MARKER-FREE AUTOMATIC MATCHING OF RANGE DATA

Rui Liu, Gerd Hirzinger

Institute of Robotics and Mechatronics, German Aerospace Center (DLR)
Oberpfaffenhofen, 82234 Wessling, Germany. E-mail: (Rui.Liu, Gerd.Hirzinger)@dlr.de

KEY WORDS: automated data processing, matching, 3D-reconstruction, segmentation, classification, surface registration

ABSTRACT:

Matching of multiple views is often addressed in 3D-model generation and is normally a two-stage process consisting of a coarse and a fine matching stage. Coarse matching, that is the pre-alignment of the surfaces for the complex forms, which can be positioned far away from each other in 3D space, is a difficult problem to solve. Fine matching on the other hand can be performed accurately using either the ICP (iterative closest point) method or the least square surface matching method. Nevertheless, ICP involves an iterative solution which consumes much computing time, and it requires models with a considerable degree of overlap at the starting position. This is because it treats the closest point in the other model as the corresponding point and updates the corresponding relationship in each iterative step. If the models have insufficient overlap, ICP will converge to a false result. Consequently, a good coarse matching is a precondition for a successful ICP. The other matching method - least square surface matching - needs a pre-aligned corresponding relationship between the surfaces of complex objects, exactly the task of the coarse matching process.

This paper presents a novel algorithm to perform coarse matching with an innovative data structure, a "matching tree", which is a combination of an interpretation tree and a bipartite matching graph. The whole systematic process can be divided into three steps: firstly, it performs segmentation of the laser range scan data according to the geometric characteristics; secondly, a coarse matching is conducted to solve the pre-alignment problem; and finally, an efficient fine matching aligns the models accurately. The coarse matching is not affected by the position of the models, because it is generated from a matching tree using invariant relationships from the models themselves. This method is particularly suitable for laser range scan point cloud matching of rooms during the reconstruction of historic buildings, since it requires neither GPS nor wall markings. Experiments have been performed based on reconstructing King Ludwig II 's working room in the real Neuschwanstein Castle and the carnival model of Neuschwanstein Castle.

1. INTRODUCTION

Matching multiple views is an interesting but hard problem for the reconstruction of historic buildings. Because of the demand of high resolution, the processing time of each scan is quite long. Therefore the relative distance between the scans is usually large. Additionally, many different objects appear in each view. Due to these difficulties, the existing matching techniques are insufficient for automatic matching of such scans for building reconstruction. Therefore, in industry, the typical solution to this problem is to use artificial markers.

This paper attempts to solve this problem by introducing a novel data structure, called "matching tree" to pre-align two scans automatically and uses standard fine matching methods subsequently to achieve a high accuracy. Experiments have been performed by reconstructing the working room of King Ludwig II in the real Neuschwanstein Castle and the carnival model of Neuschwanstein Castle. All of the scan data are acquired with the advanced laser scanner of the high tech company Zoller & Froehlich (ZF).

The paper is organized as follows. In section 2, previous work is briefly summarized. Then, section 3, describes the details of the automatic matching process, which is divided into three stages: segmentation, coarse and fine matching. The implementation of the algorithm and the experimental results are shown in this section. After this, the possible improvements in the future are addressed in the conclusion section.

2. RELATED WORK

Automatic Matching without an additional tracking system has always been a hot topic in the 3D modelling field. The research can be categorized into: coarse matching and fine matching, two views matching and multiple matching.

Coarse matching, namely pre-aligning, is usually the precondition of fine matching. It does the global registration task. The preparatory works for the fine matching, such as automatically allocating of the correspondent relationship, aligning views approximately et cetera belong to this step.

For single-object views, one normally utilizes "principal axis transformation" [1]. For multi-object views, the objects' global registration problem must be solved. A possible technique is skeleton based matching [2], which encodes the geometrical and topological information in form of a skeletal and uses graph matching techniques to achieve the destination. Since the construction of skeleton and the matching algorithms of it are sophisticated, we did not choose this method for our application, but used a novel graph-structure, which will be introduced in section 3.

Fine matching is a fairly adult field in science. The most robust and frequently used method is ICP (*Iterative Closes Point*) [3] and numerous variants of it: ICCP (*Iterative Closest Compatible Point*) [4][5][6], ICPIF (*Iterative Closest Points using Invariant Features*) [7] etc. The basic idea of ICP is treating the nearest point in the other view as correspondent point. SVD (Singular Value Decomposition) is used to deduce the transformation-matrix during iterative steps to align the views to each other. The nearest point can in distance field (original ICP), or in diverse feature field (ICPIF), and by weighted correspondent pair or by reducing the search space (ICCP) to accelerate the convergence. The limit of these ICPs is that they only reach local minima. That is, if most of the nearest points just lie in the false direction of the true correspondent points, they will converge to a false result. This problem can be solved either with dynamic programming or by a reasonable pre-alignment.

Adaptive least square matching [8] is an effective method for matching of 3D surface patches. However, it is not suitable for multi-object scenes, because it does not deal with the automatic correspondence problem, but needs an initial approximation.

Multi-view matching can be solved either incrementally or simultaneously. This is not the main focus of this paper. The interested reader is referred to the article of Cunnington and Stoddart [9], who gave a comparison of three n-view point set registration algorithms.

3. MAIN METHOD

In this section, the proposed method is described in detail. The whole process can be divided into three steps: firstly, it performs segmentation of the laser range scan data according to the geometric characteristics; secondly, a coarse matching is conducted to solve the pre-alignment problem; and finally, an efficient fine matching aligns the models accurately.

3.1 Segmentation

By the characteristics of the surfaces, the model in one view is segmented into diverse objects. Here, the difference between normal vectors between adjacent points is treated as segmenting criteria.

As illustrated in Figure 1, if the difference between the normal vectors of p and q is fairly large, then the boundary between two objects will be set through here.

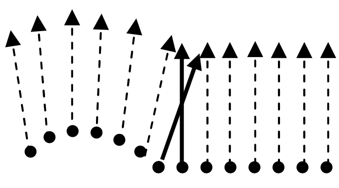


Figure 1: criterion for segmentation

The estimation of the normal vector is done by least squares fitting of a tangent plane through the point neighborhood. For the 2.5 D range image, the information from the points can be stored in a matrix, so the access time to the neighborhood is O(1).

The whole matrix will be treated as a graph. Each point in the matrix is like the vertex of the graph, and an edge is constructed if the adjacent points fit the segmenting criteria. By BFS (breadth first search), the whole graph is segmented. The runtime of BFS is O(m + n), in which m is the number of the edges and n is the number of the vertices. Because the edges exist only between adjacent points, the maximal number of edges on a vertex is 8. So the runtime of segmentation is linear, i.e. O(n).

Because the differences between normal vectors are identical in diverse views, segmentation of each view should be the same. After a successful segmentation, the invariant characteristics will be taken from the segments.

We define a form-descriptor as a two-attribute vector consisting of the sine value of the angle between the normal vectors and the radius of the surface curvature.

According to the form-descriptor-vector, a list of correspondent segment-pairs is established with a certain tolerance by calculating the normalized Euclidean distance.

For the experimental reconstruction of King Ludwig II's working room, the number of segments in each scan is about 100. The segmentation and computing time of the values of form-descriptor requires 10-20s.

3.2 Coarse Matching

In this step, the goal is to find the best allocation of the correspondent relationship between segments of different views automatically. We introduce a novel data-structure to achieve it: the "matching tree".

The matching tree is a combination of an interpretation tree and bipartite matching graph. The basic idea is to gain the benefits and overcome the shortcomings of both structures.

An interpretation tree represents the complete search space for a problem [10]. For a 2-view matching problem, if there are two segments in one view and three segments in the other view, the correspondent interpretation tree will be as illustrated in Figure 2. That is, the two layers represent the two segments of the first view, and the four children of each node represent the four matching possibilities, either one of the three segments of the other view, or none of them. The edges of the tree are interpreted as the matching quality. Each path from root to leaf represents a solution of the matching problem. Getting the best matching between two views means finding the optimal path from root to leaf.

The advantage of an interpretation tree is that it represents all of the possibilities for matching. But it has two essential shortcomings: firstly, it doesn't represent the space relationship between the segments directly. That means the maximally weighted path can be a false matching; the second shortcoming is the runtime problem. It is NP-complete, that is, if there are n segments in one view and m segments in the other view, the runtime is O(n^{m+1}). Many people have resolved this problem by backtracking, that is, cutting off the sub-trees from the main tree by setting some boundary criteria.

The advantage of the bipartite matching graph is that it represents the correspondent relationship between the pairs. But it does not deliver the 3D topological information between the segments.

The structure of the matching tree and its relationship between the interpretation tree and matching graph are shown in figure 2.

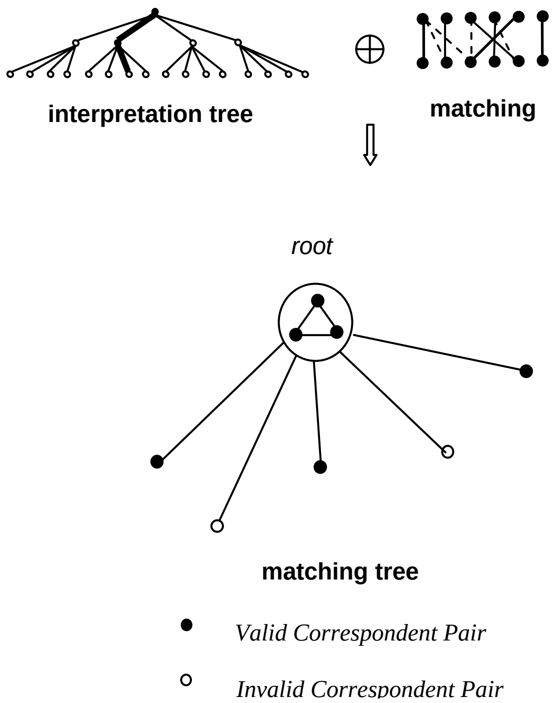


Figure 2: Relationship of interpretation tree, matching and matching tree

The matching tree is a tree specially designed for matching. Each node of the tree represents a correspondent pair. The black color means that this correspondent relationship is valid in current matching, while white means invalid. The root of the matching tree consists of three matched basis-correspondent pairs. The relative distances between the three base nodes in two views are also “matched” with a certain error tolerance. To assure the stability of the tree structure, the three nodes in the root should not be collinear. This will be represented as a stable triangle in the root of the tree. Thus the validity of the other correspondent pairs can be determined by their directed distances to the root. This will be illustrated in Figure 3. To avoid the scan point density changing in two views, we take the form centroid of the segment to represent the segment.

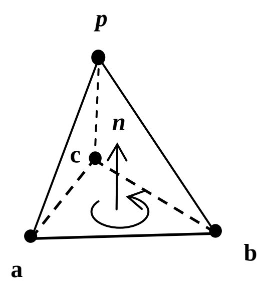


Figure 3: Validation of the correspondent pair

Δabc is the triangle in the root. *n* is the cross product of $\overrightarrow{ab}$ and $\overrightarrow{ac}$. p is the considered correspondent pair. $\overrightarrow{ap}$, $\overrightarrow{bp}$, $\overrightarrow{cp}$ are three directed distances, the direction of which can be calculated by the scalar product with *n*.

For example, (a, a’), (b, b’), (c, c’) are three correspondent pairs in the root. (a, b, c) is from one view, and (a’, b’ ,c’) is from the other. As illustrated in Figure 4, p and p’ has the same directed distances to the three pairs of the basis, then (p, p’) is valid. In contrast, (q, q’) is invalid.

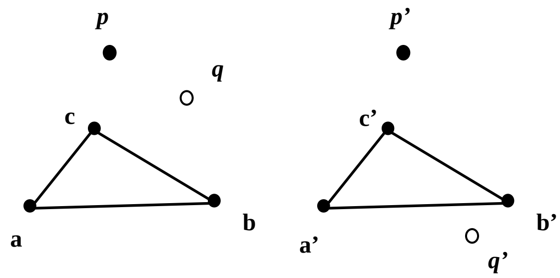


Figure 4: Example of the valid and invalid correspondent pair

A matching tree is a tree, and also a matching. That means that the black nodes in one tree are disjunctive to each other, because none of the segment in one view can correspond to more than one segment in the other view. The black node is like the solid line in the matching graph, and the white one is like the dotted line.

We define an income and cost function for the matching tree. The income is the whole area of the matched segments, and the cost rate is the relative matching error. Therefore, the weight of the matching tree can be calculated as the difference between the incomes and costs. The choice of the root results in different weights of the tree. Our goal is to find the tree with maximal weight. If there are n correspondent pairs, the runtime is maximal O(n^4). That is, the different combinations of the root is maximal O(n^3), and the validation of the other pairs in each iteration costs linear time. By pre-cutting and wise strategy for the selection of the root, the process can be accelerated significantly.

After defining the segment correspondence correctly, one can calculate the transformation between the views in different ways.

One way [10] is: translate the views to the center of the correspondent pairs, and calculate the rotation matrix of the second view by singular value decomposition. The algorithm is as follows, with run time of O(*n*).

Input: n correspondent pairs (p, q) with correspondent weight w.
Output: rotation matrix R.

1. Compute the covariance matrix K:

$$K = \begin{pmatrix} K_{00} & K_{01} & K_{02} \\ K_{10} & K_{11} & K_{12} \\ K_{20} & K_{21} & K_{22} \end{pmatrix},$$

with $K_{ij} = \sum_{m=1}^{n} p_{mi} \cdot q_{mj} \cdot w_m$

2. Compute the Singular Value Decomposition

$$K = UDV^T$$

3. Compute the rotation matrix R:

$$R = V \begin{pmatrix} 1 & & \\ & 1 & \\ & & \delta \end{pmatrix} U^T,$$

and set δ = 1 or–1, whatever is closest to $\det(VU^T)$.

Figure 5 shows two views before and after the coarse matching during reconstruction of a wooden scaled carnival model of Neuschwanstein Castle.

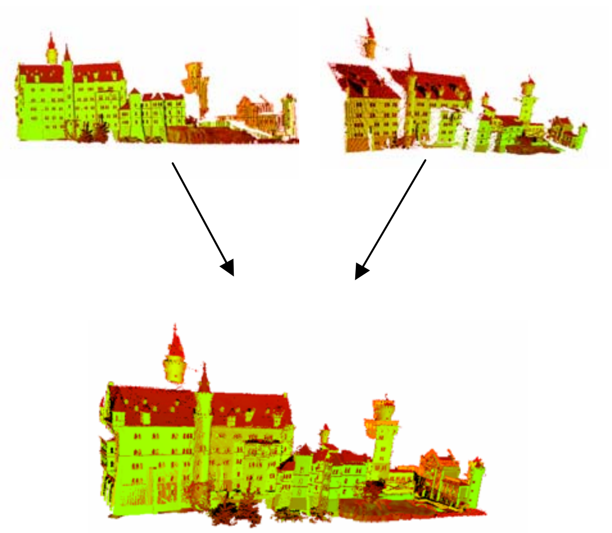
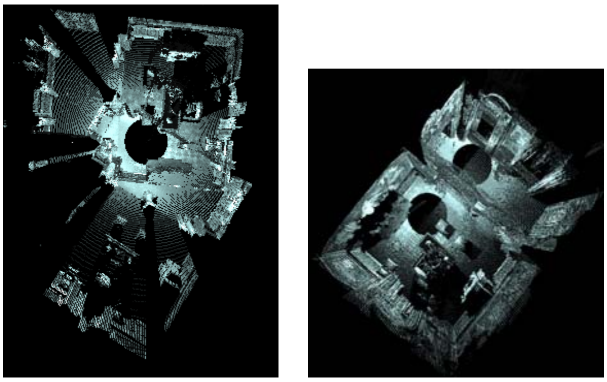


Figure 5: Coarse matching: the carnival model of Neuschwanstein Castle

During reconstruction of King Ludwig II's working room, coarse matching for two views runs within about 6s. Figure 6 shows two views before and after the coarse matching.



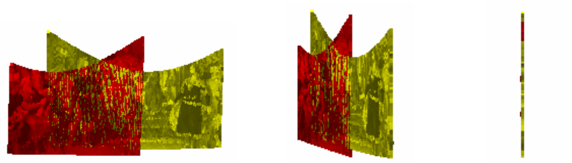
(a) before (b) after

Figure 6: Coarse matching: King's working room

3.3 Fine Matching

The correct correspondent relationships between segments are decided in the first two stages. We can generate an arbitrary number of control-point-pairs by projecting the sampling points of one segment to its correspondent segment. And by the use of iterative actions, the result will be refined.

Figure 6 shows a piece of the wall from the 2-view matching result of the king's working room. By experiment the matching accuracy is within 2mm, which is restricted by the accuracy of the laser scanner. It can be improved much more if both of the views exactly represent the same real models.

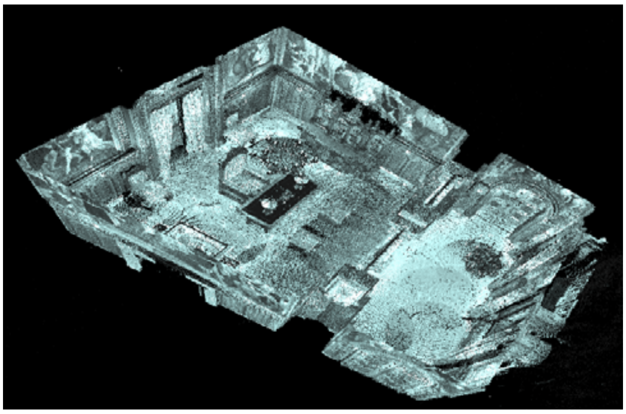


(a) (b) (c)

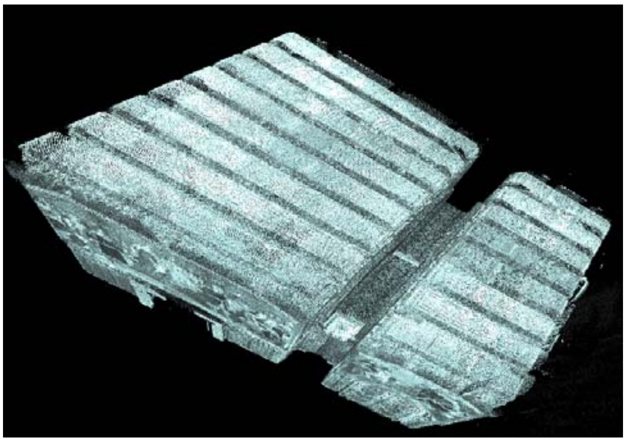
Figure 7: A part of a wall after fine matching in King Ludwig II 's working room in Neuschwanstein Castle : (a), (b) and (c) are from different viewing angles: 90º in (a), about 45º in (b) and 0º in (c).

Our fine matching method is a variant of ICP. Of course, another method could be used to do the fine matching. As described in the section 2, there are many excellent methods in this field, such as accelerated ICP and adaptive least squares etc. The research in this field is very fruitful.

Multiple matching is not the research focus of this paper. We computed the result from the 15 views of Ludwig II's working room incrementally. For better visualisation, we illustrate a lower part of the room, too.



(a) lower part



(b) whole

Figure 8: King Ludwig II's working room: matching result from 15 views

4. CONCLUSION

In this paper, an approach to marker free automatic matching for historic building reconstruction is presented. The algorithm consists of three steps: segmentation, coarse matching and fine matching.

In the second step, a "matching tree" is introduced to find the best pre-alignment between views. But we think that it is only a prototype. There is considerable room for improvement, both in the structure itself and in the algorithm for finding the best matching.

As the runtime of the coarse matching is decided by the number of correspondent segments, the advancing of the segmentation-technique and refining of the form-descriptor is also important work for the future.

The fine matching result should be adjusted by multi-view matching in the future. Furthermore, multi-view matching can perhaps be integrated in the matching tree algorithms, and the three steps, coarse- fine- and multiple-matching, can be integrated into one.

5. ACKNOWLEDGMENTS

Thanks to Johann Heindl, Bernhard Strackenbrock, Dr. Heiko Hirschmueller, Michael Suppa, Tim Bodenmueller and Wolfgang Sepp for the useful discussions and delivery of related data.

Further thanks to the colleagues from Co. Zoller und Froehlich: Dr. Christoph Froehlich, Franz Haertl, Tomas Abmayer and Inga Wolf for their support for our work.

6. REFERENCES

[1] Michael Mayr
Matching zweier Punktwolken zur intraoperativen Patienten-registrierung p. 38
Diplomarbeit für Institut für Robotik und Mechatronik
DLR, Oberpfaffenhofen April 2004

[2] H. Sundar, D. Silver, N.Gagvani, S.Dickinson
Skeleton Based Shape Matching and Retrieval
International Conference on Shape Modeling and Applications 2003, May 12 - 15, 2003, Seoul , Korea. p. 130

[3] Besl, P.J., and McKay, N.D., 1992.
A method for registration of 3D shapes.
IEEE Pattern Analysis and Machine Intelligence, 14 (2)

[4] Godin, G., Rioux, M., Baribeau, R., 1994.
Three-dimensional registration using range and intensity information
SPIE vol.2350, Videometrics III, pp. 279-290.

[5] Godin, G., Boulanger, P., 1995.
Range image registration through viewpoint invariant computation of curvature.
IAPRS, 30 (5/W1), pp.170-175

[6] Godin, G., Laurendeau, D., Bergevin, R., 2001.
A method for the registration of attributed range images.
Int. Conf. on 3D Imaging and Modeling,
Quebec, May 28 – June 1, pp. 179-186.

[7] Sharp, G.C., Lee, S.W., Wehe, D.K., 2002.
ICP registration using invariant features. IEEE Pattern Analysis and Machine Intelligence, 24 (1), pp. 90-102.

[8] Armin Gruen, Devrim Akca
LEAST SQUARES 3D SURFACE MATCHING
"1.PanoramicPhotogrammetry Workshop", 19-22 Feb.2004, Dresden, Germany
http://www.tu-dresden.de/fghgipf/photo/PanoramicPhotogrammetryWorkshop2004/Proceedings.htm

[9] S. J. Cunnington and A. J. Stoddart
N-View Point Set Registration: A Comparison
Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, GU2 5XH, UK
http://www.ee.surrey.ac.uk/Research/VSSP/3DVision/virtuous/Publications/cunnington-bmvc99.pdf

[10] Emanuele Trucco, Alessandro Verri:
Introductory Techniques for 3-D Computer Vision
Prentice Hall, 1998