An algorithm for the inexact matching of high level 3D polyhedric representations

Hans Peter Amann, Heinz Hugli

University of Neuchâtel, Institute for Microtechnology
Rue A.-L. Breguet 2, CH-2000 Neuchâtel, Switzerland

Abstract

This paper describes a powerful inexact matching algorithm which has been applied with success to high-level 3D object representations in a 3D object recognition system. The algorithm combines in a promising way several approaches proposed in the last couple of years: an extension to the backtrack strategies for inexact matching of attributed relational sub-graphs, error correction isomorphism, determination of local attribute similarity and global transformation fitting, features which are efficiently used for search-tree pruning. The algorithm was tested successfully in a series of experiments involving scenes with single and multiple objects.

Index terms: 3D object recognition, scene analysis, inexact sub-graph matching, machine vision

1. Introduction

Vision is the most important way for humans to localize, to estimate shapes and to recognize objects in the real world. Therefore, quite naturally, efforts have been made since a couple of years in order to integrate similar facilities in automatic vision systems. Two typical examples of application domains are assembly lines and robot vision. Rather good results have been obtained for two-dimensional objects in artificial, human-made environments and for well defined problems. For three-dimensional objects in turn, the efforts made have, up to now, essentially shown the high complexity of the problem. Nevertheless, researchers are now in the position to test particular 3D object recognition algorithms, to realize prototypes with limited features and to apply them to well defined, relatively simple problems.

The problem

The problem of 3D object recognition using machine vision can roughly be described as follows:

> Given a scene containing one or more 3D objects and a library of reference objects, identify one (or more) of the 3D scene objects either partially or completely with respect to the given reference object library's elements.

A frame for an object recognition system

This rough problem definition gives rise to a frame for an object recognition system, consisting of the following elements:

i) Acquisition of the rough 3D data and extraction of a convenient high level object representation.
ii) Matching of the acquired and extracted high-level representation with some reference or model representations, resulting in a list of hypotheses on the best correspondence.
iii) Verification and classification of the resulting hypotheses, the combination of compatible partial results, the extraction of useful information on orientation, the localization of recognized elements etc.

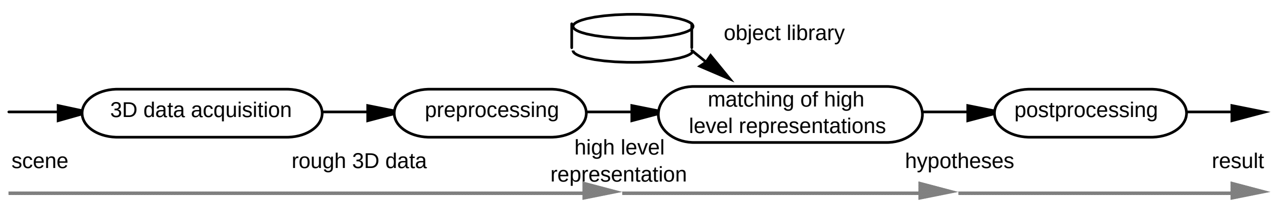

fig 1.1: a frame for an object recognition system

Previous work

This work is based on several papers published during the last couple of years: Tsai and Fu [1] proposed an algorithm for error correction isomorphism, Eshra and Fu [2, 3] extended the idea in their backtrack strategy for inexact matching of attributed relational sub-graphs. For search tree pruning, some ideas of Shapiro [4] concerning similarity measures and of Faugeras and Hebert [5] concerning global transformation fitting have been used.

Contents of the paper

Amann H.P.& Hügli H., "An algorithm for the inexact matching of high level 3D polyhedric representations", Model-Based Vision Development and Tools, SPIE proceedings Vol 1609, pp 134-140, 1991
(Proc. SPIE "Intelligent Robotic Systems", Boston, 10-15 November 1991)

In the present paper, we describe first the preprocessing which is applied to the rough scene data (§2). Once this preprocessing done, we can define the object recognition problem more precisely as a (sub-)graph matching problem. Inherently, due to the nature of the acquired data and the preprocessing, the matching will be inexact, giving rise to a possibly complex search of correspondences. Implementation dependent local and global criteria based on heuristics allow to keep the complexity of the process reasonably low. Next, we discuss some of the related, previous work (§3). Many of the ideas that have been integrated in our implementation of the matching algorithm which are discussed in paragraph 4, including some remarks on the postprocessing. The remaining paragraphs are devoted to the presentation of some results (§5), the concluding remarks, an out-view on further subjects of interest and finally the references.

2. Acquisition of data and the extraction of high level representations

In order to give a complete view of the environment for which the algorithm has been developed, we present shortly the data acquisition device and the preprocessing applied (figure 2.1).

2.1 Data acquisition and preprocessing

The input device is a laser range finder using light planes [6], which delivers a huge set of 3D measurements of the visible surfaces. A region growing process is used in order to approximate the 3D point set by surfaces, in our implementation patches of planar surfaces [7, 8, 9]. The resulting set of surface patches, standing for the scene objects visible surfaces, can now be represented at a higher level of abstraction, e.g. in the form of attributed relational graphs (ARGs). Obviously, these ARG representations are neither complete (occlusion) nor precise (limited resolution, preprocessing, choice of attributes, etc.) representations of the scene objects.

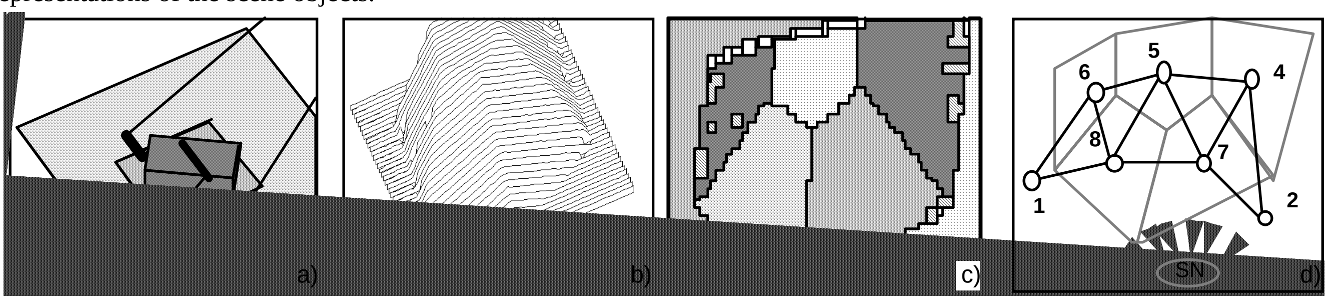

figure 2.1: from 3D data acquisition to high-level representation:
a) laser range finder, b) 2.5D depth map, c) region/border representation, d) attributed relational graph (ARG) representation

2.2 High level representation: our choice

Two constraints have been specified for our problem: i) the limitation to polyhedric objects and ii) the use of a high-level representation based on planar surfaces. While both constraints do not modify the nature of the important problems in 3D object recognition, they reduce significantly the complexity of the implementation.

We use a high-level representation called planar face representation graph (PFRG), a special form of an ARG. Each node stands for a region of an object, approximated by a planar surface patch and each arc for a neighborhood-relation between two regions. The node attributes are the area and normal vector of the represented surface. The only arc attribute is the length of the border between adjacent regions.

2.3 Preprocessing

PFRGs of reference objects will be rather concise, PFRGs of acquired scene data are probable to be cluttered due to the reasons mentioned before (§2.1). Therefore, some simple algorithms have been used which merge regions which have accidentally been separated and which remove dummy regions, essentially the ones situated near to the borders of real surfaces [10].

2.4 Re-definition of the task

Once the high level representation and its relation to real world data being defined, we now can give a more exact task description:

> With both the scene and the reference objects represented in the form of PFRGs, establish an ordered list of hypotheses on similar (sub-)PFRGs, taking into account both structural and attributal similarities.

3. Related work

Many of the ideas implemented in our algorithm of paragraph 4 for the inexact matching of PFRGs are issued, or have been triggered by work done in the last couple of years: Tsai and Fu [1] established a first base for error-correcting isomorphic matching. This work has been continued by Eshra and Fu [2, 3] in their backtrack-based inexact matching scheme. Shapiro's paper [4] on the use of metrics was very helpful for the determination of dissimilarities between attributed sub-graphs. The important concept of object rigidity (§4), and the mathematical tools necessary for its convenient implementation as a global search tree heuristic, is due to Faugeras and Hebert [5], while Oshima and Shirai [11] discuss the interesting feature of “root-matches” which allows the determination of interesting start points for the search trees.

4. Inexact matching of high level representations

The present paragraph is devoted to the inexact matching of two high level 3D polyhedric representations. Given a scene and a reference PFRG we seek a list of promising matches M which can be both partial and inexact. For each of the matches M, we measure also the overall similarity as well as the transformation T the reference object would have to undergo such that it matches "best" the scene object (figure 4.1).

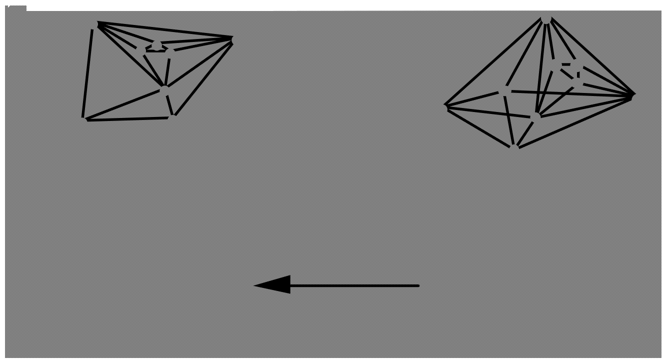

fig 4.1: the "matching" process: left the scene object and its PFRG, right the reference object and its PFRG

4.1 Notations and definitions

Scene and reference PFRGs are denoted with $G_S = (N_S, A_S)$ and $G_R = (N_R, A_R)$ respectively, where N_S and N_R stand for the respective node sets and A_S and A_R for the respective sets of arcs. $A(N_{Sm}, N_{Sn})$ and $A(N_{Rm}, N_{Rn})$ stand for a non-directed arc interconnecting adjacent scene and reference nodes respectively. Finally, $M(N_S, N_R)$ stands for an established correspondence between scene node N_S and reference node N_R M is a set of 1-1 node correspondences between N_S and N_R.
With these notations, we can define the problem definition more precisely:

> Given two planar face representation graphs (PFRGs) G_S and G_R, find the "best" set M of 1-1 scene-node / reference-node correspondences according to an overall "cost" function.

Note that using this approach, the match is guided by the establishment of node-node correspondences and their respective dissimilarities. While arc-arc correspondences will be established implicitly where possible (inexact matching) and therefore will participate in the global 'cost'-function, missing nodes will neither be detected nor considered.

4.2 A state space lattice for the search of the best match

In order to find the "best" match between reference PFRG G_R and scene PFRG G_S, we apply a "best-first" search algorithm in a state-space lattice where each state S_k corresponds to a set of 1-1 node correspondences M_k. In our implementation, this match is represented by the core-node sets $C_{Rk} = \{.., N_{Ri},..\}$ and $C_{Sk} = \{.., N_{Sj},.\}$ resp. and the set of 1-1 correspondences $M_k = \{.., (N_{Ri}, N_{Sj}),..\}$ of their nodes. Furthermore, each state S_k has an associated set of terminal nodes T_{Rk} resp. T_{Sk}.

Amann H.P.& Hügli H., "An algorithm for the inexact matching of high level 3D polyhedric representations", Model-Based Vision Development and Tools, SPIE proceedings Vol 1609, pp 134-140, 1991
(Proc. SPIE "Intelligent Robotic Systems", Boston, 10-15 November 1991)

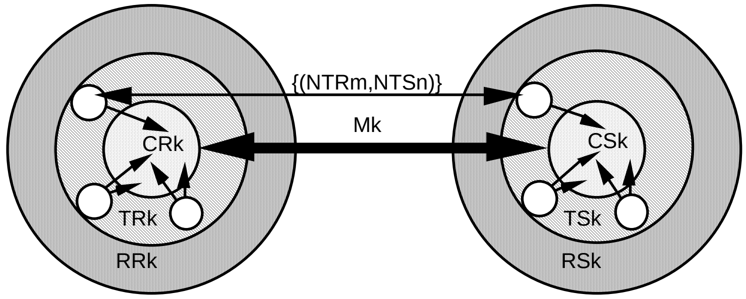

figure 4.2: reference node set (left) and scene node set (right)

These sets contain all the nodes which have at least one arc to any of the core nodes of the respective core node set. The remaining nodes are contained in the rest node sets R_{Rk} and R_{Sk}. Each state S_k of the search tree has an associated set of attributes: i) the overall cost of the match M_k, ii) the "best-fitting" transformation T_k and iii) the number of arcs inserted or deleted in order to obtain isomorphism. With the search strategy selected ("best-first"), the leaf state with the minimum overall cost is expanded first. Obviously, a non-restricted expansion of the search tree would lead to a non-affordable amount of computation. Therefore, we introduce a set of search tree pruning criteria. The most important ones will be discussed (§4.3), for a complete discussion see [10].

4.3 Search tree pruning criteria

Connectivity condition

The first search-tree pruning criterion, the connectivity condition, has already been introduced before: a sub-graph match can only be extended by pairs of terminal nodes. In the present design, this rule is particularly weak with respect to the rules introduced in previous work [3]. It allows for good flexibility in inexact matching while still limiting the expansion of the search tree. If ever the application of this rule would lead to multiple sub-graph matches, post-processing could help to resolve this problem (§4.5).

Rigidity condition

For common objects, it is straightforward to introduce the hypothesis of rigidity: hence, the angles between the surface normals are constant and invariant to rotation and translation. (While the approach is more general, the use of PFRGs limits us currently to rotations). We therefore determine for sub-graph matches M_k a "best fitting rotation" Rot_{Sk} for the set of unitary normal vectors of the respective core node sets [5, 12]. Knowing the rotation Rot_{Sk}, we can restrict the set of possible node matches (NT_{Sm}, NT_{Rn}) to the couples of nodes which fulfill the rigidity condition; the difference of orientation must be smaller than a given threshold: $|\sphericalangle (normal(NT_{Sm}), (Rot_{Sk}(normal(NT_{Rn}))))| < Th_{Rot}$.

Visibility condition

With conventional data acquisition equipment, a single view of the scene is taken. Thus, visible faces have an orientation such that the angle between the observation vector and the surface normals is smaller then $\pi/2\pm\varepsilon$. The transformation of the observation vector back to the reference space allows hence to prune the set of reference terminal nodes depending on the surface normal attributes.

Neighbor match condition

The candidate couple (NT_{Rm}, NT_{Sn}) is candidate for a new match if and only if there exist at least one adjacency $A(NT_{Rm}, NC_{Ri})$, $A(NT_{Sn}, NC_{Sj})$ and a core-node correspondence (NC_{Ri}, NC_{Sj}) as elements of the previous match set M_k. This condition guarantees that the terminal nodes have adjacencies to at least one matched pair of core nodes: neighboring reference nodes will hence be matched with neighboring scene nodes.

Cost conditions

The transition of one state S_k in the search tree to a next state S_i corresponds to the extension of the match M_k to match M_i by a new couple of nodes (NT_{Rm}, NT_{Sn}), modifying the overall matching cost of paragraph 4.4 by $\Delta cost(S_i, S_k)$. With the idea in mind that search branches which would lead to an important increment in the global cost should be discarded, this "modification-cost" can be used as an additional pruning condition. Moreover, particular similarity conditions can be established for each attribute individually.

4.4 The cost function

In order to determine an overall cost function, the dissimilarities determined for each type of attribute and for structural differences are combined such that search-tree states of different levels, representing matches of different sizes, can be compared: the weighting must depend on the number of elements involved.

$$\boxed{\mathrm{cost}(S_k) = \frac{\alpha\, \Sigma_{area}(S_k) + \varepsilon\, e_{rotfit}(S_k)}{\text{number of nodes}(S_k)} + \frac{\beta\, \Sigma_{border}(S_k) + \gamma\, A_{ins}(S_k) + \delta\, A_{del}(S_k)}{A_{match}(S_k) + A_{ins}(S_k) + A_{del}(S_k)}}$$

figure 4.3: the cost function

The cost function used is composed of two parts, i) the weighted sum of the accumulated dissimilarities of area of the matched surface patches $\Sigma_{area}(S_k)$ plus the mean squared rotation fitting error $e_{rotfit}(S_k)$ divided by the number of matched nodes and

ii) of an arc dissimilarity measure composed of the sum of the arc attribute dissimilarities $\Sigma_{border}(S_k)$, increased by a dissimilarity measure for inserted arcs (A_{ins}) and deleted arcs (A_{del}) divided by the number of arcs involved (matched, inserted and deleted). The weights α, β, γ, δ and ε have been determined heuristically by a series of experiments on real data.

4.5 Postprocessing

Postprocessing covers all the steps subsequent to the match necessary in order to complete the recognition task. Let us just mention the two most interesting ones among them: i) the combination of partial hypotheses and ii) the verification of the results at lower levels of abstraction For further details refer to [10].

5. Results

Determination of the weighing parameters

In a first series of tests, the weighting factors for the overall cost have been determined. It showed up that for all of the subsequent experiments, a single, well chosen set of parameters was sufficient (α=0.05, β=0.04, γ=0.01, δ=0.01, ε=0.9).

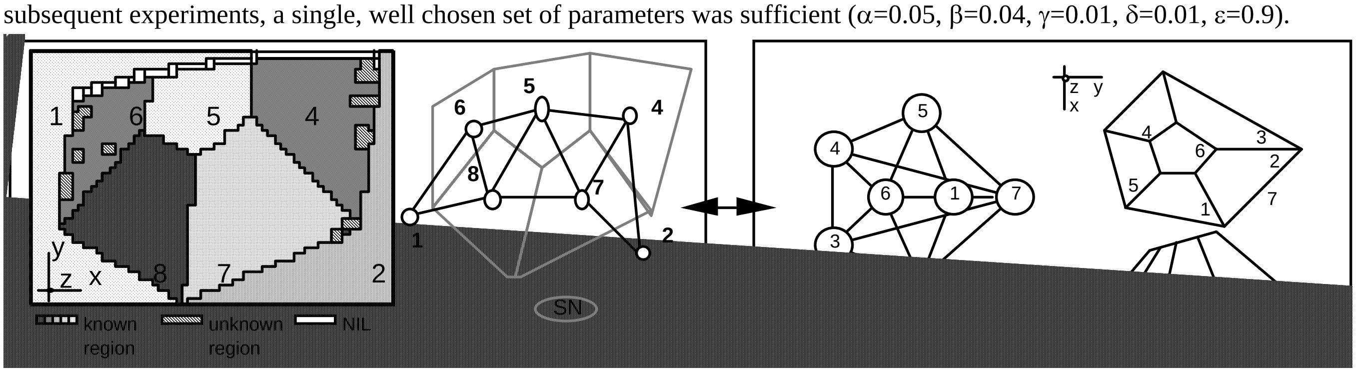

figure 5.1: determination of the weighting parameters (test object "P" left and reference object "Pyramid2" right)

Object recognition: single-object / single-reference

The next tests have been conducted in order to check the selectivity achieved with the defined cost function. The results show that a system which uses exclusively surface orientations as attributes, e.g. using gaussian images, does not behave sufficiently well. With the use of the surface area attribute (corresponding to a representation using extended gaussian images EGIs [13]) and the border-length attribute, the selectivity becomes sufficient despite of the simplicity of the chosen representation. Obviously, both the surface area attribute and the border-length attribute are sensitive to occlusion. Therefore, a reliability measure should be introduced which allows to weigh the influence of the node matches and arc matches in the overall cost function.

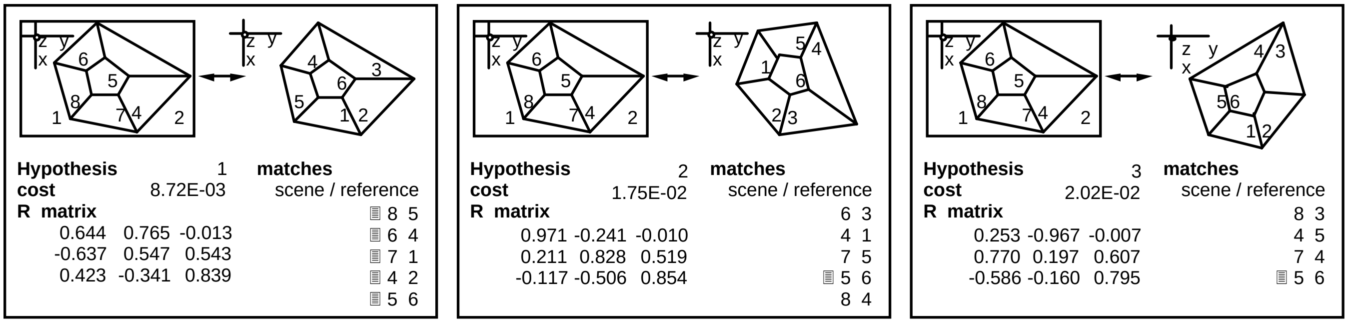

figure 5.2: the first three hypotheses for the problem of figure 5.1

Figure 5.2 shows from left to right the first three hypotheses obtained for a matching attempt between the scene object "P" (fig. 5.1, left) and the reference object "PYRAMID2" (fig 5.1, right).
The first hypothesis is the correct one, the second and the third, having considerably higher costs, represent the matches made with the scene object rotated about the z-axis "one face left resp. right". The evolution of the cost, the number of node correspondences and the number of correct node correspondences are shown in figure 5.3.

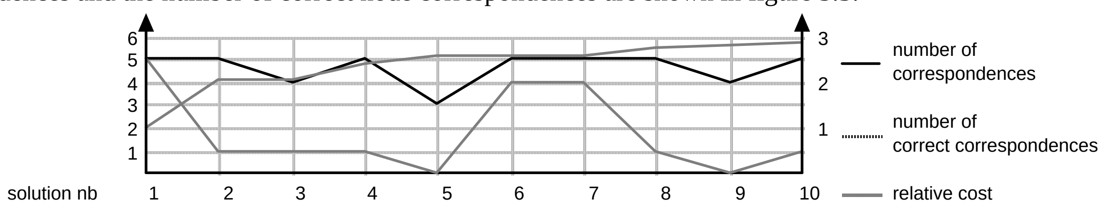

figure 5.3: statistics for the first ten solutions for the problem of figure 5.1

Object recognition and separation

After a series of tests with single-object scenes, the more ambitious task of multiple-object scenes has been addressed. The PFRG of a scene "CP", showing a cube which occludes partially a pyramid, has been tested vs. the reference PFRGs "CUBE1" and "PYRAMID2".

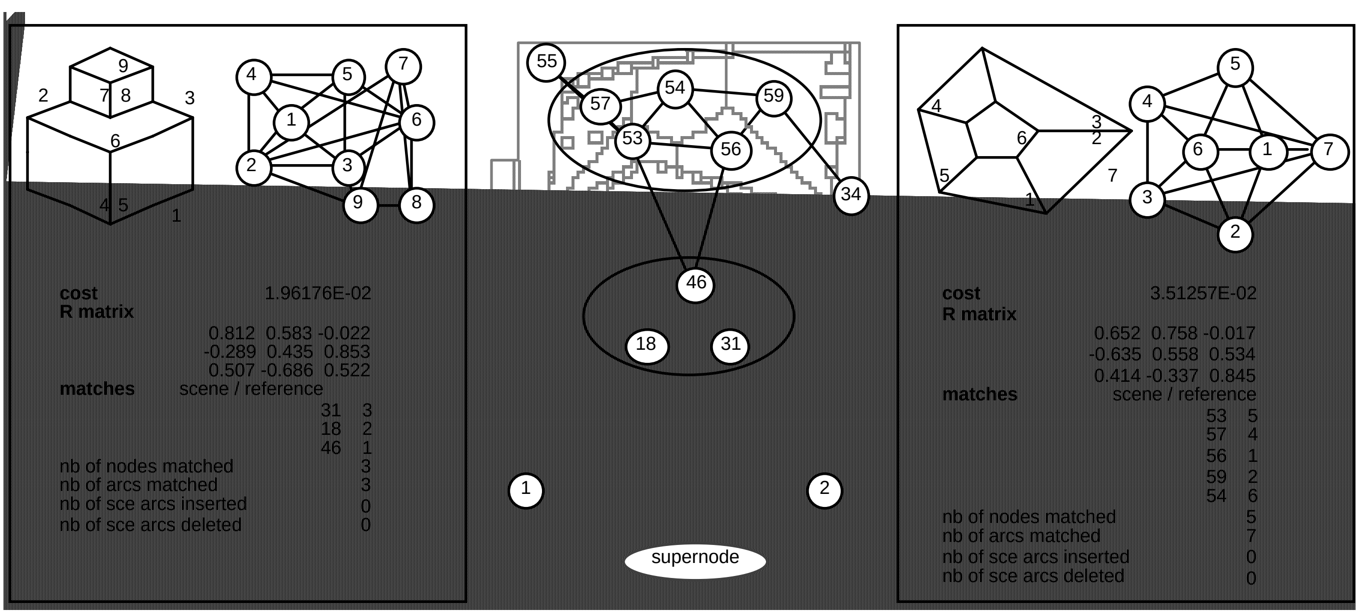

fig 5.4: two compatible solutions: scene “CP” (center) vs. references "CUBE1" (left) and "PYRAMID2" (right)

With the algorithm described before and a simple test of compatibility of the solutions, the desired result, represented in figure 5.4, has been obtained. The "best" set of compatible solutions contains the correct match between the reference object "CUBE1" and the part of the scene "CP" representing the cube and the reference object "PYRAMID2" with the part of the scene representing the pyramid.

Amann H.P.& Hügli H., "An algorithm for the inexact matching of high level 3D polyhedric representations", Model-Based Vision Development and Tools, SPIE proceedings Vol 1609, pp 134-140, 1991
(Proc. SPIE "Intelligent Robotic Systems", Boston, 10-15 November 1991)

Nevertheless, with our choice of high-level representation and correspondence search criteria, the reliability of the correspondences decreases rapidly in the case of partially occluded objects. The results of the match must be completed, either by subsequent (low-level) verification or by the use of richer data in both the high-level representation and the correspondence search criteria.

6. Conclusions and outview

In the present paper, we have presented a solution to the inexact high-level matching task in the context of 3D object recognition. Given two high level object representations in the form of attributed relational graphs, we extract the most promising (partial) matches so that we can establish hypotheses on the scene objects. The matching uses a similarity measure for the matched parts of both reference and scene objects as well as a transformation the reference object has to undergo such that it fits best the scene data, are determined.

We treat the subject of graph matching as a tree-search problem, using a "best-first" search. The present work is particular in the sense that it uses a very weak rule for the tree expansion such that a wide spectrum of topological inexactitude can be accepted. Usually, this inexact matching feature must be payed heavily by an important increase of the computational complexity. Therefore, a number of criteria have been introduced which allow to prune the search tree efficiently. The most important condition we use is the rigidity condition. Despite of the good results obtained with this single condition, additional tree-pruning conditions based on the remaining graph attributes had to be added in order to deal with cases where the former condition is not sufficient, e.g. for man-made objects where perpendicular and parallel faces are common.

The matching is subject to the structural and attributal similarity of the scene and reference objects. A final match is characterized by an overall similarity measure as well as by the geometrical transformation that leads best from the reference to the scene. Various experiments have shown good results obtained with the described algorithm, i) for object recognition with single object scenes and ii) for object recognition and object separation for multiple object scenes, including objects which occlude themselves mutually.

As such, the present solution should be considered as a building block for a more elaborate recognition system that considers also , among other things, verification of the generated hypotheses at a lower level of description and object representation involving additional aspects like more complete descriptions of edges and surface patches.

7. Acknowledgements

This work has been supported by the Swiss National Foundation for Scientific Research under project numbers FN2265, FN2000-5.138 and FN 20-25572.

8. References

1. W. H. Tsai, K. S. Fu, "Sub-graph error-correction isomorphism for syntactic pattern recognition" IEEE Trans. on Systems, Man, Cybernetics, Vol 13, No 1, 1983
2. M. A. Eshera, K. S. Fu, "A similarity measure between Attributed Relational Graphs for Image Analysis" 7th Conf Pattern Recognition, Montreal 1984
3. M. A. Eshera, K. S. Fu, "An Image Understanding System Using Attributed Symbolic Representation and Inexact Graph Matching", IEEE PAMI-8, Vol5, Sept 1986
4. L. Shapiro, R. M. Haralick, "A Metric for Comparing Relational Descriptions" IEEE PAMI-7, No 1, January 1985
5. O. D. Faugeras, M. Hebert, "The Representation, Recognition and Positioning of 3D-Shapes from Range Data" in "Three-Dimensional Machine Vision", ed T. Kanade, Kluwer Academic Press, 1987, pp301
6. M. Rioux, "Laser range finder based on synchronized scanners" Applied Optics, Vol 23, No 21, Nov 1984, pp. 3837-3844
7. G. Maître, H. Hugli, F. Tièche, H.P. Amann, "Range Image Segmentation based on function approximation" Proceedings ISPRS90, Zürich, Switzerland, Sept 1990
8. G. Maître, H. Hugli, "Range Image Segmentation by controlled-continuity spline approximation for parallel computing", Proceedings SPIE Conf. on Intelligent Robotic Systems, Boston, November 1991
9. P. J. Besl, R. C. Jain, "Segmentation through variable-order surface fitting" PAMI-10, No 2, March 88
10. H. P. Amann, "3D Object Recognition Based on Surface Representations" PhD thesis, University of Neuchâtel (Switzerland), Oct 1990

11. M. Oshima, Y. Shirai, "An object recognition system using three-dimensonal information" in "Three-Dimensional Machine Vision", ed T. Kanade, Kluwer Academic Press, 1987, pp355

12. I. L. Kantor, "Hypercomplex numbers: An elementary introduction to algebras" Springer, New York, 1989

13. B.K.Horn, "Extended Gaussian Images" Proc. IEEE, Vol 72, No 12, Dec 1984

Amann H.P.& Hügli H., "An algorithm for the inexact matching of high level 3D polyhedric representations", Model-Based Vision Development and Tools, SPIE proceedings Vol 1609, pp 134-140, 1991
(Proc. SPIE "Intelligent Robotic Systems", Boston, 10-15 November 1991)