Geometric Matching for Free-Form 3D Object Recognition

Heinz Hügli, Christian Schütz, Dimitrios Semitekos

University of Neuchâtel
Institute for Microtechnology
Rue Tivoli 28
CH-2003 Neuchâtel
Switzerland

hugli@imt.unine.ch

Abstract

This paper investigates a new approach to the recognition of 3D objects of arbitrary shape. The proposed solution follows the principle of model-based recognition using geometric 3D models and geometric matching. It is an alternative to the classical segmentation and primitive extraction approach and provides a perspective to escape the difficulties found with it when dealing with free-form shapes. The heart of this new approach is geometric registration which is performed by a closest point matching algorithm. Reported investigations examine the practical effectiveness of this approach for views obtained from range imaging and address relevant aspects of associated computational costs. The paper proposes solutions allowing to keep track with these costs and presents results assessing the practical feasibility of this approach.

Introduction

Traditional approaches to 3D vision proceed according to the signal to symbol paradigm. A basic assumption behind it is the existence of significant tokens that can be extracted from the signal and which intrinsically characterize the objects. Unfortunately, the true existence of significant and universal tokens is still an open question and after years of investigations and partial successes with tokens like planar or curved algebraic patches, it appears that their generalization for complex shapes is difficult and that with it, it is hard to progress towards the recognition of objects of arbitrary shapes.

To further investigate model-based 3D vision for arbitrary-shaped objects, we opted for a recognition principle that proceeds by geometric registration of 3D shapes and works directly on the 3D coordinates of the object surface as measured by a range finder. An important component of this approach lies in the fact that the method is independent from object geometry assumptions: the representation of objects by sets of 3D points confers the method high shape modeling versatility, a property that permits to describe arbitrary shapes [3].

Geometric registration

The needed geometric registration implies to find a best fit between a reference and test set of 3D data. Recently, an iterative closest point algorithm (ICP) [1] [2] was proposed to solve this problem. The algorithm proceeds iteratively by changing the objects relative poses (position and orientation) until convergence towards a best fit is obtained. Because the full search for optimal registration is computationally costly, we examine in this paper theoretical and practical possibilities to lowering it by use of adequate and fast search methods.

In a previous paper [5], we considered the case of free-form 2D shapes and a simple 3D object. These experiments have since then been extended [6] and the present paper presents results for complex 3D objects.

Recognition configuration

Our investigations refer to a recognition configuration used for classification and pose estimation of 3D industrial objects in automatic

assembly. The purpose of the recognition is to update the virtual representation of the assembly workspace when objects are moved or new objects are introduced in the workspace [4]. Objects are described by 3D data. The test object is described by a single range image whereas reference objects are sets of 3D points obtained by merging several range images. Recognition encompasses comparison of the test with a set of known reference objects.

Costs of recognition

In order to assess the computational costs for recognition, we analyze the most demanding operation which is the search for an optimal match between a test and K references.

Heart of the search is the ICP algorithm which iteratively finds a better match between a test object and a reference object. Computation time associated with it is a function $T_1(M,N)$ of the sizes M and N of the test respectively reference data set. Considering then I iterations which are required to obtain convergence, the total computation time for ICP becomes

$$T_{ICP} = T_1(M, N) \cdot I$$

a) Test objects obtained from range imaging
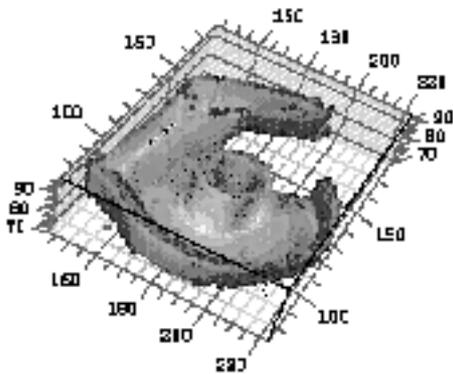
b) Reference object reconstructed from multiple range images

Fig. 1: Objects being matched

Whereas the ICP algorithm is guaranteed to converge towards a best fit, convergence towards the optimal fit heavily depends on the nature of data and more important, on the initial configuration, which is defined as the relative pose of test and reference when starting the algorithm. Searching the optimal fit clearly requires to repeat the ICP algorithm for different initial configurations which number L can possibly be large.

Finally, repeating the whole for each of the K references the resulting computation time for recognition is

$$T_{rec} = T_1 \cdot (M, N) \cdot I \cdot L \cdot K$$

where

M= size of test;

N= size of reference;

I= number of iterations;

L= number of initial configurations;

K= number of references.

We will now successively discuss these factors.

Costs of ICP

Computation of ICP is dominated by finding closest points. The problem is to find, for every element of the test set T, the element of reference data set R which is closest to it. Given a test and reference data set T and R of sizes M and N respectively, the closest point search algorithm finds the M pairs (t,r), t□T, such that r is the element of R closest to t.

For a trivial sequential search, complexity of $T_1(M, N)$ is $O(M \cdot N)$.

Whereas the complexity in M cannot be reduced because the test is used only once, the complexity in N can be reduced by suitable initial work performed once on the reference data set.

Fast search methods

Time complexity of T_1 in N can be reduced by various techniques which include projection methods, grid methods and tree search methods. For inhomogeneously distributed 3D data, the first kind of methods are inadequate because the cells they define are not populated evenly enough, which leads either to overpopulated cells or to a very large number of cells, or even both, depending on the chosen cell size.

With tree search methods, the reference data set is organized into a tree and each element of the test

data set is used in the nearest neighbor search. k-D tree search is a suitable method.

k-D tree search

A k-D tree is a binary tree used to search data with k keys, i.e. it applies to k-dimensional data. Each nonterminal node represents a partitioning of the data set according to one key.

In the classical k-D tree, the discriminator for each node is chosen on the basis of its level in the tree; the discriminator is obtained by cycling through the keys in order. We call this method cyclic k-D tree.

An alternative choice is to select the discriminator at each node in a more flexible way. An interesting choice is to use the key which at each node provides the largest spread in values and to use the median of the key as discriminator value [7]. This leads to a balanced binary tree and theoretically optimal search performance.

3-D tree search algorithms

For object recognition, we considered the two 3-D tree search algorithms as follows.

The algorithm proposed by Zhang [8] uses a cyclic 3-D tree. It also uses an additional parameter D_{max}, which is an a priori distance used to speed up the search. This distance is interpreted as the upper limit of the distance to the nearest neighbor. The reported expected time complexity is $O(N^{2/3})$

The second algorithm uses a balanced 3-D tree as proposed in [7]. The expected time complexity is $O(\log N)$

We compare four methods for computing ICP.

a) Sequential search (Seq), the simplest method, of time complexity $O(N)$

b) Cyclic key 3-D tree search (C3D1) with D_{max}; the value for D_{max} is selected as the minimum distance to the 6 neighbors which lie on the bucket bounds found in an additional and first descent of the 3-D tree.

c) Cyclic key 3-D tree search (C3D2) with D_{max}; the value for D_{max} is estimated from the known distance to the data set of a neighbor of t in T.

d) Balanced 3-D tree search (B3D) , as proposed in [7].

Experiments refer to finding the nearest neighbor pairs from two sets T and R. Euclidean distance is used. Figure 2 reports $T_1(N)$, the execution time as a function of the reference data set size N, for the different tree search methods. We observe

1) Tree search performs better and has lower complexity than sequential search.

2) Among 3-D tree search methods, best performance is obtained with balanced 3-D tree search. We observe logarithmic complexity with N.

Importance of data distribution

Computing time of k-D tree search methods highly depends on the distribution of the data set. To illustrate this, we measured the execution time of an iteration of the nearest neighbor search as a function of the initial distance between test and reference.

Figure 3 shows the strong increase in execution time as the distance between test and reference is

Assessing the computation cost of ICP
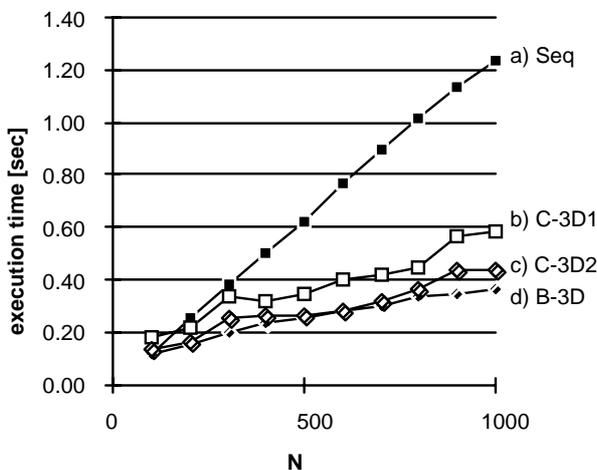
Fig. 2 Computation cost for ICP with 4 search algorithms
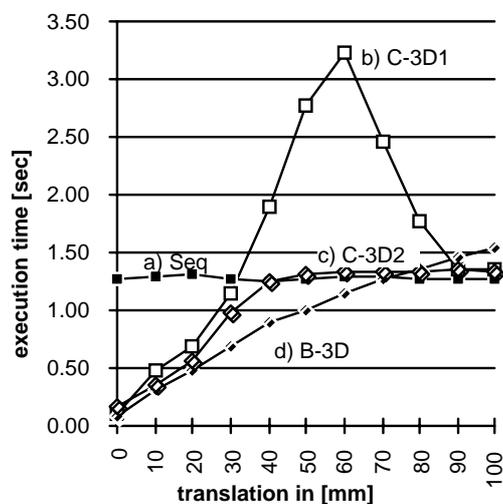
Fig. 3 Computation cost for ICP with initial translation between test and reference objects

increased from 0 to 100 mm. At a distance of about the size of the object, it appears that 3-D tree search is no more better than the sequential search; for much larger distances, sequential search is even preferable to tree search.

This increase of execution time with larger distances is explained by the fact that all distances are very similar. Therefore, the discrimination power of single keys falls short to separate points with similar Euclidean distances.

Number of initial configurations

In the general case, when starting ICP, test and reference have unknown relative poses. As not all poses will bring ICP to converge to the best global match, we start ICP with various different poses we call initial configurations. Later are selected such that a global match exists for at least one of the configurations. Also, we wish their number L to be small in order to keep track with computational costs.

We introduce the notion of zone of optimal convergence to characterize the subspace of the poses which lead to an optimum match. Large zones of optimal convergence are an advantage and permit to lower L.

Initial configuration

An initial configuration is defined according to the following three considerations.

First, given a view axis defined by the camera pointing towards the center of mass of the test, we place the reference behind the test, as shown in Fig. 4. This placement ensures that the test surface not visible from the camera always faces the reference

and also excludes that test surfaces are compared with invisible reference surfaces.

Second, it is important that the two objects are not too far away from each other to avoid unstable point coupling in the first iteration of ICP. This happens when all points of the test are coupled with a single point in the reference and may result in a bad rotation during distance minimization. Practically, we choose the maximal reference radius as distance between both centers of mass.

Third, we select a point of view under which the camera observes the reference. Considering the sphere circumscribing the reference as drawn in Fig. 4, we define the orientation of the camera axis in the reference coordinate system, by three angles, latitude θ, longitude Φ and ω, the rotation angle around the viewing axis itself.

Assessing the zone of optimal convergence

With this definition of an initial configuration, we performed experiments aimed at measuring the size of the zone of optimal convergence. This size can now be expressed as a range of values θ, Φ and ω within which optimal convergence occurs.

The experiments refer to the tape dispenser part of figure 1 measured by range imaging. Obtained results show that the zone of optimum convergence depends only on latitude θ and rotation ω, as follows.

Figure 5 shows the matching error as a function of latitude θ and rotation ω. We observe an obvious zone of optimal convergence, characterized by low

Fig. 4 Initial configuration of test and reference
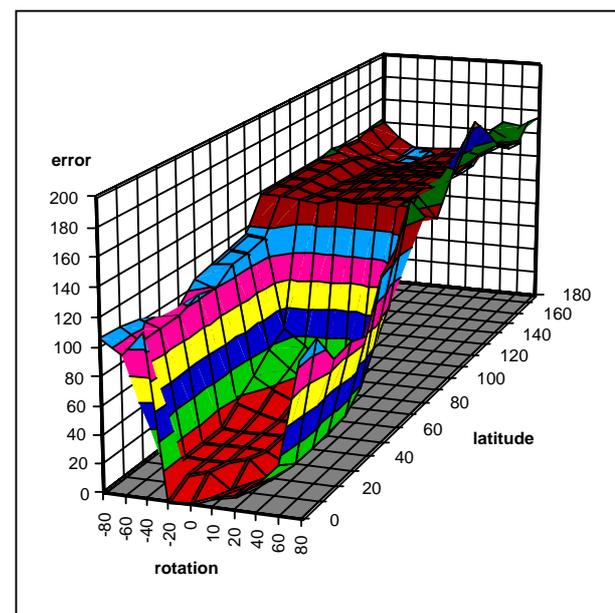
Fig. 5 Zone of optimal converge in θ-Φ space

values of the matching error. It can be described in terms of a range of latitude and rotation which appears to be rather large. Measured values are roughly [‖θ‖<50°, ‖ω‖<30°].

Number of initial configurations

Knowing the average size of the zone of convergence, we can now determine the number L of required initial configurations for a given recognition problem. Considering the worst case, when absolutely no knowledge is available about the objects poses, we can estimate L by enforcing the union of all zones of convergence, displaced at each initial configuration, to cover the whole θ-ω space. While the first range of latitude [‖θ‖<50°] implies a number of viewpoints of about 8 and the second range of rotation [‖ω‖<30°] a number of about 6, we end up with a number of initial configurations of about 50 in our example of a tape dispenser.

Number of iterations

Computation time is linear with the number of iterations of ICP. To keep this number as low as possible, we prune the search by rejecting poor solutions detected after only few iterations I. With a nominal value of 20 for I, and considering a suited distance measure [6], we experienced that this value can be reduced to an average value of about 3 per reference.

Recognition performances

Routinely, we run experiments which perform the recognition of the three different parts from a tape dispenser. Typical parameters are M= 200, N= 500, K=3. Computation runs on a Sun workstation with an iteration time $T_1(M,N)$ of 0.1 s for above values. With no knowledge at all, a full search with three reference objects K=3 will need L♠50 and I♠3, i.e. an average computation time of T_{rec}= 0.1·3·50·3= 45s. With knowledge of the reference object and the initial pose, we have K=1, L=1 and I♠20 and the computation time is T_{rec}= 0.1·20= 2s.

The good recognition performances observed with these objects simply described by sets of 3D points show the feasibility of this geometric matching for recognizing free-form objects.

Conclusions

We analyzed the computational costs of geometric matching by ICP for 3D object recognition and conducted experiments involving 3D objects obtained from range imaging. Comparing experimentally different search methods for speeding up nearest neighbors computation, we found that 3-D tree search methods are efficient provided that attention is given to special cases where inadequate data distributions hinders improvement. We found the balanced 3-D tree search to be experimentally optimal. Considering then the problem of finding the global minimum, we showed that the objects can have zones of optimal convergence which are relatively large, a fact indicating that the number L of required initial configurations can be kept low. Finally, experiments showed that the search can be pruned by rejecting poor solutions detected after only few iterations I of the ICP algorithm. Finally, the good recognition performances observed on parts simply described by set of 3D points show the potential of this approach to recognize free-form objects.

References

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

[2] Feldmar J. & Ayache N., "Rigid and affine registration of smooth surfaces using differential properties", Lecture Notes in Computer Science, Vol. 801, No. 2, 1994, pp. 397-406.

[3] P. Besl, "Triangles as Primary Representation", NSF/ARPA Workshop on 3D Object Representation for Computer Vision, Dec. 1994

[4] Hügli H., Schütz Ch., Baur Ch. & Natonek E., "Knowledge-based 3-D vision system", Proceedings of Information Conference, Module 2, Knowledge based systems, Swiss National Science Foundation, 1994, pp. 46-53.

[5] Schütz Ch. & Hügli H., Recognition of 3-D objects with a closest point matching algorithm", Proceedings of ISPRS Intercommission Workshop "From Pixels to Sequences", Vol. 30, No. 5W1, 1995. pp. 128-133.

[6] Christian Schutz, Heinz Hugli, "Towards the recognition of 3D free-form objects", SPIE Int. Symposium on Intelligent Robots and Advanced Manufacturing, Intelligent Robots and Computer Vision XIV, Philadelphia, Oct. 1995

[7] J.H. Friedman, J.L. Bentley & R.A. Finkel, "An algorithm for finding best matches in

logarithmic expected time", ACM Trans. on Mathematical Software, Vol. 3, No. 3, Sept. 1977, pp 209-226.

[8] Zhang Z., "Iterative point matching for registration of free-form curves and surfaces", International Journal of Computer Vision, Vol. 13, No.2, 1994, pp. 119-152.