

# Coarse orientation of terrestrial laser scans in urban environments

C. Brenner\*, C. Dold, N. Ripperda

*Institute of Cartography and Geoinformatics, Gottfried Wilhelm Leibniz  
Universität Hannover, D-30167 Hannover, Germany*

---

## Abstract

The use of terrestrial laser scanners is becoming increasingly popular. For the acquisition of larger scenes, it is usually necessary to align all scans to a common reference frame. While there are methods using direct measurement of the orientation, due to simplicity and costs, mostly artificial targets are used. This works reliably, but usually adds a substantial amount of time to the acquisition process. Methods to align scans using the scan data itself have been known for a long time, however, being iterative, they need good initial values.

In this paper, we investigate two different methods targeted at the determination of suitable initial values. The first one is based on a symbolic approach, using corresponding features to compute the orientation. The second one is based on an iterative alignment scheme originally proposed in the robotics domain. To assess the performance of both methods, a set of 20 scans has been acquired systematically along a trajectory in a downtown area. Reference orientations were obtained by a standard procedure using artificial targets. We present the results of both methods regarding convergence and accuracy, and compare their performance.

*Key words:* Terrestrial laser scanning; Orientation; Registration; Coarse alignment; Initial values

---

## 1 Introduction

Orientation of measurement data in relative or absolute frames is a fundamental topic of photogrammetry since its very beginnings. Instead of orientation,

---

\* Corresponding author.

*Email address:* [claus.brenner@ikg.uni-hannover.de](mailto:claus.brenner@ikg.uni-hannover.de) (C. Brenner).

*URL:* [www.ikg.uni-hannover.de](http://www.ikg.uni-hannover.de) (C. Brenner).

the term *registration* has also been used, especially for the alignment of 3D scan data, putting emphasis on the role of the data itself rather than additional measurements.

### 1.1 *Orientation of terrestrial laser scanner data*

In terrestrial laser scanning orientation, two situations can be distinguished. In the dynamic case, the scanner is moving and there is no stable exterior orientation across all scan points of one frame. This is the standard case in mobile mapping, where external sensors, usually GPS and an inertial measurement unit (IMU), are used for direct georeferencing (Talaya et al., 2004). In contrast, the static case is the usual method in surveying, where the instrument is not moved during the capture of one entire frame. Still, the orientation can be measured using additional sensors (Asai et al., 2005). However, for reasons of cost-effectiveness, the scan data of the instrument itself are usually used to obtain the orientation. There are also mixed cases where the scanner is fast relative to the platform movement. This is the usual situation in robotics, but applications in city model generation have also been reported (Früh and Zakhor, 2001).

In surveying, artificial targets are typically used for scan alignment. The targets are optimized for easy automatic detection, using distance (e.g., spheres) or intensity data (e.g., cylinders or planes covered with retroreflective foil). After an initial scan and interactive target selection, they are individually fine-scanned to improve redundancy, and thus, accuracy. Corresponding targets in other scans can then be selected fully automatically using a search procedure.

Although this process works reliably, it has several drawbacks. Artificial targets have to be distributed in the scene and collected later on; target selection usually requires manual intervention; and a fine scan of every target is required. According to our experience, the amount of time spent for all this exceeds the net scanning time by a factor of about five. In addition, since the number of targets is limited and their placement is often influenced by ad-hoc decisions or dictated by the site, a poor coverage of the measurement volume, and thus, an uneven distribution of alignment errors can result. Thus, methods which use the scan data itself for alignment are more desirable.

### 1.2 *Registration algorithms*

Registration is the process of finding the geometric transformation which makes corresponding locations in the two datasets  $S_1$  and  $S_2$  coincide. Due to

the six degrees of freedom to place and orient the acquisition sensor in space, any two corresponding points  $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^3$  with  $\mathbf{x}_1 \in S_1, \mathbf{x}_2 \in S_2$ , are related by an Euclidean (rigid) transformation

$$\mathbf{x}_1 = \mathbf{R}\mathbf{x}_2 + \mathbf{t}, \tag{1}$$

where  $\mathbf{R}$  is a  $3 \times 3$  rotation matrix, and  $\mathbf{t} \in \mathbb{R}^3$  is the translation vector. Usually, due to errors, the transformed point of  $\mathbf{x}_2$ , denoted as  $\mathbf{x}'_2$  (i.e.,  $\mathbf{x}'_2 = \mathbf{R}\mathbf{x}_2 + \mathbf{t}$ ), and its counterpart  $\mathbf{x}_1$  from  $S_1$ , do not exactly coincide. Then, the transformation parameters for  $\mathbf{R}$  and  $\mathbf{t}$  can be found by (least-squares) minimization of the (non-zero) sum  $\sum \|\mathbf{x}_1 - \mathbf{x}'_2\|^2$ . Closed form solutions in the more general case of a similarity transformation have been given by Sansò (1973) and Horn (1987). Therefore, the major problem is to establish correspondences between the two datasets  $S_1$  and  $S_2$ .

If a good initial guess for the similarity transformation is available, a proximity search can be used to define correspondences, which in turn can be used to optimize  $\mathbf{R}$  and  $\mathbf{t}$ . An iterative scheme, which alternately establishes correspondences and recomputes the transformation parameters has been introduced as the *Iterative Closest Point (ICP)* algorithm by Chen and Medioni (1991) and Besl and McKay (1992). It is nowadays used widely and is also available in commercial software. Many variants of ICP were proposed, differing in the selection, matching, weighting and rejection of correspondences, as well as the employed error metric. ICP is guaranteed to converge, since both the search for closest points and the optimization of the transformation parameters reduce the error. Optimization may however end up in a local minimum. Therefore, methods to obtain a good initial alignment are of importance. For a survey of ICP variants and coarse alignment, see (Rusinkiewicz and Levoy, 2001; Gruen and Akca, 2005).

### 1.3 Structure of the paper

The aim of this paper is to assess two different methods for coarse alignment of terrestrial laser scans. For this purpose, a test scene has been acquired in an urban environment, as described in section 2. Section 3 describes the first method, which uses extraction and assignment of planar patches, and shows the results obtained. Section 4 introduces the second method, which is based on the normal distributions transform known from robotics. We describe the modifications we applied and present the results. Section 5 draws conclusions and identifies possible future work.

## 2 The test scene

Often, synthetic test data are used to evaluate the performance of registration algorithms (Rusinkiewicz and Levoy, 2001), the advantage being that exact reference values are available. However, it is our intention to assess what can be achieved when typical real-world data are used. In order to obtain reference values, manual alignment using artificial targets has been carried out, leading to errors generally in the range of a few millimeters. Thus, the reference data obtained in this manner are sufficiently accurate to evaluate the performance of coarse registration algorithms.

The scans have been acquired using a Riegl LMS-Z360I scanner, which has a single shot measurement accuracy of 12 mm, field of view of  $360^\circ \times 90^\circ$  and a range of about 200 m. At  $0.12^\circ$  step width, a full scan takes approximately four minutes and results in a maximum of  $3000 \times 750 = 2.25$  million scanned points, of which in the test scene, 1.9 million are valid on average. A digital Nikon D100 SLR camera is mounted on the scanner with calibrated relative orientation. This allows to provide color information for each scanned point.

We selected an area called “Holzmarkt” in the historic district of Hannover, Germany, as an example for a densely built-up area. Fig. 1 shows the point cloud, combined from all scans.

For evaluation of the proposed registration algorithms, we placed the scan positions systematically along a trajectory with a spacing of approximately 5 meters. Due to the short distance, successive scans show a large overlap. In total, we acquired 20 scans of which 12 were taken (approximately) upright, and another 8 with a tilted scan head. The tilted scans were acquired at the same positions as the upright scans. Fig. 2 shows all 12 scan locations in a cadastral map, where tilted scans are marked with an ‘a’ suffix.

For each scan, reference values for the position and orientation were obtained using the standard procedure provided by the software RiSCAN PRO. It is based on the placement of artificial targets (retro-reflective cylinders) in the scene. High resolution scans of the targets are used to compute the target center points with a high redundancy. Center point pairs are then used to compute the orientation parameters. The original point clouds were not changed by this and the reference values were only used to assess the results of our registration algorithms. Table 1 shows the relative positions and orientations of the scans for those combinations that have been used for the alignment tests. One can verify that the scanner has been placed at approximate distances of 5 m and with arbitrary orientation (except being upright).

Using the reference values, we also calculated the overlap between scans. For each scan pair  $(S_1, S_2)$ , points from  $S_2$  were transformed to the frame of  $S_1$

Pair	X [m]	Y [m]	Z [m]	$\omega$ [°]	$\phi$ [°]	$\kappa$ [°]
01-02	-5.50	0.96	0.02	-1.088	-0.112	51.731
01-03	-10.69	1.87	0.08	0.551	0.419	57.447
01-03a	-10.64	1.96	0.05	-25.707	15.540	62.495
01-04	-16.77	2.53	0.14	1.984	0.481	119.261
01-05	-21.05	4.24	0.16	-0.692	0.678	-118.535
01-05a	-21.12	4.11	0.09	40.577	-19.379	-111.274
01-06	-24.71	2.74	0.29	-0.154	0.276	29.409
01-06a	-24.71	2.71	0.28	13.897	-1.678	79.449
01-07	-30.30	2.46	0.30	-0.838	0.063	178.528
01-08	-31.63	-3.20	0.46	0.836	0.544	166.929
01-08a	-31.53	-3.22	0.42	8.467	29.713	164.756
01-09	-33.50	-8.03	0.54	0.001	-0.113	31.594
01-09a	-33.91	-8.17	0.49	-38.073	-29.496	-56.814
01-10	-33.48	-11.96	0.58	-1.120	1.491	-106.207
01-10a	-33.67	-11.97	0.51	-5.763	-46.751	169.994
01-11	-34.18	-17.32	0.51	0.115	-0.461	-57.484
01-11a	-34.00	-17.59	0.47	11.362	28.378	-19.364
01-12	-37.38	-28.76	0.54	0.674	-0.795	169.779
01-12a	-37.53	-28.72	0.47	-9.712	-45.053	165.549
02-03	-2.50	4.64	0.08	1.432	-0.958	5.733
03-04	-2.72	5.47	0.01	0.824	-1.174	61.834
04-05	3.58	2.90	-0.08	1.482	2.238	122.148
05-06	3.07	-2.51	0.07	0.096	0.665	147.948
06-07	-5.01	2.50	-0.01	-0.701	0.150	149.118
07-08	1.18	5.69	0.07	-1.661	-0.525	-11.598
08-09	0.73	5.13	0.14	0.664	0.829	-135.342
09-10	-2.04	-3.36	0.04	-0.115	1.954	-137.813
10-11	5.34	0.82	-0.19	1.530	1.731	48.711
11-12	7.92	-8.84	0.08	0.582	0.292	-132.744

Table 1

Reference values for the relative orientation of scan pairs. First part: relative orientation of SP 1 and all other scans. Second part: relative orientation of successive scans.

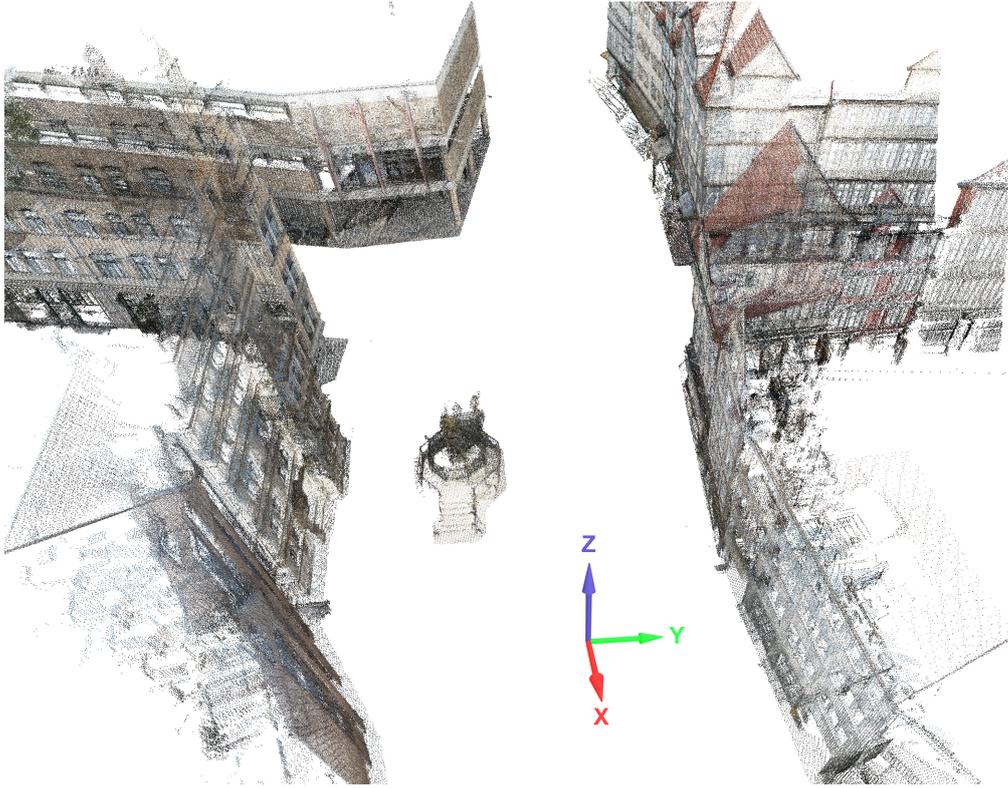


Fig. 1. Point cloud of the test scene (combination of all scans). The structure in the middle is a fountain. Scene dimensions are approximately  $70 \times 70 \times 25 \text{ m}^3$  ( $L \times W \times H$ ). The cloud was thinned and points on the ground were removed for better visualization.

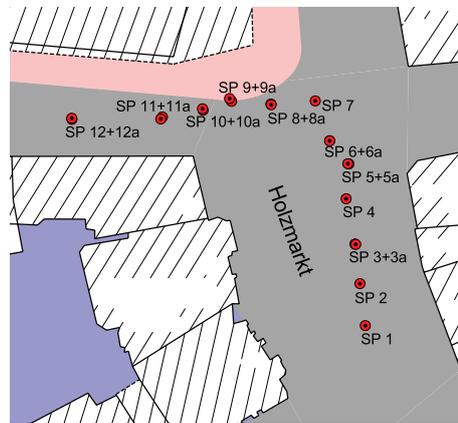


Fig. 2. Placement of scan positions along a trajectory, shown in a cadastral map.

using the reference transformation. If a point in  $S_1$  was found within a distance of 0.5 m in any of the coordinates, this was considered to be an overlapping point. Division of the number of overlapping points by the total number of points yields the overlap percentages shown in Table 2. Invalid points (such as points in the sky) do not take part in the computation.

Pair	[%]	Pair	[%]	Pair	[%]
01-02	83.1	01-08a	45.7	02-03	82.6
01-03	77.7	01-09	27.3	03-04	81.3
01-03a	73.3	01-09a	28.8	04-05	83.6
01-04	68.8	01-10	16.2	05-06	80.3
01-05	63.0	01-10a	11.4	06-07	81.0
01-05a	59.7	01-11	9.9	07-08	81.3
01-06	50.5	01-11a	12.2	08-09	81.0
01-06a	54.5	01-12	3.6	09-10	82.9
01-07	50.7	01-12a	2.3	10-11	77.2
01-08	43.6			11-12	74.9

Table 2

Overlap percentage for the scan pairs used for the alignment tests. First two columns: overlap of SP 1 with all other scans. Last column: overlap of successive scans.

### 3 Registration using planar patches

#### 3.1 Registration based on primitives

Single points  $\mathbf{x}_1 \in S_1$ ,  $\mathbf{x}_2 \in S_2$  do not carry enough information to assess if they form a valid point pair which is related by Eq. 1. A standard approach, especially for finding an initial alignment, is therefore to extract more meaningful, attributed, higher level features from both scenes  $S_1$ ,  $S_2$ , and to estimate the transformation using pairs of those features. This has several advantages. First, as compared to the original point sets, there are much fewer pair candidates. Second, attributes of features can be used to prune the search tree (as long as they are invariant under the considered transformation). And third, depending on the kind of features, fewer feature pairs may be required to fix all degrees of freedom of the transformation. All this helps to reduce the size of the search tree. In general, this approach is equivalent to the one in feature-based object recognition, where  $S_1$  is the scene and  $S_2$  is a given model (Grimson, 1990).

What kind of features make sense depends strongly on scene contents. Especially for free-form surfaces, local surface attributes like normal vector and curvature (Bae and Lichti, 2004), oriented point pairs (Winkelbach et al., 2006), or more complex representations like spin images (Johnson and Hebert, 1997) have been used. In industrial environments, standard shapes, especially pla-

nar and cylindrical surfaces, have been proposed for model-based registration (Rabbani et al., 2007). In city modelling, polyhedral objects can be expected, and consequently, planar patches have been employed (von Hansen, 2006; Dold and Brenner, 2006).

### 3.2 Planar segmentation

Segmentation of laser scans is a very extensive research field. We can narrow this down in our special case. First, we are interested in planar patches only. Second, each scan is segmented separately, so that neighborhoods imposed by the regular scanning raster can be exploited (usually called *range image* segmentation). There are many different strategies for segmentation, using top down (e.g., split and merge), bottom-up (e.g., region growing) or global (e.g., Hough transform) approaches. Regarding bottom-up strategies, scanline grouping (Jiang and Bunke, 1992) is fast and has been shown to yield good results in a comparative study (Hoover et al., 1996). On the other hand, its anisotropic working principle often leads to jagged region borders. For this reason we used standard region growing, working on the regular raster of scan points.

Region growing iterates the two steps of seed region selection and region expansion. Seed regions are prioritized according to their local planarity, which is computed using the residuals of a local best-fit plane. Once a seed region is selected, scan points along the region border are added if they lie in the plane (within a threshold). After this, the plane equation is updated. The region is grown until no more points can be added, in which case the algorithm proceeds by picking the next seed region.

Both the selection of seed regions and the update of the plane equation during region growing requires the determination of a best-fit plane, given  $k$  points  $\mathbf{x}^{(i)}$ ,  $1 \leq i \leq k$ . Using least squares minimization, the desired plane  $\langle \mathbf{n}, \mathbf{x} \rangle - d = 0$ , defined by (unit) normal vector  $\mathbf{n}$  and distance to origin  $d$  minimizes

$$\sum_{i=1}^k \left( \langle \mathbf{n}, \mathbf{x}^{(i)} \rangle - d \right)^2 \quad (2)$$

under the condition  $\|\mathbf{n}\| = 1$  ( $\langle \cdot, \cdot \rangle$  denotes the scalar product). It is well known from principal component analysis (Duda and Hart, 1973), that the solution normal vector  $\mathbf{n}$  is the eigenvector belonging to the smallest eigenvalue of the  $3 \times 3$  matrix  $\sum_{i=1}^k \tilde{\mathbf{x}}^{(i)} \tilde{\mathbf{x}}^{(i)\mathbf{T}}$ , where  $\tilde{\mathbf{x}}^{(i)} = \mathbf{x}^{(i)} - \bar{\mathbf{x}}$  are point coordinates reduced by the center of gravity  $\bar{\mathbf{x}} = \sum_{i=1}^k \mathbf{x}^{(i)} / k$ . The solution plane also includes the center of gravity, such that  $d = \mathbf{n}^{\mathbf{T}} \bar{\mathbf{x}}$ , and the minimum eigenvalue is equal to the sum of squared distances in Eq. 2.

### 3.3 Determination of the transformation parameters

If three correspondences of planar patches are known, all parameters of the Euclidean transformation in Eq. 1 can be determined. We denote the three plane equations as

$$\langle \mathbf{n}_i, \mathbf{x} \rangle - d_i = 0 \quad (3)$$

$$\langle \mathbf{m}_i, \mathbf{x} \rangle - e_i = 0 \quad (4)$$

$$\langle \mathbf{p}_i, \mathbf{x} \rangle - f_i = 0 \quad (5)$$

where  $\mathbf{n}_i$ ,  $\mathbf{m}_i$ ,  $\mathbf{p}_i$  are normal vectors of unit length,  $d_i$ ,  $e_i$ ,  $f_i$  are the plane distances from the origin, and for each of the equations,  $i = 1$  and  $i = 2$  form a pair. Note that more than three plane pairs can be used to improve the result, however, when using schemes such as random sampling consensus (RANSAC), it is often desired to compute the transformation from a minimum number of correspondences.

#### 3.3.1 Rotation

For the rotational component, two normal vector pairs, for example  $(\mathbf{n}_1, \mathbf{n}_2)$  and  $(\mathbf{m}_1, \mathbf{m}_2)$ , are required. Since each pair fixes two degrees of freedom but  $\mathbf{R}$  has only three degrees of freedom, one constraint is redundant. In the presence of noise, however, the angles  $\angle(\mathbf{n}_1, \mathbf{m}_1)$  (in  $S_1$ ) and  $\angle(\mathbf{n}_2, \mathbf{m}_2)$  (in  $S_2$ ) are usually different so that the normal vectors cannot be aligned perfectly. The methods proposed differ in how they distribute the residuals.

Grimson and Lozano-Perez (1984) propose a geometric approach. First, the (unit vector) rotation axis  $\mathbf{r}$  is obtained from

$$\tilde{\mathbf{r}} = (\mathbf{n}_1 - \mathbf{n}_2) \times (\mathbf{m}_1 - \mathbf{m}_2), \quad \mathbf{r} = \tilde{\mathbf{r}} / \|\tilde{\mathbf{r}}\|.$$

To compute the rotation angle  $\phi$ , let  $\mathbf{w} = \langle \mathbf{n}_2, \mathbf{r} \rangle \mathbf{r}$  be the projection of  $\mathbf{n}_2$  onto  $\mathbf{r}$ . Adding  $\mathbf{u} = \mathbf{n}_2 - \mathbf{w}$  and  $\mathbf{v} = \mathbf{r} \times \mathbf{u}$  yields a local Cartesian coordinate frame  $\{\mathbf{u}, \mathbf{v}, \mathbf{w}\}$ . Using this frame, the rotation of  $\mathbf{n}_2$  around  $\mathbf{r}$  by the angle  $\phi$  is given by

$$\mathbf{n}'_2 = \mathbf{u} \cos \phi + \mathbf{v} \sin \phi + \mathbf{w}. \quad (6)$$

Conversely, if  $\mathbf{n}'_2$  is required to be equal to  $\mathbf{n}_1$  (the rotation of  $\mathbf{n}_2$  should align with  $\mathbf{n}_1$ ) and  $\phi$  is sought for, Eq. 6 can be used to obtain  $\phi$ , since  $\langle \mathbf{n}_1, \mathbf{u} \rangle = \langle \mathbf{n}'_2, \mathbf{u} \rangle = \|\mathbf{u}\|^2 \cos \phi$ , and  $\langle \mathbf{n}_1, \mathbf{v} \rangle = \|\mathbf{v}\|^2 \sin \phi$ . Expanding  $\mathbf{u}$  and  $\mathbf{v}$ , noting that  $\langle \mathbf{n}_1, \mathbf{r} \rangle = \langle \mathbf{n}_2, \mathbf{r} \rangle$  and normal vectors are unit vectors,

$$\begin{aligned}\cos \phi &= \frac{\langle \mathbf{n}_2, \mathbf{n}_1 \rangle - \langle \mathbf{n}_2, \mathbf{r} \rangle \langle \mathbf{n}_1, \mathbf{r} \rangle}{1 - \langle \mathbf{n}_2, \mathbf{r} \rangle \langle \mathbf{n}_1, \mathbf{r} \rangle} \\ \sin \phi &= \frac{\langle \mathbf{r} \times \mathbf{n}_2, \mathbf{n}_1 \rangle}{1 - \langle \mathbf{n}_2, \mathbf{r} \rangle \langle \mathbf{n}_1, \mathbf{r} \rangle}.\end{aligned}$$

Substitution into Eq. 6 then yields

$$\begin{aligned}\mathbf{n}'_2 &= \mathbf{n}_2 \cos \phi + \langle \mathbf{n}_2, \mathbf{r} \rangle \mathbf{r} (1 - \cos \phi) + \mathbf{r} \times \mathbf{n}_2 \sin \phi \\ &= [\mathbf{I} \cos \phi + \mathbf{r} \mathbf{r}^T (1 - \cos \phi) + [\mathbf{r}]_{\times} \sin \phi] \mathbf{n}_2 \\ &= \mathbf{R} \mathbf{n}_2,\end{aligned}$$

where  $\mathbf{R}$  is the desired rotation matrix. Although the computation of the rotation axis  $\mathbf{r}$  is symmetric, the determination of  $\phi$  is not, since only the pair  $(\mathbf{n}_1, \mathbf{n}_2)$  is used. Thus, in the presence of noise (when  $\angle(\mathbf{n}_1, \mathbf{m}_1)$  and  $\angle(\mathbf{n}_2, \mathbf{m}_2)$  are different as noted earlier), the solution will exactly align  $\mathbf{n}_1$  and  $\mathbf{n}_2$ , but not  $\mathbf{m}_1$  and  $\mathbf{m}_2$ . As the authors note, this can be alleviated by averaging  $\phi$  over several normal vector pairs.

A simpler solution is obtained by selectively discarding constraints (Horn, 1987). Using  $\mathbf{n}_1$  and  $\mathbf{m}_1$ , a Cartesian coordinate frame  $\{\mathbf{u}_1, \mathbf{v}_1, \mathbf{w}_1\}$  can be constructed by

$$\begin{aligned}\mathbf{u}_1 &= \mathbf{n}_1 & (7) \\ \tilde{\mathbf{v}}_1 &= \mathbf{m}_1 - \langle \mathbf{m}_1, \mathbf{u}_1 \rangle \mathbf{u}_1, \mathbf{v}_1 = \tilde{\mathbf{v}}_1 / \|\tilde{\mathbf{v}}_1\| & (8) \\ \mathbf{w}_1 &= \mathbf{u}_1 \times \mathbf{v}_1,\end{aligned}$$

where the computation of  $\mathbf{v}_1$  uses standard Gram-Schmidt orthonormalization, i.e.,  $\mathbf{u}_1$  and  $\mathbf{v}_1$  span the same plane as  $\mathbf{n}_1$  and  $\mathbf{m}_1$ . The vector  $\mathbf{w}_1$  is constructed to be normal to this plane. Then, let  $\mathbf{M}_1 = [\mathbf{u}_1 \mathbf{v}_1 \mathbf{w}_1]$ , writing  $\mathbf{u}_1, \mathbf{v}_1, \mathbf{w}_1$  as column vectors. In the same way,  $\mathbf{M}_2$  can be obtained from  $\mathbf{n}_2$  and  $\mathbf{m}_2$ . Since both  $\mathbf{M}_1$  and  $\mathbf{M}_2$  are orthogonal by construction,

$$\mathbf{R} = \mathbf{M}_1 \mathbf{M}_2^T \quad (9)$$

is orthogonal and in fact is the desired rotation matrix (since  $\mathbf{M}_2^T \mathbf{n}_2$  gives the components of  $\mathbf{n}_2$  along the axes  $\{\mathbf{u}_2, \mathbf{v}_2, \mathbf{w}_2\}$  and  $\mathbf{M}_1$  maps this back to the first coordinate frame). As the  $\mathbf{m}_i$  are only used in Eq. 8 to define the plane, the solution is not symmetric (Eq. 8 discards one constraint). Again, it will align  $\mathbf{n}_1$  and  $\mathbf{n}_2$  and leave all residuals at  $(\mathbf{m}_1, \mathbf{m}_2)$ . However, the error can be distributed evenly by a simple modification, using the bisector  $\tilde{\mathbf{u}}_1 = \mathbf{n}_1 + \mathbf{m}_1$ ,  $\mathbf{u}_1 = \tilde{\mathbf{u}}_1 / \|\tilde{\mathbf{u}}_1\|$  instead of Eq. 7.

If more than two plane correspondences are to be used to obtain  $\mathbf{R}$ , the solutions given by Sansò (1973) or Horn (1987) can be applied. Denoting the  $i \geq 2$

normal vectors in  $S_1$  and  $S_2$  now with  $\mathbf{n}_{1,i}$  and  $\mathbf{n}_{2,i}$ , respectively, the goal is to find the rotation  $\mathbf{R}$  which minimizes the quadratic error  $\sum_i \|\mathbf{R}\mathbf{n}_{2,i} - \mathbf{n}_{1,i}\|^2$  or, equivalently, maximises the sum over the dot products,

$$\sum_i \langle \mathbf{R}\mathbf{n}_{2,i}, \mathbf{n}_{1,i} \rangle. \quad (10)$$

Using unit quaternions  $\hat{\mathbf{q}}$ , a rotation  $\mathbf{R}\mathbf{n}$  is expressed as  $\hat{\mathbf{q}}\mathbf{n}\hat{\mathbf{q}}^*$ , where  $\hat{\mathbf{n}}$  is the (purely imaginary) quaternion representation of vector  $\mathbf{n}$ , and  $\hat{\mathbf{q}}^*$  is the conjugate of  $\hat{\mathbf{q}}$ . Manipulating Eq. 10,

$$\begin{aligned} \sum_i \langle \hat{\mathbf{q}}\hat{\mathbf{n}}_{2,i}\hat{\mathbf{q}}^*, \hat{\mathbf{n}}_{1,i} \rangle &= \sum_i \langle \hat{\mathbf{q}}\hat{\mathbf{n}}_{2,i}, \hat{\mathbf{n}}_{1,i}\hat{\mathbf{q}} \rangle \\ &= \sum_i \langle \bar{\mathbf{N}}_{2,i}\hat{\mathbf{q}}, \mathbf{N}_{1,i}\hat{\mathbf{q}} \rangle \\ &= \sum_i \hat{\mathbf{q}}^T \bar{\mathbf{N}}_{2,i}^T \mathbf{N}_{1,i} \hat{\mathbf{q}} \\ &= \hat{\mathbf{q}}^T \mathbf{N} \hat{\mathbf{q}} \end{aligned} \quad (11)$$

where  $\mathbf{N}_{1,i}$ ,  $\bar{\mathbf{N}}_{2,i}$  are matrix representations of their respective  $\hat{\mathbf{n}}_{k,i}$  (for details, see (Horn, 1987)) and

$$\mathbf{N} = \sum_i \bar{\mathbf{N}}_{2,i}^T \mathbf{N}_{1,i}. \quad (12)$$

It is well known that the quadratic form in Eq. 11 is maximized under the condition  $\|\hat{\mathbf{q}}\| = 1$  when  $\hat{\mathbf{q}}$  is the eigenvector of  $\mathbf{N}$  belonging to the largest eigenvalue. After  $\hat{\mathbf{q}}$  is determined, the desired rotation matrix  $\mathbf{R}$  can be obtained as the Rodriguez matrix. To summarize, given both sets of normal vectors  $\{\mathbf{n}_{1,i}\}$ ,  $\{\mathbf{n}_{2,i}\}$ ,  $\mathbf{N}$  is computed from Eq. 12, and the eigenvector of  $\mathbf{N}$  belonging to the largest eigenvalue is converted to a rotation matrix  $\mathbf{R}$  using the Rodriguez matrix. Note that if only two normal vector pairs are used, the result is identical to the bisector variant of Eq. 9.

### 3.3.2 Translation

To determine the translation, three plane pairs are required. Assume  $S_2$  has already been rotated, so that only the translation component  $\mathbf{t}$  in Eq. 1 has to be determined. Then, Eq. 3 becomes

$$\begin{aligned} \langle \mathbf{n}_1, \mathbf{x} \rangle - d_1 &= 0 \\ \langle \mathbf{n}'_2, \mathbf{x} - \mathbf{t} \rangle - d_2 &= 0. \end{aligned}$$

Assuming a correct rotation,  $\mathbf{n}_1 = \mathbf{n}'_2 = \mathbf{n}$ . Eliminating  $\mathbf{x}$  yields  $\langle \mathbf{n}, \mathbf{t} \rangle = d_1 - d_2$ . Doing the same for Eqs. 4 and 5 yields

$$\mathbf{A}\mathbf{t} = \begin{bmatrix} \mathbf{n}^T \\ \mathbf{m}^T \\ \mathbf{p}^T \end{bmatrix} \mathbf{t} = \begin{bmatrix} d_1 - d_2 \\ e_1 - e_2 \\ f_1 - f_2 \end{bmatrix} = \mathbf{l} \quad (13)$$

where  $\mathbf{A}$  consists of  $\mathbf{n}$ ,  $\mathbf{m}$ ,  $\mathbf{p}$  written as row vectors. In accordance with intuition,  $\mathbf{t}$  can only be determined if  $\mathbf{n}$ ,  $\mathbf{m}$ ,  $\mathbf{p}$  span 3D space, in which case  $\mathbf{A}$  is nonsingular. If more than three plane pairs are available, Eq. 13 can be used to determine  $\mathbf{t}$  using least squares adjustment,

$$\mathbf{t} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{l}. \quad (14)$$

### 3.4 Search for corresponding patches

As noted, at least  $k \geq 3$  corresponding patches are required to establish an Euclidean transformation. Suppose a total of  $p_1$  and  $p_2$  patches are extracted from  $S_1$  and  $S_2$ , respectively. Then the number of possible selections of  $k$  pairs is

$$\binom{p_1}{k} \cdot \binom{p_2}{k} \cdot k!$$

where the first two terms are due to the possibilities to pick  $k$  patches and  $k!$  is due to pair permutation. In the case of  $k = 3$ , the absolute number can be reduced by a factor of two due to chirality, nevertheless asymptotic algorithmic complexity remains  $O(p_1^3 p_2^3)$ .

A standard technique to deal with complexity is to compute feature vectors for the extracted patches, compare them or their relations during patch assignment and prune the search tree accordingly (Grimson, 1990). The basic drawback is that if the feature computation is not sufficiently reliable, the correct solution may be pruned as well. We tried patch features such as area, circumference, height-to-width ratio of the bounding box and mean intensity value in an earlier investigation (Dold and Brenner, 2006). However, those features were not as reliable as we had hoped. Especially if scan positions are further apart, occlusions and differences in sampling density can lead to different feature values. Also, if many buildings stand in a row, with facades forming (approximately) one plane, it may happen that the plane is subdivided

vided into patches differently so that the borders of the patches are not very reliable.

Therefore, we adopted a priority scheme instead. A sensible criterion for optimal plane triples would be that the error to determine  $\mathbf{t}$  in Eq. 14 is minimized. As is well known, the corresponding covariance matrix is  $\Sigma_{\mathbf{t}} = (\mathbf{A}^T \mathbf{A})^{-1}$ . Let  $\mathbf{U} \mathbf{S} \mathbf{V}^T$  be the singular value decomposition of  $\mathbf{A}$ , with  $\mathbf{U}$ ,  $\mathbf{V}$  orthogonal and  $\mathbf{S} = \text{diag}(s_1, s_2, s_3)$  containing the singular values, then  $\mathbf{A}^T \mathbf{A} = \mathbf{V} \mathbf{S}^2 \mathbf{V}^T$ , or

$$\Sigma_{\mathbf{t}} = (\mathbf{A}^T \mathbf{A})^{-1} = \mathbf{V} \text{diag}(1/s_1^2, 1/s_2^2, 1/s_3^2) \mathbf{V}^T. \quad (15)$$

The variance along any axis  $1 \leq i \leq 3$ , and also the sum of variances along all axes is bound by

$$\frac{1}{s_i^2} \leq \frac{1}{s_1^2} + \frac{1}{s_2^2} + \frac{1}{s_3^2} \leq \frac{3}{s_1^2 s_2^2 s_3^2},$$

since  $\text{trace}(\mathbf{A}^T \mathbf{A}) = \text{trace}(\mathbf{A} \mathbf{A}^T) = 3 = s_1^2 + s_2^2 + s_3^2$  (note that normal vectors in Eq. 13 have unit length). Thus, large values of  $s_1^2 s_2^2 s_3^2 = \det^2(\mathbf{A})$  guarantee small values for the sum of variances (and also, the variance  $1/s_i^2$  along any axis). In accordance with intuition, the error is minimized when  $|\det(\mathbf{A})|$  is largest, or equivalently, the absolute value of the triple product of  $\mathbf{m}$ ,  $\mathbf{n}$ ,  $\mathbf{p}$  is largest. This happens when all normal vectors are perpendicular ( $s_1 = s_2 = s_3 = 1$ ).

To find good matches first, we compute the triple products of all plane triples from  $S_1$  and  $S_2$  separately and sort them in decreasing order. From the list, triple pairs are selected with large triple products first, plane pairs are assigned and the transformation is computed. This transformation is then accepted or rejected based on a score computed using all planar patches. The score is defined as the count of patches having similar plane parameters. Two planes are considered similar if the directions of their normal vectors and their distances to the origin are similar. Using Eq. 3, this can be expressed as

$$\langle \mathbf{n}_1, \mathbf{n}'_2 \rangle \geq \cos \theta, \quad |d_1 - d'_2| \leq \delta, \quad (16)$$

where  $\theta$  is the normal vector angle threshold and  $\delta$  is the distance threshold. Since the best score is not known in advance as it depends on the (unknown) overlap, we implemented a voting scheme. Instead of testing all triple combinations, iteration stops as soon as the same transformation is found for a predefined number of times. This reflects the assumption that correct triple assignments lead to one and the same transformation, whereas wrong assignments lead to many different transformations. In the voting, only transformations with a minimum score take part, which is usually 40% of the total

number of planes.

### 3.5 Experiments

As described in section 3.2, the first step is the planar segmentation of both scans. As an example, Fig. 3 shows the segmentation results for scan positions SP 1 and SP 2. The numbers shown in the figure are the patch numbers, which are assigned independently in each scan. The threshold defining the maximum deviation of a point from the patch during region growing was set to 6 cm. With smaller thresholds, we observed a tendency towards oversegmentation. Only the 30 largest patches are shown in Fig. 3. In general,  $n$  largest patches are kept for further processing, with  $n = 50$  typically. For scans known to have little overlap,  $n$  can be set higher to increase the chance to find the correct transformation, however this increases the processing time also.

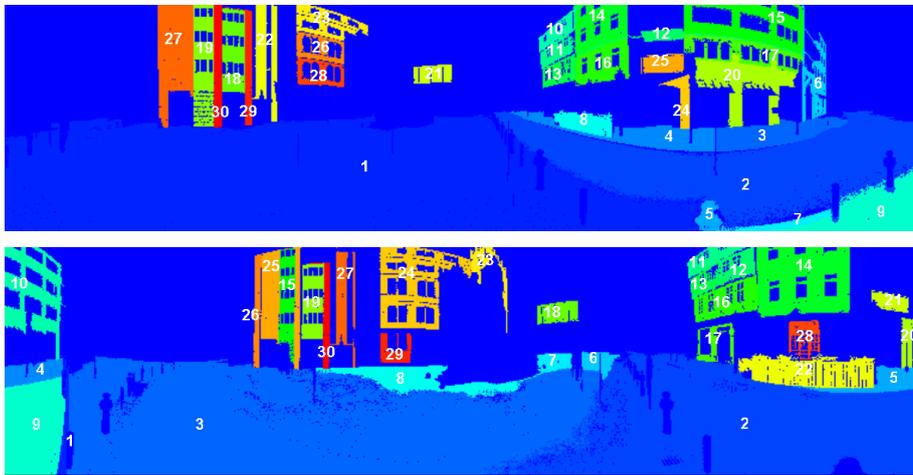


Fig. 3. Planar segmentations computed for SP 1 (top) and SP 2 (bottom).

In order to evaluate the segmentation result, we compared the extracted planar patches using the reference orientation. The overlap is quantified by counting points in the overlapping patch areas. As a result, approximately 50% of the segmented area in SP 1 has no correspondence in SP 2, although the scans are overlapping by 83% (Table 2). This difference is due to points not belonging to any patch as well as too small patches, which have been discarded. An additional 40% of the corresponding patches belong to the ground. Consequently, there are approximately only 10% of the scan left to identify other patch correspondences required to define the remaining two spatial directions. Table 3 shows some correspondences for the largest patches of Fig. 3.

After segmentation, the algorithm proceeds by computing all patch triples, then examining triple correspondences in decreasing order of their triple product, and computing a score based on the angle  $\theta$  and distance  $\delta$  thresholds

Patch number in SP 1	Patch numbers in SP 2	Overlap
1	1,2,3,6,8,9	28.7%
2	2,3,9	9.9%
8	22	1.5%
16	14	1.5%
4	2,4,5	2.3%
27	25,26	0.5%
14	14	0.8%
13	16	0.7%

Table 3

Patch correspondences between SP 1 and SP 2 for the largest patches. Overlap is the percentage of overlapping points relative to the total number of points.

(Eq. 16). In order to analyze how those thresholds should be selected, we varied  $\theta$  and  $\delta$  over a large range and evaluated the score for each setting. Fig. 4 shows the dependency of the score on  $\theta$  and  $\delta$  for two example transformations, one being correct and one being wrong. One can see that in case of a correct transformation, even small thresholds will lead to a high score. In contrast, the wrong solution leads to a low score for all reasonable thresholds. On the basis of our analysis, we used  $2^\circ$  to  $3^\circ$  for  $\theta$  and 0.2 m to 0.4 m for  $\delta$ .

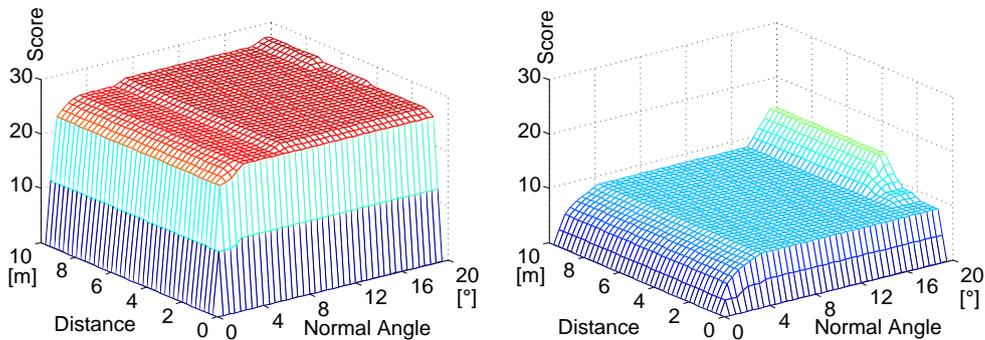


Fig. 4. Dependency of the score function on normal vector angle ( $\theta$ ) and distance ( $\delta$ ) threshold. Left for a correct, right for a wrong transformation.

We applied the registration algorithm to the test scene. The first scan is treated as the reference and all other scans are registered to it. Because of the placement of the scan positions along the chosen trajectory (Fig. 2), the overlap with the first scan decreases with increasing scan number.

As a result of our evaluation, we found a quite stable and accurate alignment. Of the 20 scans, in 14 cases (positions 2 to 10) the correct transformation parameters were found, failing only for cases 10a (tilted scan) and larger. All

the failed scans have an overlap area with the first scan which is smaller than 12.2%. The last scan we were still able to register overlaps by approximately 16%. Table 4 shows the deviation from the reference orientation for all successful alignments. One can see that the differences are less than 20 cm in the plane, 40 cm in height, and  $0.5^\circ$  in orientation, which is certainly good enough for a successive iterative alignment.

Scan	$\Delta X$	$\Delta Y$	$\Delta Z$	$\Delta\omega$	$\Delta\phi$	$\Delta\kappa$
pairs	[m]	[m]	[m]	[ $^\circ$ ]	[ $^\circ$ ]	[ $^\circ$ ]
01-02	-0.022	-0.033	-0.002	0.132	-0.082	0.050
01-03	0.025	-0.018	0.054	0.089	-0.185	-0.075
01-03a	0.045	-0.043	-0.046	-0.155	-0.122	-0.161
01-04	0.090	-0.039	0.091	-0.080	-0.237	-0.049
01-05	-0.073	-0.006	-0.038	0.039	0.083	0.023
01-05a	-0.143	0.103	-0.008	0.086	-0.389	0.069
01-06	0.052	0.056	0.040	-0.029	-0.307	0.400
01-06a	0.077	0.091	0.013	0.073	-0.006	0.440
01-07	-0.050	-0.035	0.017	-0.145	0.149	0.031
01-08	-0.080	-0.006	0.269	-0.053	-0.502	0.420
01-08a	-0.098	0.154	0.259	-0.196	0.042	-0.469
01-09	-0.163	0.026	0.343	-0.286	-0.052	-0.109
01-09a	-0.076	-0.021	0.398	-0.303	0.487	0.430
01-10	-0.097	-0.006	0.240	-0.420	0.004	-0.258

Table 4

Deviation of the translation and rotation parameters from the reference values for the registration based on planar patches.

Considering the required computation time, segmentation of one scan takes around 14 to 26 seconds with an average of 19 seconds on a standard 2 GHz Pentium PC. Matching took an average of 35 seconds when  $n = 50$  planes were kept. Thus, in total, 72 seconds were required on average to match two scans.

## 4 Registration using the normal distributions transform

### 4.1 The normal distributions transform (NDT)

The NDT is a method originally proposed by Biber (2003) for the purpose of robot navigation. In robotics, a well-known problem is to navigate a robot in an unknown environment, and to build a map of this environment at the same time (called simultaneous localization and mapping (SLAM), see Thrun et al. (2005)). In this context, major concerns are convergence radius and speed. The original method applies to 2D laser scan data (acquired in a single plane), as follows.

The NDT converts the original point cloud of the first scan  $S_1$  into an alternative representation, which captures the distribution of the points, rather than individual points. For this, the area covered by  $S_1$  is subdivided into a regular grid of cells. It is assumed that the distribution of points inside each cell is characterized to a sufficient extent by a normal distribution. Thus, for each cell  $C_i$  the mean  $\mathbf{q}_i$  and the covariance matrix  $\Sigma_i$  are calculated, using

$$\mathbf{q}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in C_i} \mathbf{x},$$

$$\Sigma_i = \frac{1}{n_i - 1} \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{q}_i)(\mathbf{x} - \mathbf{q}_i)^T$$

where  $n_i$  is the number of points in cell  $C_i$ . The probability of measuring a point  $\mathbf{x}$  in cell  $C_i$  is then modeled by the normal distribution  $N(\mathbf{q}_i, \Sigma_i)$ ,

$$p(\mathbf{x}) = c \cdot \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{q}_{i(\mathbf{x})})^T \Sigma_{i(\mathbf{x})}^{-1} (\mathbf{x} - \mathbf{q}_{i(\mathbf{x})})\right),$$

where  $i(\mathbf{x})$  is the index of the cell the point  $\mathbf{x}$  falls into. For the purposes of the NDT,  $c$  is set to 1. Fig. 5 shows an example of original scan points and the corresponding NDT.

In order to register two scans, the original points  $\mathbf{x}_2$  of the second scan are transformed using Eq. 1 in two dimensions with  $\mathbf{R} = \mathbf{R}(\kappa)$  being a  $2 \times 2$  rotation matrix and  $\mathbf{t} \in \mathbb{R}^2$ . This is fully described by its three parameters  $\mathbf{a} = (\mathbf{t}^T, \kappa)^T$ . Using the transformed points  $\mathbf{x}'_2$ , a score

$$\text{score} = \text{score}(\mathbf{a}) = \sum_{\mathbf{x}'_2} p(\mathbf{x}'_2) \quad (17)$$

is assigned. The parameters  $\mathbf{a}$  are to be estimated so as to maximize the score

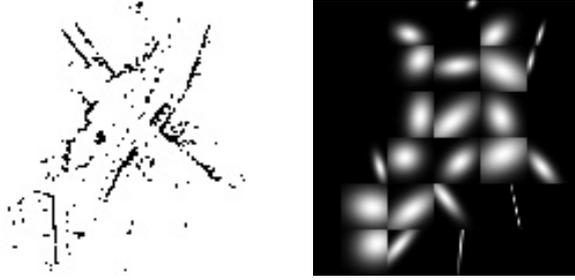


Fig. 5. Example for a 2D scan (left) and corresponding NDT (right). White pixels represent a high, black pixels a low probability.

function in Eq. 17. Since the optimization algorithms usually minimize values, the function  $f = -\text{score}(\mathbf{a})$  is used. The following steps are iterated until a convergence criterion is reached.

- (1) Transform each point  $\mathbf{x}_2$  of the second scan according to Eq. 1, using the current transformation parameters  $\mathbf{a}$ .
- (2) Compute the cell index  $i = i(\mathbf{x}'_2)$  in which each point  $\mathbf{x}'_2$  lies and retrieve the parameters  $\mathbf{q}_i$  and  $\Sigma_i$ .
- (3) Calculate the score value,  $\text{score}(\mathbf{a})$ , according to Eq. 17.
- (4) Compute new parameters by optimizing the function  $f = -\text{score}(\mathbf{a})$  using Newton's algorithm

$$\Delta \mathbf{a} = -\mathbf{H}^{-1} \nabla f^T, \quad (18)$$

with  $\mathbf{a}_{\text{new}} = \mathbf{a} + \Delta \mathbf{a}$ , where  $\mathbf{H}$  is the Hessian of  $f$ .

## 4.2 Modifications of the original NDT

### 4.2.1 Slicing 3D scans

As noted, the NDT has originally been applied to 2D laser scan data, which is a reasonable assumption in the area of robot positioning. In case of object reconstruction, however, full 3D laser scan data are available. It would be straightforward to extend the NDT to the 3D case, using voxels instead of grid cells. However, the processing involved would possibly negate the advantages of the NDT.

Considering the typical application, though, it is reasonable to assume that the laser scanner is set up approximately upright for each scan. Also, the approximate distance to the ground will be always the same or, if it varies, it will be technically simple to measure it. Therefore, the 2D NDT can be applied directly by cutting out a slice from the 3D data parallel to the ground.

However, a single slice may not contain enough data. As compared to the

case in robotics, where highly structured indoor environments prevail, natural scenes may lack the desired amount of information. Therefore, instead of a single slice, we extract a stack of  $n$  slices (see Fig. 6). Each slice leads to an independent NDT. The total score is defined as the sum of the individual scores,

$$\text{score}(\mathbf{a}) = \sum_{k=1}^n \text{score}_k(\mathbf{a}), \quad (19)$$

where  $\text{score}_k(\mathbf{a})$  is the score function in Eq. 17, taken for slice  $k$  and parameters  $\mathbf{a}$ . Equation 19 is optimized in the same way as the single score solution of Eq. 17.

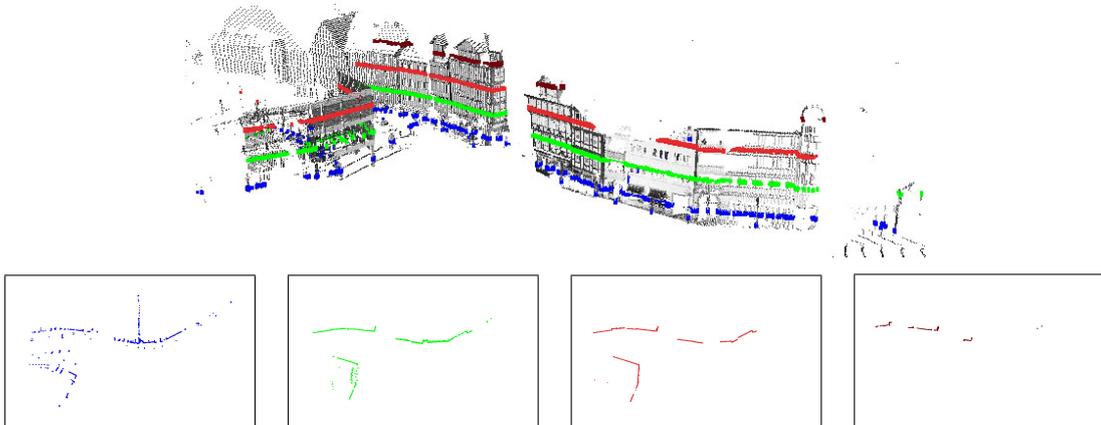


Fig. 6. Top: single scan with four slices. Bottom row: Top view of each slice.

#### 4.2.2 Coarse-to-fine strategy

Just as with the ICP algorithm, the NDT may yield the wrong solution in case of poor initial values. In fact, since the score function in Eq. 17 relies on points hitting cells of the NDT, no estimation will take place at all if the second data set does not overlap any non-empty cells of the NDT of the first data set. If only a few cells overlap, which easily happens in case of a wrong initial rotation, the distribution in these cells strongly affects the ability of the algorithm to converge to the correct solution. Especially if the cells are small, their distribution (particularly, their covariance matrix  $\Sigma_i$ , which is crucial to obtain the orientation) may not be meaningful.

In order to address this problem, the original NDT algorithm was extended to include a coarse-to-fine strategy. The cell size of the NDT is chosen large in the beginning and decreases in the course of the iteration (Fig. 7). For any cell size, the algorithm performs a fixed number of iterations, then it proceeds with the next smaller size.

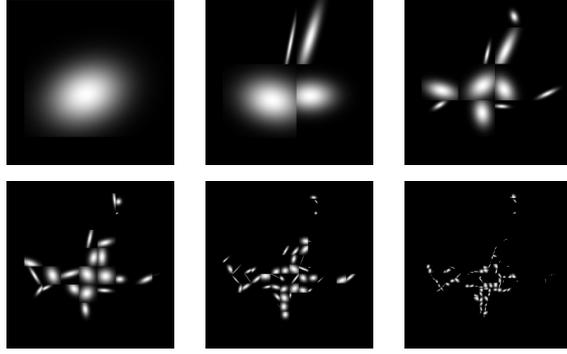


Fig. 7. Coarse to fine strategy used for the cell sizes of the NDT (all for SP 8, lowest slice). From left to right and top to bottom: 250, 125, 62, 31, 16, 8 meters cell width.

### 4.3 Experiments

To test the NDT approach, we cut four horizontal slices of 40 cm width, at heights 0 m, 4.5 m, 9 m and 13.5 m where 0 m is the height of the scan head. Slices one and two contain an average of 30,000, slice three 15,000, and slice four 5,000 points. The number decreases because there are not many high buildings in the scene. We ran the tests with one, two, three and four slices in all possible combinations.

The initial grid size is  $250 \text{ m} \times 250 \text{ m}$  decreasing in eight steps down to  $2 \text{ m} \times 2 \text{ m}$ . For each cell size we iterate 1,000 times. Regarding scan positions, we perform two different tests. The first test is the same as in section 3.5 and aligns all scans to the first scan, i.e. scan pairs SP 1-SP 2, SP 1-SP 3, SP 1-SP 4, etc. The second test aligns successive scans, i.e. scan pairs SP 1-SP 2, SP 2-SP 3, SP 3-SP 4, etc. Due to the slicing principle, the tilted scans are not used.

We explored the convergence radius by modifying the orientation  $\kappa$  from  $-180^\circ$  to  $+180^\circ$  in 100 steps. For each  $\kappa$ , the NDT alignment is run, and it is determined if the algorithm finally reached the correct solution by comparison with the reference values. Using the results, the convergence radius is determined, which is the maximum interval of  $\kappa$  values which converge to the correct solution. The result of this test is shown in Fig. 8 for the alignment with the first scan and in Fig. 9 for the alignment of successive scans. In the figures, the given convergence radii are averaged over all combinations of slices. In Fig. 8, scan positions larger than seven are not shown because there was no convergence.

When looking at the convergence radius, a decrease is seen in Fig. 8, and there is no convergence at all for scan positions beyond SP 7. Note that from Fig. 2, SP 7 is approximately in the center of the intersection, with positions SP 1 to SP 7 being along one, and SP 7 to SP 12 being along another of

the intersecting streets, so that SP 7 can be considered as a bisecting point. Note also that while no convergence is obtained beyond SP 1-SP 7, Fig. 9 shows that the convergence radius for successive scans gets smaller at the SP 5-SP 6, SP 6-SP 7, SP 7-SP 8 combinations, but gets larger again at later combinations.

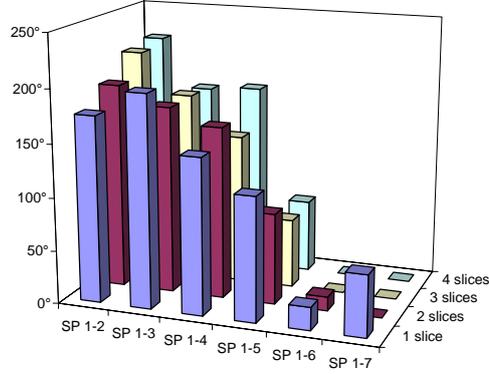


Fig. 8. Convergence radii for the alignment of SP 1 with SP 2 to SP 7.

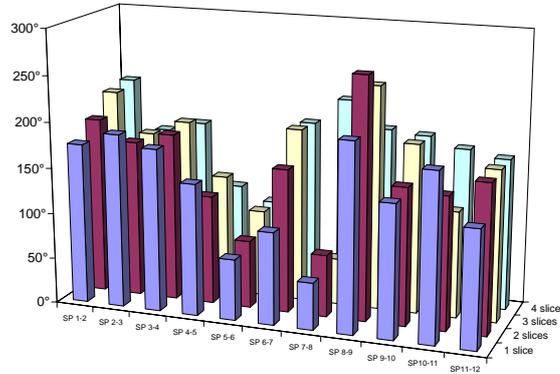


Fig. 9. Convergence radii for the alignment of successive scans.

To investigate this, we compared two cases in detail, the alignment of SP 5-SP 6 and SP 2-SP 3. Fig. 10 shows the error of the final orientation  $\kappa$  (as compared to the reference orientation) and the final score for all 100 tested start orientations in the range  $-180^\circ \leq \kappa < +180^\circ$  for SP 5-SP 6. As can be seen from the figure, there are essentially four results, corresponding to orientations  $90^\circ$  apart. This is due to the characteristics of the data. At the intersection (Fig. 10, bottom), two streets cross in an approximately right angle, and thus there is a fourfold symmetry. Consequently, with wrong start angles  $\kappa$ , the method leads to local minima. Comparing this to the results in Fig. 11 shows that in contrast, a mainly linear scene leads to only two different solutions. Note that in any case, the score gives a valid indication of when the correct solution is obtained. Note also, that a convergence radius of  $90^\circ$  or even  $180^\circ$  is rather large. The results indicate that in order to prevent wrong solutions, one can run the algorithm with a few (say, four) start orientations, distributed evenly over the  $-180^\circ \leq \kappa < +180^\circ$  range. Since the score proved to be a reliable indication for a correct solution, one can also start with the given orientation

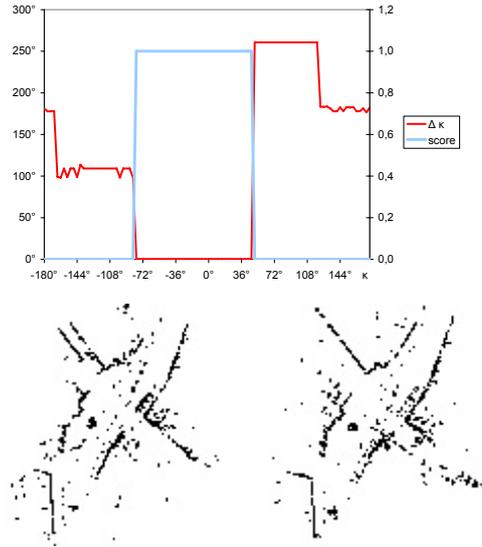


Fig. 10. Convergence (top) for all start orientations  $-180^\circ \leq \kappa < +180^\circ$  for SP 5-SP 6 (bottom left, right), located near an intersection of two streets. Shown is the error of the final orientation ( $\Delta\kappa$ , red) and the final score (blue).

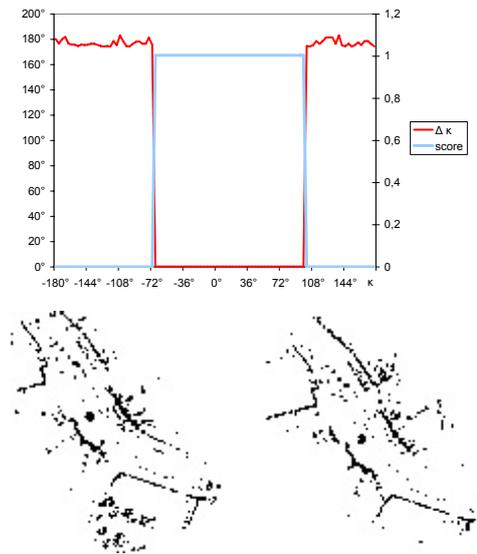


Fig. 11. Convergence (top) for all start orientations  $-180^\circ \leq \kappa < +180^\circ$  for SP 2-SP 3 (bottom left, right), located in a street corridor.

and add successively additional orientations spread over the  $-180^\circ \leq \kappa < +180^\circ$  range until a high score is obtained.

Considering the number of slices, one can see from Figs. 8 and 9 that taking more slices does not generally improve the convergence radius. Adding slices may potentially also add NDTs that do not represent their slices very well, and thus, may actually deteriorate the result. In a few cases, namely SP 5-SP 6, SP 6-SP 7, SP 7-SP 8, higher slices apparently helped to reduce the number of symmetries and thus enlarged the convergence radius (Fig. 9).

Scan	$\Delta X$	$\Delta Y$	$\Delta \kappa$
pairs	[m]	[m]	[°]
1-2	0.102	0.100	0.192
1-3	-0.064	-0.078	0.176
1-4	-0.053	-0.214	0.636
1-5	0.315	0.000	0.179
1-6	0.022	0.410	1.586
1-7	0.524	0.652	-1.140
2-3	0.025	-0.034	-0.001
3-4	-0.015	0.272	-0.341
4-5	-0.022	0.380	-1.806
5-6	0.064	-0.014	0.007
6-7	-0.195	-0.091	0.438
7-8	-0.068	0.119	-0.018
8-9	0.360	0.475	-0.310
9-10	0.659	-0.460	1.016
10-11	0.278	-0.178	-0.300
11-12	0.138	-0.077	-1.455

Table 5

Deviation of the translation and rotation parameters from the reference values for the NDT registration.

The resulting position and orientation errors for all NDT experiments are shown in Table 5. Note that due to the horizontal slicing scheme we adopted, in contrast to Table 4, only two translations and one rotation are determined and all the tilted scans are missing. Comparing the results, we see that the orientation by planar patches was able to still align SP 1-SP 10 which has an overlap of only 16%. In contrast, the NDT fails after SP 1-SP 7, which has a considerably larger overlap of 51%. On the other hand, the distance between SP 1 and SP 7 is 30 m, which is probably anyhow beyond the distance between two scans one would prefer to obtain a dense city scan with few occlusions. Also, the NDT has the advantage of being algorithmically simple and does not rely on the presence of planar structures. Considering accuracy, note that although the NDT leads to slightly worse results, the maximum error is below 1 m in the plane, and 1.8° in orientation, which is below what is required for coarse orientation.

Regarding execution time, one run takes about 30 seconds on a standard

2 GHz Pentium PC (one slice, one start orientation, eight cell sizes with 1,000 iterations each). Using four start orientations, as we proposed, requires about 120 seconds. Using more slices also increases the time required, however only linear in the number of added points rather than in the number of slices. It is obvious that the arbitrarily chosen fixed number of 1,000 iterations per cell size should be replaced by a more elaborate termination scheme. In order to assess by how much the number of iterations can be reduced, we decreased the number of iterations and found that we were still able to align all scans SP 1-SP 2 to SP 1-SP 7 using only 300 iterations. Then, only 30 seconds are required per alignment (one slice, four start orientations).

## 5 Conclusions and further work

In this paper, we compared two different methods for finding the coarse orientation of terrestrial laser scans without usage of artificial markers. For evaluation, we acquired a test scene in downtown Hannover, Germany, with 20 terrestrial scans placed systematically along a trajectory. Reference orientations were obtained by the standard orientation procedure using retroreflective targets and manually assisted target selection.

The first method we investigated follows the classical feature-based object recognition approach. Planar patches are extracted in two scans independently, plane triples are assigned, transformations are computed and scored, and the transformation with the highest score wins. As a first result, we found that although the scene is in a dense built-up area, there are surprisingly few (large) planar patches which can be assigned. In our scene, around 50% of the scan points are not part of a patch or cannot be assigned to a patch in the other scan. A further 40% belong to the ground, so that only 10% are facade patches. Note, however, that 10% still represent approximately 200,000 scan points. For our scene, we observed correct solutions for scan pairs with an overlap as low as 16%. All solutions were quite accurate, with a maximum deviation from the reference of about 20 cm in the plane, 40 cm in height, and  $0.5^\circ$  in the rotation angles.

The second method is a non-symbolic approach based on the normal distributions transform (NDT). Assuming the scanner is set up approximately upright, horizontal slices are cut, and slices from one scene are iteratively matched to the NDT of the slices of the other scene. The iteration is guided by the minimization of a score function. Unlike to what we expected, adding slices did not consistently improve the convergence radius for the rotation angle. Matching of scenes worked down to an overlap of 51% only. The iterative estimation is also susceptible to symmetries in the scene, however the score function reliably indicated the correct solution so that starting with a few orientations

evenly distributed over the full 360° range is considered a feasible approach. Regarding accuracy, the maximum deviation from the reference was around 1 m in the plane, and 1.8° in orientation.

Comparing both methods, the planar patch approach was able to register scans farther apart with a very small overlap. Also, it seems that the NDT alignment is more influenced by scene contents. Moreover, in our slice-based implementation, the NDT can only handle upright scanner setups with constant height. On the other hand, we did not assess how both methods compare when the number of planes in the scene is smaller, in which case the NDT may become superior. Also, the NDT is conceptually simple and fast. Thus, which method is to be preferred depends strongly on the application. Comparing accuracy, the planar patch approach proved to be more accurate. However, it is our opinion that both methods are accurate enough for the intended purpose of coarse orientation.

For the future, several issues are worth investigating. Especially for the planar patch approach, the influence of different (faster) segmentation methods, such as the one by Jiang and Bunke (1992), on the results can be assessed. In addition to planar patches, other primitives can be used. Other properties than the triple vector product can probably be used for the prioritization of the correspondences. And finally, since many scanners include a camera with known relative orientation, image data can be used for the alignment as well (Dold and Brenner, 2006).

## Acknowledgements

This work has been funded by the VolkswagenStiftung, Germany. We thank the anonymous reviewers for their helpful comments.

## References

- Asai, T., Kanbara, M., Yokoya, N., 2005. 3D modeling of outdoor environments by integrating omnidirectional range and color images. In: Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling. Ottawa, Ontario, Canada, pp. 447–454.
- Bae, K.-H., Lichti, D. D., 2004. Automated registration of unorganised point clouds from terrestrial laser scanners. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 35 (Part B5). pp. 222–227.
- Besl, P. J., McKay, N. D., 1992. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (2), 239–256.

- Biber, P., 2003. The normal distributions transform: A new approach to laser scan matching. Tech. Rep. 3, Wilhelm Schickard Institute for Computer Science, Graphical-Interactive Systems (WSI/GRIS), University of Tübingen.
- Chen, Y., Medioni, G., 1991. Object modeling by registration of multiple range images. In: International Conference on Robotics and Automation. Vol. 3. pp. 2724–2729.
- Dold, C., Brenner, C., 2006. Registration of terrestrial laser scanning data using planar patches and image data. In: Maas, H.-G., Schneider, D. (Eds.), The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 5). pp. 78–83.
- Duda, R. O., Hart, P. E., 1973. Pattern Classification and Scene Analysis. John Wiley and Sons, New York.
- Früh, C., Zakhor, A., 2001. Fast 3D model generation in urban environments. In: IEEE Conf. on Multisensor Fusion and Integration for Intelligent Systems, Baden-Baden, Germany. pp. 165–170.
- Grimson, W. E., Lozano-Perez, T., 1984. Model-based recognition and localization from sparse range or tactile data. The International Journal of Robotics Research 3 (3), 3–35.
- Grimson, W. E. L., 1990. Object Recognition by Computer. Series in Artificial Intelligence. MIT Press, Cambridge, Mass.
- Gruen, A., Akca, D., 2005. Least squares 3D surface and curve matching. ISPRS Journal of Photogrammetry and Remote Sensing 59 (3), 151–174.
- Hoover, A., Jean-Baptiste, G., Jiang, X. Y., Flynn, P. J., Bunke, H., Goldgof, D. B., Bowyer, K., Eggert, D. W., Fitzgibbon, A., Fisher, R. B., 1996. An experimental comparison of range image segmentation algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (7), 673–689.
- Horn, B. K. P., 1987. Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America 4 (4), 629–642.
- Jiang, X. Y., Bunke, H., 1992. Fast segmentation of range images into planar regions by scan line grouping. Technical Report IAM-92-006, Institute of Informatics and Applied Mathematics, University of Berne, Switzerland.
- Johnson, A., Hebert, M., May 1997. Surface registration by matching oriented points. In: Proc. International Conference on Recent Advances in 3-D Digital Imaging and Modeling. Ottawa, Ontario, pp. 121–128.
- Rabbani, T., Dijkman, S., van den Heuvel, F., Vosselman, G., 2007. An integrated approach for modelling and global registration of point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 61 (6), 355–370.
- Rusinkiewicz, S., Levoy, M., 2001. Efficient variants of the ICP algorithm. In: Proc. Third Int. Conf. on 3D Digital Imaging and Modeling (3DIM). Quebec, Canada, pp. 142–152.
- Sansò, F., 1973. An exact solution of the roto-translation problem. Photogrammetria 29 (6), 203–216.
- Talaya, J., Alamus, R., Bosch, E., Serra, A., Kornus, W., Baron, A., 2004. Integration of a terrestrial laser scanner with GPS/IMU orientation sensors. In: The International Archives of the Photogrammetry, Remote Sensing and

- Spatial Information Sciences 35 (Part B7). pp. 990–995.
- Thrun, S., Burgard, W., Fox, D., 2005. Probabilistic Robotics. The MIT Press, Cambridge, MA, London, England.
- von Hansen, W., 2006. Robust automatic marker-free registration of terrestrial scan data. In: Förstner, W., Steffen, R. (Eds.), The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3). pp. 105–110.
- Winkelbach, S., Molkenstruck, S., Wahl, F. M., 2006. Low-cost laser range scanner and fast surface registration approach. In: Franke, K., Müller, K.-R., Nickolay, B., Schäfer, R. (Eds.), Pattern Recognition (DAGM 2006), Lecture Notes in Computer Science 4174. Springer, pp. 718–728.