Registration of range images based on segmented data *

Bojan Kverh and Aleš Leonardis

Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25
Ljubljana, Slovenia
tel: (386)-61-1768-381
e-mail: {bojank,alesl}@razor.fer.uni-lj.si

Abstract. We present a new method for registration of range images, which is based on the results we obtain from the segmentation process. To obtain the first set of points needed for registration, we use set of points from first range image. The novelty is how we obtain the second set of points. To obtain the second set we project the first set of points onto geometric parametric models obtained in the second range image. Then we compute the transformation between the two sets of points. The results have shown a significant improvement in precision of the registration in comparison with traditional approach.

Keywords: registration, segmentation, range images.

1 Introduction and motivation

Range image registration is considered an important part of any vision system that requires data acquisition of parts that are not visible from any single viewpoint. There is no 3D sensor system available on the market that would obtain 3D data of the complete surface by a single scan. To extract complete information about the surface of an object we need to scan it from several viewpoints and then merge the data together. This requires that we compute the transformation between data sets obtained from different viewpoints. A rough approximation of the transformation can be obtained by involvement of a human operator, from positioning devices, e.g., a turntable, or automatically by detecting corresponding features in both sets. For some tasks, like reverse engineering, the accuracy of this approximate transformation has to be further improved by one of the refinement methods. In this paper we present a new refinement method which is based on segmented data, i.e., descriptions obtained by Recover-and-Select paradigm. Refinement methods usually compute the transformation either between two sets of points [1, 10] or between a set of points and a set of some surface

* This work was supported by a grant from the Ministry of Science and Technology of Republic of Slovenia (Project J2-0414-1539-98).

elements, e.g., triangles [9]. For each point from the first set, these methods find a correspondent point or surface element in the second set. Transformation between two sets of corresponding points is then computed by methods usually based on least squares. However, due to scanner sampling it is impossible to find the exact corresponding point or surface element for each point in the first set. This situation is illustrated in Fig. 1, which clearly shows that if the refinement is performed in this way, small errors will affect the computed transformation. Therefore we propose a novel approach which alleviates this problem.

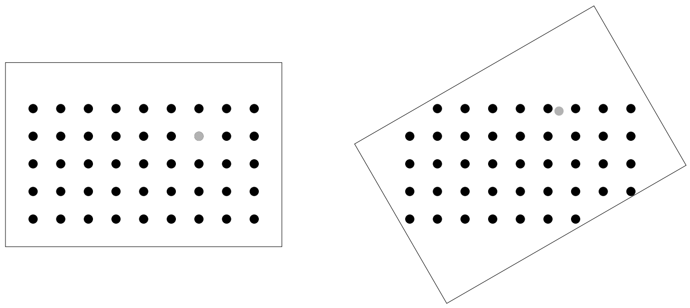

Fig. 1. An example of two corresponding points. Gray point in the right image is not captured by the scanner in the second view.

Registration of all range images is achieved by repeatedly registering pairs of range images. In each step, 3D points from both range images are first segmented into descriptions by the Recover-and-Select paradigm [5, 3, 4]. This gives us geometric parametric models which approximate each segmented region. Let us assume that we have an initial approximation of the transformation. Then, instead of computing the transformation between the two sets of 3D points, we can use one set of points and their correspondent *projections* to geometric parametric models that were obtained for the other set of points. This process can be iterated until the transformation error reaches some predefined value. In this way we cancel the effect of scanner sampling (since the projection of a point to a geometric model really corresponds with the point). The point that corresponds to a particular point from the first set, doesn't need to be captured by the scanner since it is computed from the segmented data.

The rest of the paper is organized as follows: in section 2 we introduce descriptions obtained by the Recover-and-Select segmentation process, in section 3 we present the Projection Iterative Closest Point (PICP) algorithm, geometric parametric models and projections of points onto the models are presented in section 4, while sections 5 and 6 contain experimental results and discussion, respectively.

2 Descriptions

Our method to refine the transformation between two partially overlapping range images assumes that both of them are segmented with Recover-and-Select paradigm [5, 3, 4]. The set of 3D points is segmented into regions. Each region though has its own geometric parametric model that fits best data in the region. The region and its model form a description. To express model M quality of fit to the region R, we use Eq. 1, known as description error,

$$Err = \frac{1}{|R|} \sum_{\mathbf{x} \in R} d(\mathbf{x}, M). \tag{1}$$

Function $d(\mathbf{x}, M)$ calculates distance of $\mathbf{x}$ to the model M.

The Recover-and-Select paradigm can use the following geometric models: planes, superquadrics, spheres, cylinders, cones, and tori. An example of segmented data is shown in Fig. 2. Different gray levels denote different regions.

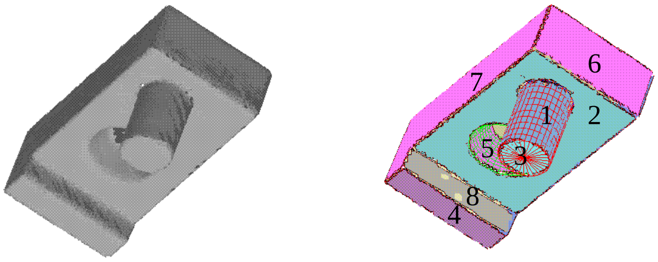

Fig. 2. Generated intensity image of an object and its range image segmentation. An index of description is displayed on each region.

Geometric parametric models in descriptions shown in Fig. 2 are planes (descriptions 2, 3, 4, 6, 7, 8, and 9), cylinder (description 1) and cone (description 5). In our transformation refinement method we use these models to compute projections of points to their surfaces.

3 PICP algorithm

We call our new transformation refinement method Projected Iterative Closest Point (PICP) algorithm, because it is similar to ICP algorithm [1]. ICP algorithm in its simplest implementation computes the transformation between two sets of points X and Y from different coordinate systems. The correspondent point to a point $\mathbf{x} \in X$ is $\mathbf{y}$ which satisfies the condition 2

$$\mathbf{y} = \arg\min_{\mathbf{t} \in Y} d(\mathbf{x}, \mathbf{y}). \tag{2}$$

Function d represents a measure of distance between two points. ICP algorithm then computes the transformation that minimizes the sum of distances between transformed points from set X and their corresponding points from the set Y. Several iterations of ICP algorithm can be performed to obtain more accurate transformation.

PICP algorithm was developed to avoid the sampling effect. Using information obtained by segmentation stage and the initial approximation of the transformation between viewpoints from which the point sets have been scanned, we are able to determine pairs of corresponding descriptions. Description D_1 from the first set of points, consisting of region R_1 and model M_1, corresponds to description D_2 from the second set of points, consisting of region R_2 and model M_2 if conditions 3, 4 and 5 are satisfied.

$$\frac{1}{|R_2|} \sum_{\mathbf{x} \in T(R_2)} d(\mathbf{x}, M_1) < d_{tr}, \tag{3}$$

$$\frac{1}{|R_1|} \sum_{\mathbf{y} \in R_1} d(\mathbf{y}, M_2) < d_{tr}, \tag{4}$$

$$\exists \mathbf{x} \in R_1 \wedge \exists \mathbf{y} \in T(R_2) : d(\mathbf{x}, \mathbf{y}) < d_{tr}. \tag{5}$$

T denotes initial approximation of the transformation, function d denotes a measure of distance between two points, or between a point and geometrical parametrical model, while d_{tr} represents a threshold value. For each pair of corresponding descriptions conditions 3 and 4 assure that points from the region of one description are close enough to the geometric model of the other description after the initial approximation of transformation has been applied to one of the sets of points. Condition 5 requires that points from regions of both descriptions are close enough, otherwise we have no information if descriptions really correspond to each other. In Fig. 3 two different segmented range images are shown.

When the pairs of corresponding descriptions are identified, PICP algorithm starts to construct sets X and Y. It starts with $X = Y = \emptyset$. For each pair of corresponding descriptions it adds points from the region of description from the first range image to set X and projections of the same points to geometric model of description from the second range image to set Y. After all pairs of corresponding descriptions have been taken into consideration, sets X and Y are completed. PICP algorithm then uses a method based on quaternions [8] to compute the parameters of the transformation.

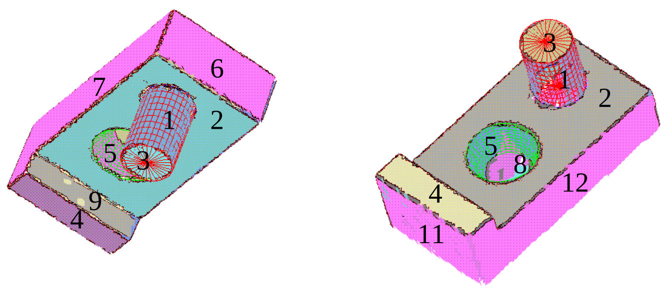

Fig. 3. Two segmented range images. Corresponding descriptions have same indices in both images.

4 Geometric parametric models and projections of points to them

In this section we present geometric parametric models used in the Recover-and-Select based segmentation stage. Also, we show how to compute coordinates of a point that has been projected onto a particular model.

4.1 Projection of a point to a plane

Planes are represented by parameters a, b, c, d and consist of points $\mathbf{t} = (x, y, z)$ that satisfy equation $ax + by + cz + d = 0$. The projection ($\mathbf{t_{proj}}$) of a point to a plane can be computed as shown in Eq. 6.

$$\begin{aligned} \mathbf{t_{proj}} &= \mathbf{t} - d_1 \mathbf{n}, \\ \text{where} \qquad d_1 &= \mathbf{nt} + d, \\ \mathbf{n} &= (a, b, c) \qquad \text{and} \\ a^2 + b^2 + c^2 &= 1. \end{aligned} \tag{6}$$

4.2 Projection of a point to a sphere

Spheres are represented by parameters a, b, c, r and consist of points $\mathbf{t} = (x, y, z)$ that satisfy equation $(x-a)^2 + (y-b)^2 + (z-c)^2 - r^2 = 0$. The projection ($\mathbf{t_{proj}}$) of a point to a sphere can be computed as shown in Eq. 7.

$$\begin{aligned} \mathbf{t_{proj}} &= \mathbf{t} - d_2(\mathbf{t} - \mathbf{c}), \\ \text{where} \qquad d_2 &= \frac{d_1 - r}{d_1}, \\ \mathbf{c} &= (a, b, c) \qquad \text{and} \\ d_2 &= ||\mathbf{t} - \mathbf{c}|| \,. \end{aligned} \tag{7}$$

4.3 Projection of a point to a cylinder

Cylinders are represented by parameters $\rho, \phi, \theta, \alpha, k$ [6, 7]. Parametrisation of a cylinder is shown in Fig. 4. Values ϕ and θ are used to compute vector $\mathbf{n}$ ($\mathbf{n} = (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)$), which is parallel to a normal vector of a tangent plane to a cylinder surface and orthogonal to a vector on the cylinder axis ($\mathbf{a}$). The projection ($\mathbf{t_{proj}}$) of a point to a cylinder can be computed as shown in Eq. 8.

$$\begin{aligned} \mathbf{t_{proj}} &= \mathbf{t} - (\|\mathbf{t} - \mathbf{v}\| - 1/k)\mathbf{r}, \\ \text{where} \qquad \mathbf{r} &= \frac{\mathbf{t} - \mathbf{v}}{\|\mathbf{t} - \mathbf{v}\|} \qquad \text{and} \\ \mathbf{v} &= (\rho + 1/k)\mathbf{n} + ((\mathbf{t} - (\rho + 1/k)\mathbf{n}) \cdot \mathbf{a})\mathbf{a}. \end{aligned} \tag{8}$$

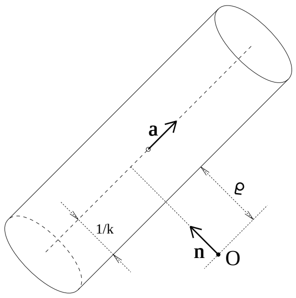

Fig. 4. Parametrisation of a cylinder

4.4 Projection of a point to a cone

Parametrisation of a cone is similar to the parametrisation of a cylinder. A cone has parameters $\rho, \phi, \theta, \sigma, \tau, k$[6, 7]. The parametrisation of a cone is shown on Fig. 5. The difference is that here vectors $\mathbf{n}$ and $\mathbf{a}$ are not necessarily orthogonal. Vector $\mathbf{n}$ is computed in the same way as for cylinder, while $\mathbf{a}$ has components $(\cos\sigma\sin\tau, \sin\sigma\sin\tau, \cos\tau)$[6, 7]. The projection ($\mathbf{t_{proj}}$) of a point to a cone can be computed as shown in Eq. 9.

$$\begin{aligned} \mathbf{t_{proj}} &= \mathbf{t} - (\|\mathbf{t} - (\rho + 1/k)\mathbf{n} - z\mathbf{a}\| - d)\mathbf{r} \\ \text{where} \qquad z &= \mathbf{a} \cdot (\mathbf{t} - (\rho + 1/k)\mathbf{n}), \\ \alpha &= \arccos(\mathbf{a} \cdot \mathbf{n}), \\ d &= \frac{1}{k\sin\alpha} - z\tan\alpha \qquad \text{and} \\ \mathbf{r} &= \frac{\mathbf{t} - (\rho + 1/k)\mathbf{n} - z\mathbf{a}}{\|\mathbf{t} - (\rho + 1/k)\mathbf{n} - z\mathbf{a}\|}. \end{aligned} \tag{9}$$

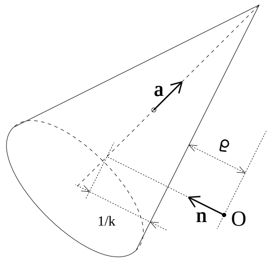

Fig. 5. Parametrisation of a cone

4.5 Projection of a point to a torus

A torus has parameters $\rho, \phi, \theta, \sigma, \tau, k, s$[6, 7]. Parameters $\phi, \theta, \sigma, \tau$ denote vectors $\mathbf{n}$ and $\mathbf{a}$, while both radii of a torus can be computed with $\frac{1}{k}$ and $\frac{1}{s}$. Parametrisation of a torus is shown on a torus cross-section in Fig. 6. The projection ($\mathbf{t_{proj}}$) of a point to a torus can be computed as shown in Eq. 10.

$$\mathbf{t_{proj}} = \mathbf{t} - (||\mathbf{t} - \mathbf{c} - \mathbf{v}|| - 1/k)\mathbf{r}, \tag{10}$$

$$\text{where} \quad \mathbf{c} = (\rho + 1/s)\mathbf{n} - |1/s - 1/k| \cos(\mathbf{a} \cdot \mathbf{n})\mathbf{a},$$

$$\mathbf{v} = |1/s - 1/k| \sin(\mathbf{a} \cdot \mathbf{n}) \frac{(\mathbf{a} \times (\mathbf{t} - \mathbf{c})) \times \mathbf{a}}{||(\mathbf{a} \times (\mathbf{t} - \mathbf{c})) \times \mathbf{a}||} \quad \text{and}$$

$$\mathbf{r} = \frac{\mathbf{t} - \mathbf{c} - \mathbf{v}}{||\mathbf{t} - \mathbf{c} - \mathbf{v}||}.$$

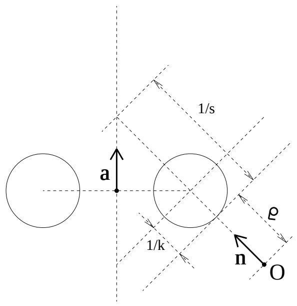

Fig. 6. Parametrisation of a torus

5 Experimental results

We used PICP algorithm in a reverse engineering application [2]. We scanned an object from four viewpoints and obtained four range images (RI_1, RI_2, RI_3, RI_4 shown in Fig. 7). These images were then segmented. After that, we registered images RI_1 and RI_2 and merged them together. At first step we registered the first and the second range image and merged them. In this way we obtained range image RI_{12}. Then we registered images RI_{12} and RI_3 and merged them to obtain the range image RI_{123}. Finally we registered images RI_{123} and RI_4 and merged them to obtain the final image RI_{1234}. When merging two segmented range images to obtain a new range image, we do the following:

1. if description D_1 from the first segmented range image, does not have a corresponding description in the second segmented range image, we insert D_1 into the new range image,
2. if description D_2 from second segmented range image, does not have a corresponding description in the first segmented range image, we insert D_2 into the new range image,
3. if descriptions D_1 and D_2 from different range images correspond and if R_1 is region of D_1 and R_2 is region of D_2, we form a new region $R = R_1 \bigcup R_2$ and fit new model M to region R. We insert description containing R and M into the new range image.

Let us think for the moment of the segmented range image which was generated by merging together two other range images. If some description D was formed by merging together two corresponding descriptions D_1 and D_2, it is obvoius that if the registration stage did not find a good transformation between the two range images, description error (Eq. 1) of D is significantly greater than description error of D_1 and D_2. Similarly, if the transformation found by the registration stage is accurate, description error of D should be similar to description errors of D_1 and D_2. Table 5 shows description errors for three descriptions shown in Fig. 2. Descriptions that were merged from two corresponding descriptions are not much worse in terms of description errors, so it can be concluded that PICP gives satisfactory results.

Description	RI_1	RI_2	RI_{12}	RI_3	RI_{123}	RI_4	RI_{1234}
cylinder (index 1 in Fig. 2)	0.091	0.063	0.1263	0.1112	0.1754	0.1366	0.1986
cone (index 5 in Fig. 2)	0.1284	/	0.1284	0.1215	0.1285	0.1423	0.1746
planar (index 2 in Fig. 2)	0.0950	/	0.0950	0.0980	0.1038	0.1037	0.1056

Table 1. Comparison of description errors between scanned and merged range images

We compared PICP algorithm with ICP in terms of description errors after merging and it turned out that after the same number of steps, PICP always achieves better precision. Descriptions obtained when ICP was used, had errors

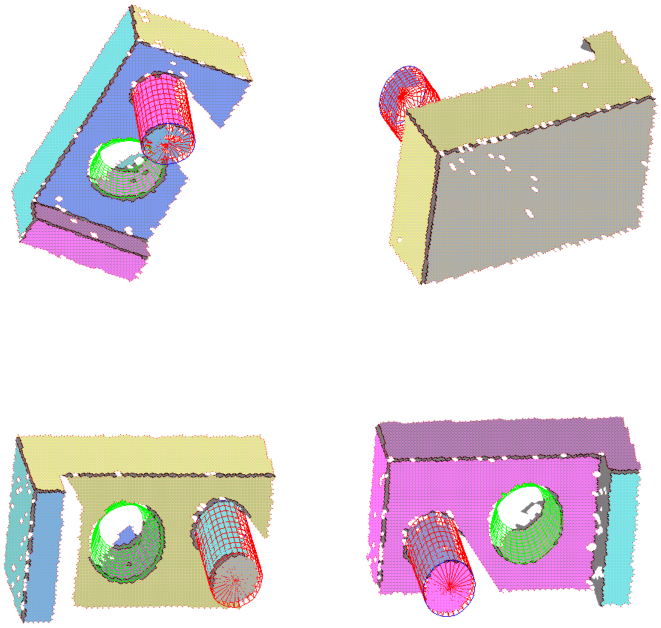

Fig. 7. Four segmented range images

that were 40% to 300% greater than descriptions obtained when we used PICP. PICP has also very good time performance since its complexity is only $O(n)$ [2].

6 Discussion

We presented a new method, PICP algorithm, for registration of range images. It is used to refine the initial approximation of the transformation between range images that were previously segmented. PICP's performance is good in terms of accuracy and required computational time. PICP has some drawbacks as well. The most significant is, that it cannot be used for objects with arbitrary free-form surfaces since our approach requires a segmentation into appropriate models which is currently limited to planes, spheres, cylinders, cones, and tori. Our future work will be directed towards including other types of models. We also intend to compare PICP algorithm with other known refinement algorithms ([9], [10]).

Acknowledgements

We thank Tomaš Pajdla for discussions about our method and Aleš Jaklič for providing a critical overview of our paper.

References

1. P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on PAMI*, 14(2):239–256, February 1992.
2. B. Kverh. Segmentation and registration of range images as a tool for modelling objects. Msc. thesis, Faculty for Computer and Information Science, University of Ljubljana, 1998.
3. B. Kverh, A. Jaklič, A. Leonardis, and F. Solina. Recover-and-select paradigm on triangulated data. In *Proceedings of the Sixth Electrotechnical and Computer Science Conference ERK'97*, volume B, pages 241–244, September 1997.
4. B. Kverh and A. Leonardis. Using recover-and-select paradigm for simultaneous recovery of planes, second order surfaces and spheres from triangulated data. In *Proceedings of the Computer Vision Winter Workshop CVWW'98*, pages 26–37, February 1998.
5. A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *International Journal of Computer Vision*, 14:253–277, 1995.
6. G. Lukács, A. D. Marshall, and R. R. Martin. Geometric least-squares fitting of spheres, cylinders, cones and tori. In R. R. Martin and T. V. ady, editors, *RECCAD Deliverable Documents 2 and 3 Copernicus Project No. 1068 Part I, Report on Basic Geometry and Geometry Model Creation with further contributions on Data Acquisition and advanced material on Merging and Applications*, GML 1997/5, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 1997.
7. G. Lukács, A. D. Marshall, and R. R. Martin. Faithful least-squares fitting of spheres, cylinders, cones and tori for reliable segmentation. In *ECCV98 - Proceedings of the 5th European Conference on Computer Vision*, volume I, pages 671–686, June 1998.
8. A. D. Marshall and R. R. Martin. *Computer Vision, Models and Inspection*, volume 4 of *World Scientific Series in Robotics and Automated Systems*. World Scientific, September 1991.
9. T. Masuda, K. Sakaue, and N. Yokoya. Registration and integration of multiple range images for 3-D model construction. In *Proceedings of the 13th International Conference on Pattern Recognition*, pages 879–883, August 1996.
10. T. Pajdla, V. Hlaváč, and D. Večerka. Semi-automatic building of complete 3D models from range images. In R. R. Martin and T. Varady, editors, *RECCAD Deliverable Documents 2 and 3 Copernicus Project No. 1068 Part II, Report on Basic Geometry and Geometry Model Creation with further contributions on Data Acquisition and advanced material on Merging and Applications*, GML 1997/5, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 1997.