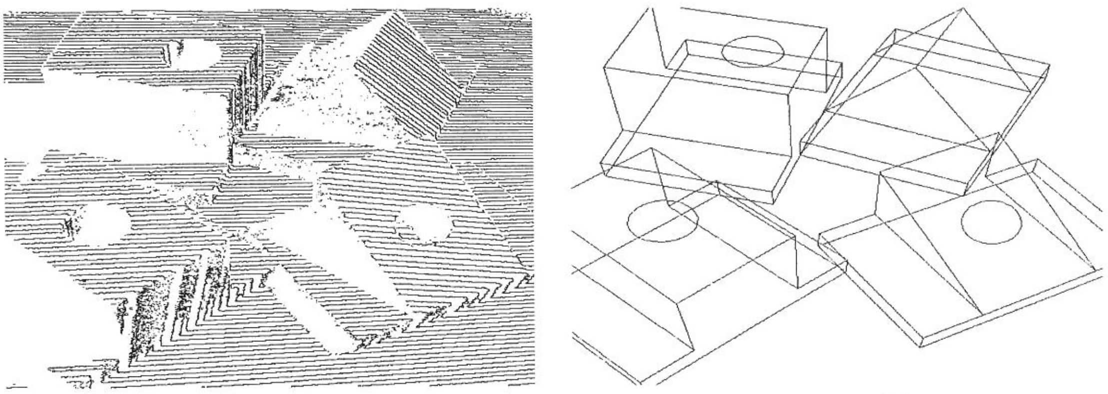
Figures 7, 8, 10 and 11 are output of this editor.

7. The 3D Robot Sensor

The modeling conception presented above was developed as a part of a 3D robot sensor system. The complete system is described in [17, 19]. But it is not restricted to this system, it should work with any kind of range data and most

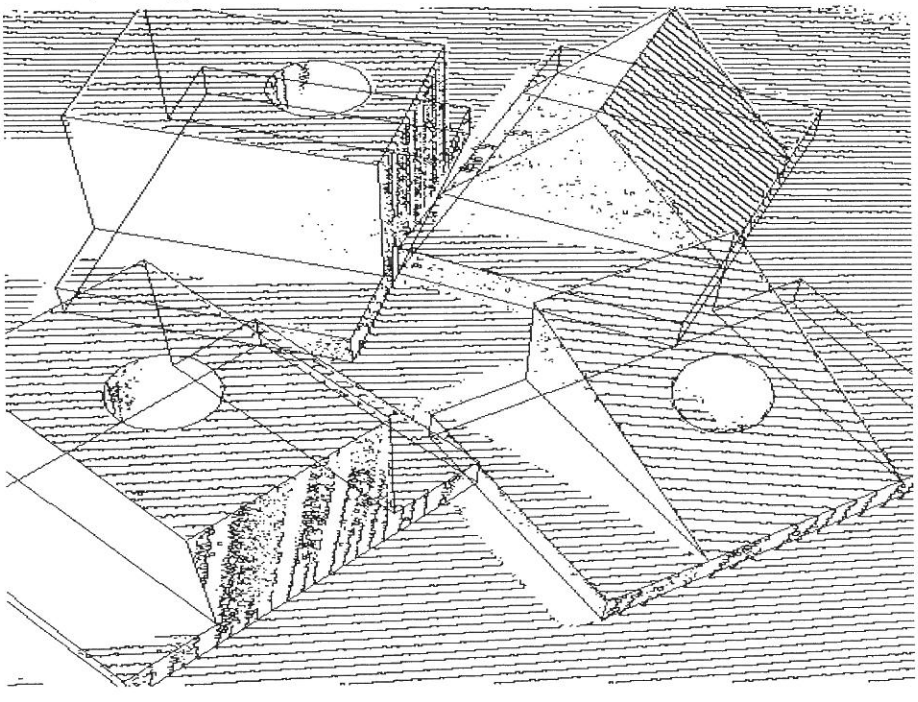
---

[1] **AutoCAD** is a trademark of Autodesk AG.



depth image

model



model overlayed to the depth image

Fig. 13. Visualization of object and pose-hypotheses

recognition strategies. So the following is just an example how the model can be used.

The 3D robot sensor allows the generation of range images of a scene on request. Therefore the camera is attached to the gripper of a robot.

Recognition and pose estimation are based on local constellations of geometric scene primitives as described above. They are matched against corresponding structures in the model. As the number of comparisons which are necessary for this process is very large, it is urgent to optimize the efficiency of this operation.

- By modeling only parts of the object the number of comparisons is reduced.
- The tree structure allows very fast access.
- Compared parameters (e.g. the angle between two neighbor planes) are preprocessed and provided by the model.
- The restriction on geometric primitives reduces the number of parameters which are to be compared.
- For every combination of primitives (e.g. two **Plane Surfaces** plus angle) the frequency of its appearance in the whole model world is calculated. Using this measure it is possible to search first for the most unusual and therefore most characteristic features.

The whole recognition process is controlled by a hypothesis generation/hypothesis verification

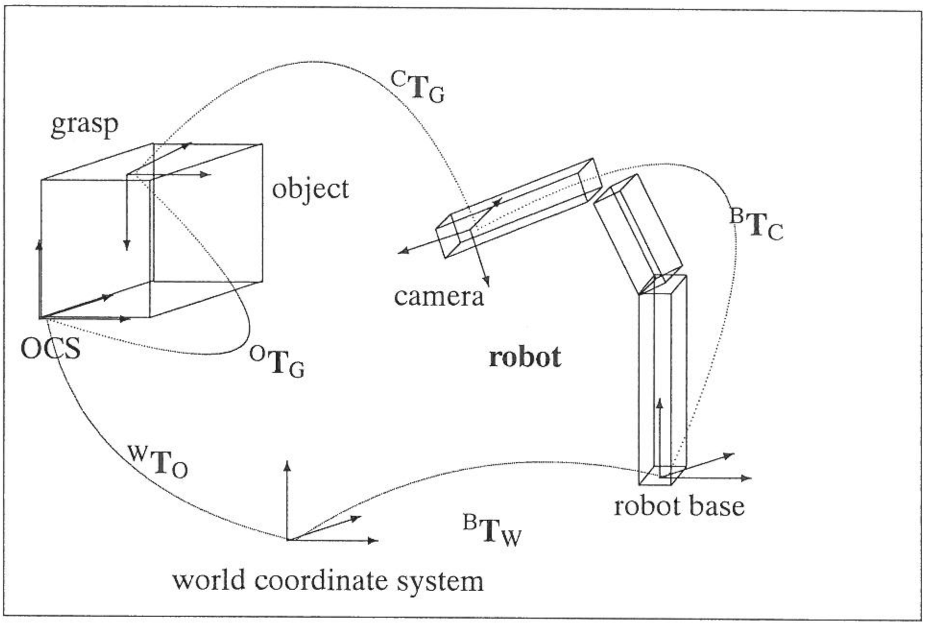


Fig. 14. 3D robot sensor system

scheme [11, 13]. After analysis of a first image, there may be multiple hypotheses for one object. To choose the right one it may e.g. be necessary to see the back of the object. As the camera is attached to a robot (Figure 14), it is possible to acquire additional views of the scene. Many problems with $2\frac{1}{2}$ dimensional data can be solved in this way, especially in industrial environments.

8. Summary

Summing up, the described method of modeling 3D objects for the interpretation of range images has the following main features. It is very suitable for industrial environments, where every object has some geometric primitives in the shape of the described surface models. The model represents circles, cylindricals and neighborhood which is very useful with the used range data. The modular structure allows the addition of new edge and surface models.

9. Acknowledgement

This work was mainly done at the Institute of Robotics and Computer Controll (IRP) at the Technical University of Braunschweig, Germany.

References

[1] Bruce R. Altschuler, editor. Three-d machine perception, Bellingham, Washington D.C., April 1981.

[2] Martin D. Altschuler, Bruce R. Altschuler and J. Taboada, Measuring surfaces spacecoded by a laser-projected dot matrix, *Imaging Applications for Automated Industrial Inspection & Assembly*, Bellingham, Washington D.C., 1979.

[3] Martin D. Altschuler, Kyontae Bae, Bruce R. Altschuler, Jerome T. Dijak, Louis A. Tamburino and Barbara Woolford, Robot Vision by Encoded Light Beams, editor Takeo Kanade, *Three-Dimensional Machine Vision*, pp. 97–149, Kluwer Academic Publishers, Boston, Dordrecht, Lancaster, 1987.

[4] Paul Besl, Active Optical Range Imaging Sensors, editor Sanz, Jorge, *Advances in Machine Vision*, pp. 1–63, Springer Verlag, Berlin, 1988.

[5] Jochen Duckeck, Objekterkennung mit einem 3D–Robotersensor: Die Modellwelt, Technical University of Brunswick, Institute for Robotics and Computer Control, December 1992.

[6] H. Grasmüller, K. Hohberger, P. Köllensperger, E. Kutzer, A. Reischl and P. Rummel, Local Feature Extraction for Model-Based Workpiece Recognition, *Proceedings 7th Int. Conf. on Pattern Recognition*, pp. 886–889, Montreal, 1986.

[7] RALF GUTSCHE, THOMAS STAHS AND FRIEDRICH M. WAHL, Path Generation with a Universal 3d Sensor., *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 838–843, Sacramento, California, April, 1991.

[8] T. GLAUSER H. BUNKE, Affine Invariant Representation and Recognition of Polygonal Faces in 3-D Space using Edge Length Ratios, Tehnical report IAM 92–014, Institute for Computer Science, University of Bern, Switzerland, 1992.

[9] FRANK HAFERKAMP, Zylindererkennung mit einem 3D-Robotersensor, Technical University of Brunswick, Institute for Robotics and Computer Control, 1993.

[10] X. Y. JIANG AND H. BUNKE, Fast Segmentation of Range Images into Planar Regions by Scan Line Grouping, Tehnical report IAM 92–006 Institute for Computer Science, University of Bern, Switzerland, April 1992.

[11] CHRISTOPH KOHNERT, Objekterkennung mit einem 3D-Robotersensor: Die Hypothesengenerierung, Technical University of Brunswick, Institute for Robotics and Computer Control, 1992.

[12] PETER KOLLER, Computation of range images from color-stereo-images by 'Simulated Annealing', Pavešić, N. and Niemann, H. and Paulus, D.editors, *Image Processing and Stereo Analysis, Proceedings of the Slovenian-German Workshop*, volume 26/1, pp. 119–130, Erlangen, February 1993.

[13] STEPHAN LOCHMANN, Objekterkennung mit einem 3D-Robotersensor: Die Hypothesenverifikation, Master's thesis, Technical University of Brunswick, Institute for Robotics and Computer Control, 1992.

[14] T. LOZANO-PÉRES, W. E. L. GRIMSON, AND S. J. WHITE, Finding cylinders in range data *Proceedings of the IEEE International Conference on Robotics and Automation*,pp. 202–207. Raleigh, NC, 1987.

[15] ROLAND ROBMANN, Ein kantenorientierter Ansatz zur Integration von Grauwert– und Tiefenbildern (An Edge Oriented Approach for Grey Level and Range Image Integration), Technical report IAM 93–016, Institute for Computer Science, University of Bern, Switzerland, August 1993.

[16] T. M. SILBERBERG, D. HARWOOD AND L. S. DAVIS, Three Dimensional Object Recognition Using Oriented Model Points, editor A. Rosenfeld, *Techniques for 3-D Machine Perception*, pp. 271–320, North–Holland, Amsterdam, New York, Oxford, Tokyo, 1986.

[17] THOMAS STAHS, Ein 3D Robotersensorsystem auf der Grundlage einer verallgeminerten Methodik zur Erstellung modellbasierter Objekterkennungsverfahren, PhD thesis, Technical University of Brunswick, Institute for Robotics and Computer Control, 1994.

[18] THOMAS STAHS AND FRIEDRICH M. WAHL, Fast and Robust Range Data Acquisition in a Low–Cost Environment, *Proceedings of the ISPRS-Conference: Close-Range Photogrammetry meets Machine Vision*, volume 1395, pp. 496–503, Zürich, 1990.

[19] THOMAS STAHS AND FRIEDRICH M. WAHL, Objekterkennung und Lagebestimmung mit einem 3D-Robotersensor, *Informationstechnik und Technische Informatik, it + ti*, 36, (1): 39–46, 2, 1994.

[20] ANDREAS UELTSCHI, Modellbasierte Objekterkennung in Tiefenbildern, Technical report IAM 93–018 Institute for Computer Science, University of Bern, Switzerland, August 1993.

[21] FRIEDRICH M. WAHL, A Coded Light Approach for Depth Map Acquisition, G. Hartmann, editor, *Proceedings 8. DAGM-Symposium*, Informatik Fachberichte 125, pp. 12–17, Springer Verlag, Berlin, 1986.

Received: November, 1994
Accepted: March, 1995

Contact address:
Bavarian Research Center
for Knowledge Based Systems
(FORWISS)
Knowledge Processing Research Group
Am Weichselgarten 7, D–91058 Erlangen, Germany
email: {duckeck,kirchner}@forwiss.uni-erlangen.de

JOCHEN DUCKECK received the Dipl.-Inform. degree in computer science from the Technical University of Brunswick, Germany, in 1993. Since 1993 he is with the Bavarian Research Center for the Knowledge Based Systems (FORWISS) in Erlangen, Germany. His field of research is knowledge-based realtime image processing.

HARALD KIRCHNER received the diploma of computer science and the Dr.-Ing. degree from the University of Erlangen-Nürnberg, Germany, in 1987 and 1992, respectively. From 1987 to 1992, he worked with the Pattern Recognition department of the University of Erlangen-Nürnberg, working in the field of image processing, image sequence analysis (motion detection, detection of moving objects) and computer vision. He is presently assistant head of the Knowledge Processing Research Group at the Bavarian Research center for Knowledge Based System (FORWISS) in Erlangen, Germany.