Sparse Registration

3D Reconstruction from Pairs of 2D Line Scans

Benjamin Huhle - Timo Schairer - Sebastian Herholz -
Andreas Schilling - Wolfgang Straßer

ISSN 0946-3852
WSI-2012-02

Wilhelm-Schickard-Institut für Informatik
Visual Computing
Sand 14
D-72076 Tübingen
Tel.: +49 7071 29-75462

Last changes: 10/15/2011 - Published: 10/31/2012

Sparse Registration

3D Reconstruction from Pairs of 2D Line Scans

Benjamin Huhle · Timo Schairer · Sebastian Herholz · Andreas Schilling · Wolfgang Straßer

Abstract We address a new registration problem: Using a coupled pair of 2d scanners, we capture range data by freely moving the system through the scene. The registration with regard to six degrees of freedom becomes solvable due to the fact that first, the pair of line scanners has different orientation, and second, we use a volume-oriented algorithm instead of commonly used surface-oriented approaches. We present a method that is based on the idea of preserving the free space represented in each of the scans. The proposed algorithm is evaluated with real range data associated with orientation estimates from an inertia sensor. Additionally, we provide quantitative results with simulated data. In both cases, the algorithm is capable to recover from large translational and moderate rotational errors in the initial configuration.

Keywords registration · scan matching · 3d acquisition

1 Introduction

Existing solutions to the registration task of 3D point data assume that each of the separate point sets represents a set of surface patches of certain extents. In this paper, we address a novel registration problem, where we are given a set of scans and each of these single scans only comprises sparse range information: We use a combination of two 2D scanners. This means that each of the two scanners captures a linear array of distance values measured along rays that lie in a common plane.

Such 2D laser scanners can work at comparably high frequency and therefore allow for a continuous form of scanning, in contrast to the so-called stop-scan-go pattern which is mostly applied when using laser-based 3D scanners. In this paper, we provide a proof of concept, namely that the registration with regard to six degrees of freedom, i.e., unconstrained motion in 3D, is possible with data from 2D laser range finders.

To overcome the high ambiguity of the pose of single 2D scans, we consider a combination of two scanners with different orientations integrated into a single acquisition device, i.e., the resulting scans are pairwise rigidly coupled. An illustration of the output of such a setup is depicted in Figure 1.

In the following, by the term *scan*, we will refer to such a combined scan pair. In the illustrations, however, we will only show single scans in favor of a better understanding.

As mentioned before, the common registration techniques that work on point data rely implicitly (Iterated Closest Point, ICP [3,4]) or explicitly (Normal Distributions Transform, NDT [12]) on the fact that the captured points locally represent a smooth surface. Surfaces, however, are not well captured in single 2D scans, and therefore, these algorithms are not directly applicable on the described type of data. The underlying idea of the novel registration algorithm proposed here is to enforce the preservation of empty space. This means that we register the scans by avoiding or resolving all mutual intrusions.

If we assume that the scans are densely sampled, then each line segment connecting two neighboring scan points together with the origin of the scan spans a triangle of free space. Two line segments of a pair of correctly registered scans may intersect. However, if a line segment of one scan intersects the triangle of free space of another one, this indicates an incorrect registration

WSI/VC
Universität Tübingen
Sand 14
72076 Tübingen
Germany
E-mail: schilling@uni-tuebingen.de

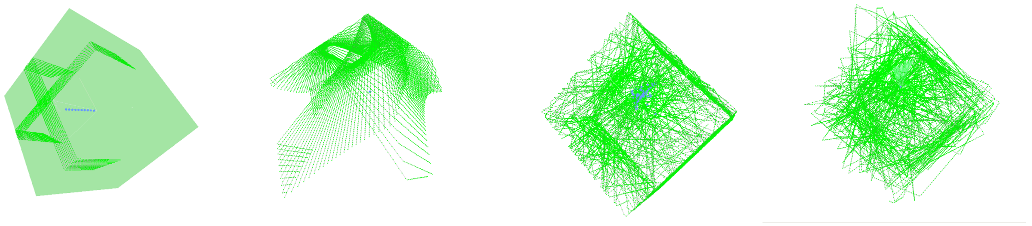


Fig. 1 Illustration of scans acquired with the proposed setup (from left to right): **1**. Registered sequence of scans of a cube model with linear motion. Scan origins are depicted in blue. **2**. dito, yet rotating during acquisition. **3**. Sequence of 100 scans, acquired with the B-Spline motion model (cf. Section 6.2). **4**. dito, with position information discarded.

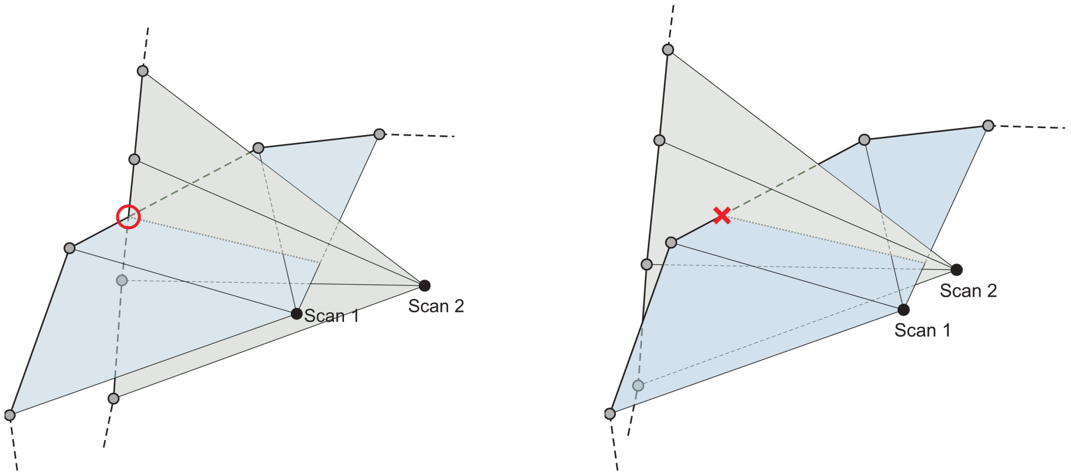


Fig. 2 If we assume that triangles spanned by two neighboring scan points and the origin of the scan represent free space, then scans that do not comply with this rule (as in the right subplot) indicate an incorrect registration. The proposed algorithm tries to resolve such intersections to achieve a configuration as depicted in the left subplot. (Note that, for simplicity, the illustration only shows single scans instead of the coulpled pairs used by the system.)

of the scans (cf. Figure 2). By means of an iterative algorithm, we register the scans by applying rigid-body transformations that resolve these intersections.

2 Related Techniques

The registration of 3D point data with regard to six degrees of freedom is a well studied topic. The iterated closest point algorithm (ICP, [3,4,9]) is a very popular method that works well if the point clouds have significant overlap or if the initial pose estimate is close to the solution. An alternative that also works on 3D point data, is the normal distributions transform (NDT, [12]) which performs comparably [20]. The acquisition of 3D models with these techniques, using a stop-scan-go pattern of the sensor platform, was studied successfully, e.g., in [10,13,18]. In the last few years, approaches to a more continuous form of 3D scanning ("scanning while moving") were taken. Cole and Newman [17] present a system that captures 3D data continuously using odometry information. They segment the data stream into chunks of smaller 3D point clouds and register these in order to improve the measured transformations. Since they assume horizontal movements exclusively, they use an additional 2D SLAM technique. Harrison and Newman [19] presented another approach whose idea is to correct the transformations such that planes that are vertical in the real world are enforced to be exactly vertical in the model. Stoyanov and Lilienthal [21] applied loop closure using ICP and a multi-level relaxation scheme to 3D scans acquired with a rotating 2D laser scanner.

Very good registration results have also been achieved by using feature points with descriptors based on 3D geometry (e.g. [6,15]) or with descriptors derived from additionally available image data (e.g. [14,22]).

In contrast to these works, we directly register single 2D scan lines and use a volumetric approach similar to the idea of space carving [8] in image based modeling. With this approach, we permit full six degrees of freedom motions as opposed to systems in the spirit of Früh and Zakhor [11], where the acquisition device only moves in the horizontal plane. A volumetric approach was also used by Tubić et al. [1] who proposed an integrated solution to the registration and surface reconstruction problem, yet relying on dense range images, in contrast to single 2D scans as in our case.

3 Preservation of Free Space

As indicated in the introduction, we generalize the idea that the space along the ray of a range measurement is not occupied by any object in the scene. Consequently, intersections of a scan line segment of an *intruder scan* with triangles spanned by a line segment and the origin of a *reference scan* indicate an inexact registration. This assumption holds true as long as the local sampling density is high, i.e., as long as the scan line represents a line on the captured surface. In this case, the proposed algorithm tries to resolve all intersections by correcting the relative poses of the scans.

In order to reduce the computational complexity, the scans are simplified in a preprocessing step. We apply the Douglas-Peucker algorithm [2] that approximates the scan by a subset of the points with a specified maximum error, which can be set according to the noise level of the range values.

Iteratively, the intersecting reference and intruder line segments are determined and the intrusions are minimized. The minimization problem can either be solved by a variational formulation or with a more illustrative physically motivated approach. We use a mass-spring model where the springs are in state of rest when the intruding line segment intersects with the line segment of the reference scan, i.e., with one boundary of the reference triangle.

Note that, in general, the problem is only solvable offline in batch-mode, also called *n-scan matching*, since smaller subsets of the scans (such as used in an incremental registration) do not provide enough spatial constraints for a robust registration and a sufficient sampling of the scene is necessary to align 2D scan lines. However, we believe that it is also possible to use this method to improve existing pre-registered models, which could be investigated in future research.

4 The Mass-Spring Model

In the following, we describe some extensions to the above model that are well allegeable from the rigid body dynamics point of view. For simplification, we omit the units of measurement in the following equations and assume respective scene units. A well comprehensible introduction to the simulation of rigid body dynamics can be found in [5].

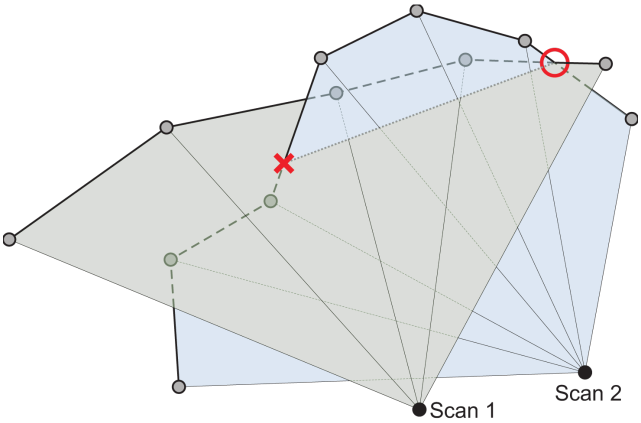


Fig. 3 A scan pair that intersects twice. Here, a global nearest neighbor search is misleading. The intersection marked with "X" is never resolved since a segment pair with smaller distance (marked with a circle) is always found.

4.1 Data Association

At a single intersection, it is necessary to not only consider the involved reference and intruder line segment but to search for a nearest neighbor pair of line segments on the reference and intruder scan. Attaching a spring to these two closest line segments in the model means that we effectively minimize their distance. This way, the optimization tends to be less local which is mandatory when the current configuration is not yet close to the correct solution. However, the radius of the nearest neighbor search around the intersection needs to be restricted when the expected remaining error becomes smaller. Otherwise, it could happen that out of two intersections of a scan pair, only one is effectively resolved (cf. Figure 3). We choose to couple the search radius $r_{NN}^{(t)}$ at time step t to the maximum of the forces (see below) from the last iteration:

$$r_{NN}^{(t)} = 2\,\|F_{max}^{(t-1)}\|_2 . \tag{1}$$

4.2 Sampling Density

In cases where the sampling density is rather low, or where parts of the scene are occluded in one scan, intersections may also occur when the scans are perfectly registered (cf. Figure 4). We therefore scale the spring rate

$$\kappa = \exp\left(-\left[\frac{2\,\theta}{3\,\pi} - \frac{1}{3}\right]^2\right) \tag{2}$$

with regard to the scan angle θ of the intruder scan (cf. Figure 5). This results in a full application of the force of the spring in cases where the sampling density

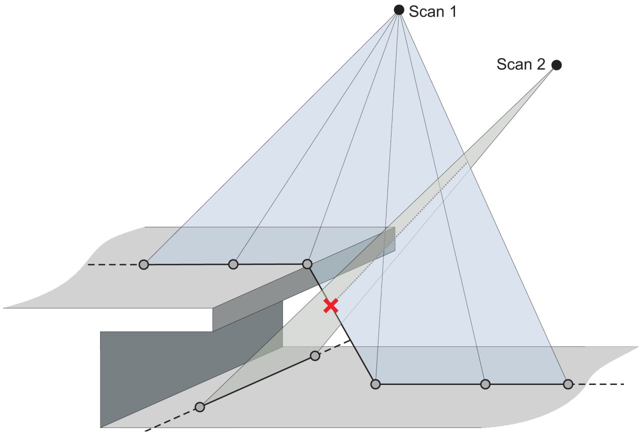


Fig. 4 Intersections may occur not only due to misalignment, but also due to occlusions and undersampling.

is maximal ($\theta = \pi/2$) and a force close to zero where the scan ray hits the surface in an acute angle ($\theta \approx 0$). Thus, we obtain the force $f = \kappa\, d$ that acts on both scans, where the vector d denotes the distance of the line segments.

4.3 Elastic Collisions

Each spring effects the reference and the intruder scan with the same force. Assuming that the scans have a certain weight, we can determine which scan should be effected to a greater or lesser extent — equivalent to an elastic collision. We equip those scans that already fit well into the 3D model with a great mass. A good fit is indicated by many intersections with other scans that only induce small forces. In each iteration, we compute a notional mass that linearly depends on the mean of the squared spring forces. Let $f_i^{(\ell)}$ denote the force on scan i which originates from intersection ℓ and let $\mathcal{I}_i$ denote the set of intersections in which the scan i is involved as either intruder or reference. For the corresponding mass, we obtain

$$m_i = \frac{|\mathcal{I}_i|}{\sum_{\ell \in \mathcal{I}_i} \|f_i^{(\ell)}\|_2^2}. \tag{3}$$

Since the spring forces also produce a rotational movement, the distribution of the mass needs to be modeled as well for each scan. We choose to equally distribute the mass among the scan points, which leads to an inertia tensor I that has constant structure in the coordinate frame of the scan and only scales with the actual mass.

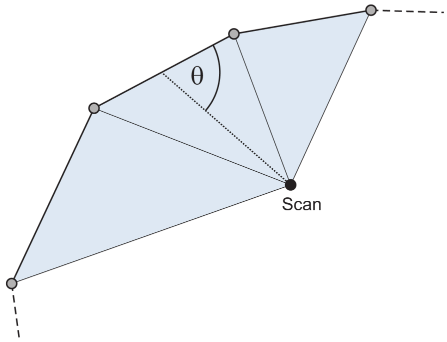


Fig. 5 Scan angle θ. A small value indicates low sampling density and potential occlusions.

4.4 Clustering Forces

For a homogeneously sampled scene, it would be correct to sum up the forces $f_i^{(\ell)}$, $\ell = 1, \ldots, n$, of all springs which are connected to a scan i in order to compute the overall force acting on this scan. Since we are dealing with imperfectly sampled scenes, we need to decouple the simulation from the sampling process that produced the scans. From Figure 6, it can be observed that the force caused by a certain registration error scales with the number of intersections. We compensate for these repeated collisions by clustering the forces with regard to their directions. For a set of cluster representatives c_k, $k = 1, \ldots, m$, we scale the contribution to a cluster by the length $p_{i,k}^{(\ell)}$ of the projection

$$p_{i,k}^{(\ell)} = c_k^\top f_i^{(\ell)}, \tag{4}$$

and obtain an overall force

$$F_i = \sum_{k=1}^{m} \frac{1}{Z_{i,k}} \left(\sum_{\ell \in \mathcal{I}_i} w_{i,k}^{(\ell)}\, p_{i,k}^{(\ell)} \right) c_k, \tag{5}$$

acting on scan i, with weights

$$w_{i,k}^{(\ell)} = \left| \frac{1}{\|f_i^{(\ell)}\|_2} p_{i,k}^{(\ell)} \right| \tag{6}$$

and normalizer

$$Z_{i,k} = \sum_{\ell \in \mathcal{I}_i} w_{i,k}^{(\ell)}. \tag{7}$$

Analogously, we compute the overall torque as

$$\tau_i = \sum_{k=1}^{m} \frac{1}{Z_{i,k}} \sum_{\ell \in \mathcal{I}_i} w_{i,k}^{(\ell)}\, p_{i,k}^{(\ell)} \left(c_k \times r_i^{(\ell)} \right), \tag{8}$$

where $r_i^{(\ell)}$ is the vector from the barycenter of the scan to the intersection point of the line segment and the reference triangle.

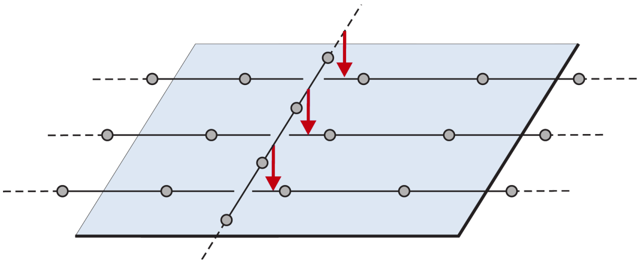


Fig. 6 Coupling of sampling and simulation: Depending on the local sampling density, a constant registration offset induces a different number of instances of the same force.

4.5 Regularization

In order to avoid local minima and to improve the robustness of the method, we incorporate additional springs into the system, i.e., we apply external regularization forces attached to the scan. Let s_i and q_i denote the origin and the orientation (in quaternion representation) of scan i. From its neighbors in time $i-$ and $i+$, and based on the original orientation measurement $\tilde{q}_i$, we compute a force

$$F_i^{reg} = \kappa_{reg} \left[\frac{1}{2} \left(s_{i-1} + s_{i+1} \right) - s_i \right] , \tag{9}$$

and a torque

$$\tau_i^{reg} = \kappa_{reg} \; I \; \omega_i , \tag{10}$$

that are applied to each scan i. The angular velocity ω_i is computed as the axis-angle representation of the deviation from the orientation measurement $q_\Delta = q_i^{-1} \otimes \tilde{q}_i$, where $\otimes$ denotes quaternion multiplication.

Both for the linear and angular regularization, we use a fixed spring rate starting at $\kappa_{reg} = 1.0$ and we anneal the system by lowering its value whenever the system meets a convergence criterion.

5 Solving the System

We solve the mass-spring model with an explicit Euler method using a large damping constant. The resulting solver is very similar to a steepest gradient descent of a variational least squares formulation of the problem. We experienced that a full minimization of the intrusions in one iteration of the algorithm is not preferable. This is due to the fact that the minimization problem very often changes from one iteration to the next one since different intersections occur. Apart from that, for the registration problem, it is necessary that a great percentage of the scan pairs intersect. Methods that exploit the sparseness of a problem are therefore not applicable and a full minimization in each iteration would be very costly. Therefore, it is reasonable to only perform small updates on the poses of the scans — as it is done with the explicit Euler method — before evaluating the intersections once again.

A summary of the iterative algorithm is given in Table 1.

```
1. detect intrusions
    - test all scans pairwise
2. search nearest neighbor line segments
    - distances represent the forces via $f = \kappa\, d$
3. compute the scan masses
    - cf. Equation 3
4. apply forces
    - partition onto both scans (elastic collision)
    - accumulate & normalize forces for each scan (Eq. 5)
    - add regularization (Eqs. 9, 10)
    - perform one Euler step
goto 1.
```

Table 1 Brief summary of the registration algorithm.

5.1 Accelerating the Registration

As mentioned earlier, at a single intersection, we not only take into account the involved reference and intruder line segments but search for a nearest neighbor pair of line segments of the reference and the intruder scan. Considering the large number of intersections that can occur at each time step, the potential complexity is in $\mathcal{O}\left(m^2 n^2\right)$ for a scene consisting of m scans of which each scan comprises n line segments.

Due to the computational load of the algorithm, we compute the nearest neighbor search and the distances of the line segments on the GPU using *CUDA*: One could perform partitioning of the input data, i.e., assign a single scan to each task and compute the corresponding intersections. This type of decomposition, however, introduces synchronization issues since a single scan not only potentially intersects many other scans, but it can also intersect other scans several times. We therefore perform partitioning on the output data, i.e., with regard to intersections: In a first step, the intersections are computed on the CPU efficiently by applying various culling techniques. Searching for the nearest neighbor in a brute-force implementation on the GPU, however, still is two orders of magnitude faster than a CPU-based implementation that makes use of diverse hierarchical data structures.

5.2 Parallel Implementation

During initialization, scan IDs, initial scan poses, and range data in the local coordinate frame of the respective scanner are uploaded to the GPU.

At every iteration, intersections of scans are computed on the CPU and uploaded to the GPU. Accordingly, a set of reference and intruder scans is provided for each *CUDA* kernel that transforms the local range data to the world coordinate frame using the current pose information. Each kernel then loops over the line segments of the reference and intruder scans to compute the pair of line segments with minimum distance while keeping to the search radius r_{NN}. After a pair of line segments has been identified, the distance vector between the segments, i.e., the force vector f, is calculated along with the intersection point in world space.

After each iteration, the indices of the nearest neighbor line segments, the corresponding forces, and the intersection points are downloaded from the GPU.

6 The Data

We show results of the registration algorithm applied to real data that we acquired using a prototype system. Here, no ground truth is available, but the performance can be evaluated by qualitative inspection of the resulting models.

For a quantitative investigation of the presented registration problem, we use simulated data for additional experiments. In this case, perfect ground truth data is available for quantifiable results and comparisons. Furthermore, we do not have to deal with time synchronization issues between the sensors.

The sensor model and the motion model, i.e., assumptions on how the sensors are moved through the scene while recording, are the same for both real and simulated data: We assume motion with moderate speed and 2D scanners working at high frequency, such that the ego-motion during the acquisition of single 2D scans can be neglected. Further, we expect that an initial guess of the orientation of each scan is available.

6.1 Real Data

To acquire real data, we use a prototype system comprising two *SICK LMS200* scanners, an *XSense* inertia sensor and a *Matrix BlueFox* color camera with fish-eye lens. This setup is depicted in Figure 7. During acquisition, the device is moved and rotated smoothly through the scene by hand. Registration is performed using the laser-based range data and the orientation readings of the inertia sensor as initial rotation estimates. In order to colorize the models, we calibrated the camera using the *OCamCalib Toolbox* [16] and project the 3D points into the camera image for a bilinear color lookup.

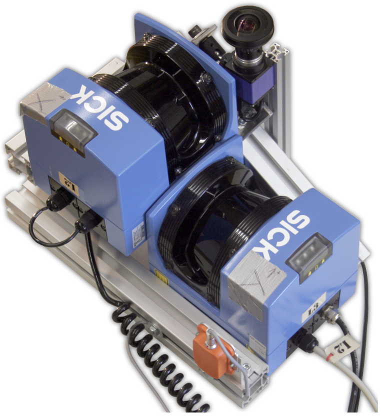


Fig. 7 Prototype system of the proposed setup consisting of two 2D laser scanners and an XSense inertia sensor. The prototype additionally includes a color camera with fisheye lens to capture texture information.

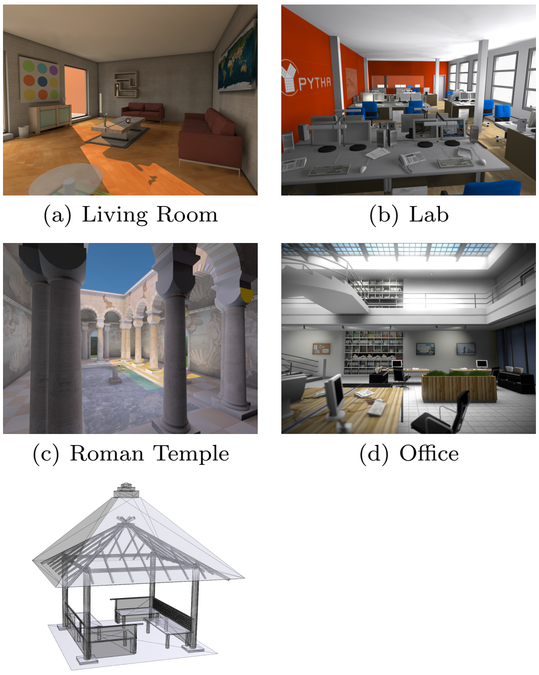
6.2 Simulating the Acquisition

We tried to simulate the acquisition as realistically as possible in order not to change the problem itself. Therefore, we employ a sensor model that reproduces different real world error sources of the scanning process.

The registration process is started with an initial guess of the orientation for each scan. Accordingly, we apply random noise to the ground truth orientation (uniformly distributed axes and normally distributed angles).

The acquisition of the range data is simulated by computing scan ray distances to the mesh-based 3D test models. To account for the systematic and the statistic errors of the sensor, in direction as well as in range, we add Gaussian noise to the depth values ($\sigma = 1.0cm$). Furthermore, we simulate the effect of *flying pixels*, that occurs whenever the spot of the scan ray covers surfaces at different distances as it happens on edges of objects. We use Monte Carlo integration with 50 samples of the scan cone and restrict the integration area to the length of the measurement impulse. The maximum range of the scanner is taken into account, i.e., measurements where less than half of the samples hit a surface of the model, are marked as *max range* readings. The choice of parameters is exemplarily based on the specifications of the *SICK LMS200* range finder [23, 24].

The linear motion of the sensor is modeled with a B-Spline curve that interpolates randomly chosen con-



(a) Living Room

(b) Lab

(c) Roman Temple

(d) Office

(e) Kyoto Shelter

Fig. 8 Virtual scenes used for the simulation. Models (a)–(d) courtesy of PYTHA Lab GmbH, model (e) of www.3dchaya.com.

trol points in 3D. For the rotation of the sensor, we use the same interpolation applied in the exponential chart [7], i.e., the axis-angle representation, relative to the first control point of the respective interval. The control points of the quaternion curves represent random orientations, i.e., two consecutive control points may have completely different orientation.

7 Experiments

We present first results using a real prototype system, and we also tested our algorithm on various synthetic data sets. Table 2 lists the specifications of the 3D models as well as the simulated scan data, and Figure 8 gives an impression of the scenes.

7.1 Settings

In all experiments using simulated data, the registration starts with orientation values that are obtained by perturbing the correct orientation by noise with standard deviation $\sigma = 3°$. The position information is completely discarded, i.e., all scans are reset to the scene origin. To account for this disparity of the errors, we decrease the torque in the mass spring model by an additional factor of 0.3 with regard to the translational step size.

For the normalization (Eq. 5), we achieved the best results when using an arbitrarily oriented basis of three orthogonal cluster representatives.

The annealing scheme applied to the regularization leads to increased run times of the algorithm, yet contributes considerably to its robustness. We use the non-monotony of the sum of squared forces $\sum_i \|F_i\|_2^2$ as convergence criterion and, each time it is met, we decrease the spring rate by a factor of 1.2.

The simulated acquisition processes in the tests are of two different types: For the first one (*Temple 1, Kyoto Shelter*), we used smooth B-Spline curves for the trajectories as well as for the course of the orientation by choosing a low ratio of control points to number of scans. In the second mode (*Living Room, Lab, Temple 2, Office*), the speed of the motion and of the change of orientation varies drastically due to a high number of control points. We run this second simulation in order to validate that a successful registration is also possible when the underlying assumptions of the regularization and the actual motion differ significantly.

7.2 Results using Real Data

Results of the registration algorithm using our real prototype system are depicted in Figures 9 and 10. Note that the scan masses (cf. Eq. 3) provide a good indication of how likely a scan can be considered an outlier. Here, we removed the 10 scans with lowest weight.

7.3 Results using Virtual Data

The quantitative reconstruction results are shown in Table 3 and different views of the resulting point clouds are presented in Figure 11.

As a measure for the quality of the reconstruction, one can look at the remaining offset of the scan origin and its orientation. In certain configurations, however, there is no unique correct pose for the scan, e.g., it can be translated along a certain direction without altering the model quality. Additionally, offsets in position and orientation can compensate each other to some degree. Therefore, in Table 3, we present the average and the maximum point-to-surface distance (PSD), i.e., the distance of each scan point to the nearest triangle of the mesh. Note that noise and flying pixels lead to a non-zero PSD also for the perfectly registered ground truth models.

Additionally, the sum of squared distances (SSD) of the reconstructed scanner positions is provided since

Name	**# vertices**	**dimensions [m]**	**# scans**	**# control points**
Living Room	72,791	8 × 7.5 × 3.2	300	75
Lab	125,805	20.9 × 6.5 × 3.2	300	75
Roman Temple 1	190,227	21.5 × 15 × 9.5	300	30
Roman Temple 2			300	75
Office	269,039	25.5 × 16.8 × 8.6	300	75
Kyoto Shelter	1,547	4.5 × 4.3 × 4.5	200	20

Table 2 Simulated datasets.

		initial		**ground truth**	**reconstructed**	
Name	# iter.	SSD	PSD	PSD	PSD	SSD
Living Room	3289	58.30	0.16 (max 1.09)	0.005 (max 0.17)	0.011 (max 0.22)	0.54
Lab	3123	51.00	0.16 (max 1.46)	0.004 (max 0.29)	0.012 (max 0.53)	0.89
Temple 1	7939	205.5	0.31 (max 1.99)	0.004 (max 0.12)	0.007 (max 0.56)	0.07
Temple 2	8162	45.79	0.15 (max 1.74)	0.004 (max 0.11)	0.013 (max 1.64)	0.87
Office	4364	58.30	0.16 (max 1.77)	0.004 (max 0.18)	0.022 (max 2.48)	0.39
Kyoto Shelter	2166	25.24	0.08 (max 0.62)	0.004 (max 0.16)	0.006 (max 0.21)	0.08

Table 3 Registration results: The barycenter and the average orientation of the scans of the reconstruction and initial configuration are aligned to ground truth before evaluation.

based on the PSD, large offsets may be underestimated. This is due to the fact that the association of a point and the originally captured surface is not taken into account.

Especially, the *Temple* and the *Kyoto Shelter* scenes were chosen as very challenging scenes, yet both are registered precisely. In the first one, the occlusions from the columns are obviously well handled by scaling the spring rate with regard to the scan angle (cf. Eq. 2). In the latter one, the simulated laser range finder returns a large amount of invalid range readings, since a major part of the shelter is open. However, the model is successfully reconstructed with a precision close to the original noise level.

Obviously, the robustness of the registration depends on the coverage of the scans. Invariably, it can happen that a few scans (often < 10) are registered less exact and contribute heavily to the overall error. This can occur due to insufficient constraints imposed by the surrounding scans, especially when the sampling in the acquisition process is not dense enough. Or, depending on the structure of the scene, scans get stuck in local minima during optimization. Therefore, it should be helpful to integrate an outlier detection step of some sort and to discard single perturbing scans during registration. As additional experiments with manually removed outliers have shown, this can lead to a significantly improved overall performance, while loosing only a marginal amount of information about the scene.

8 Conclusion

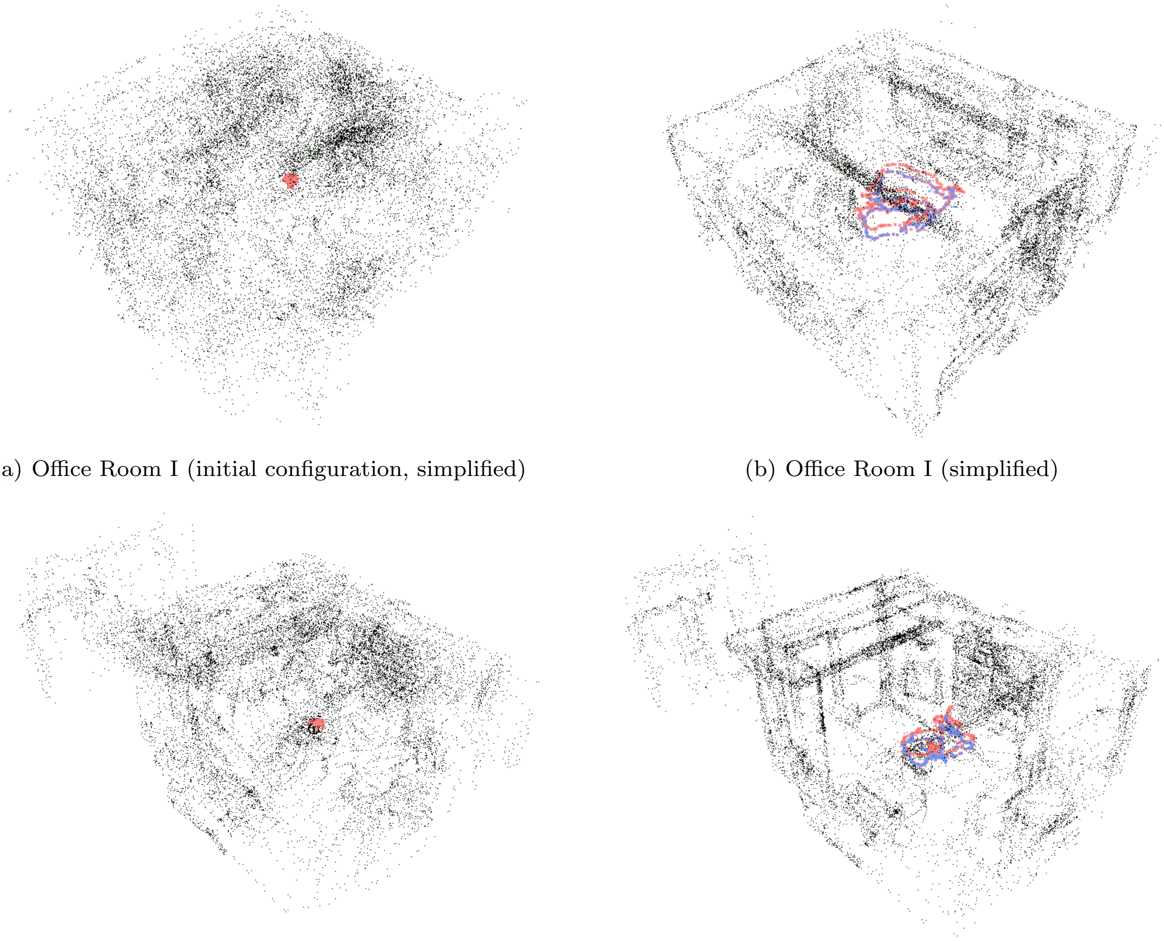
We presented a novel algorithm that allows to register single pairs of 2D scans with regard to six degrees of freedom. The volume-oriented method resolves mutual intrusions of the scans, in contrast to popular algorithms, like ICP or NDT, that maximize the congruence of the captured surfaces. The computationally intensive algorithm is feasible due to the increased performance of modern hardware and due to a fast GPU-based implementation.

We present a first proof of concept with promising results. We apply our method to data captured from real scenes and show that the proposed algorithm is capable of reconstructing correctly registered models. Additionally, we modeled the acquisition process of depth data to validate our approach based on simulated data where ground truth data is available. We present quantitative results which show that the registration is solvable with fair accuracy.

Acknowledgements We thank Philipp Jenke for some very fruitful discussions. The authors gratefully acknowledge financial support by the German Research Foundation (DFG).

References

1. Dragan Tubić and Patrick Hébert and Denis Laurendeau: A volumetric approach for interactive 3d modeling. Int. Journal of Computer Vision and Image Understanding
2. David Douglas and Thomas Peucker: Algorithms for the reduction of the number of points required to represent a



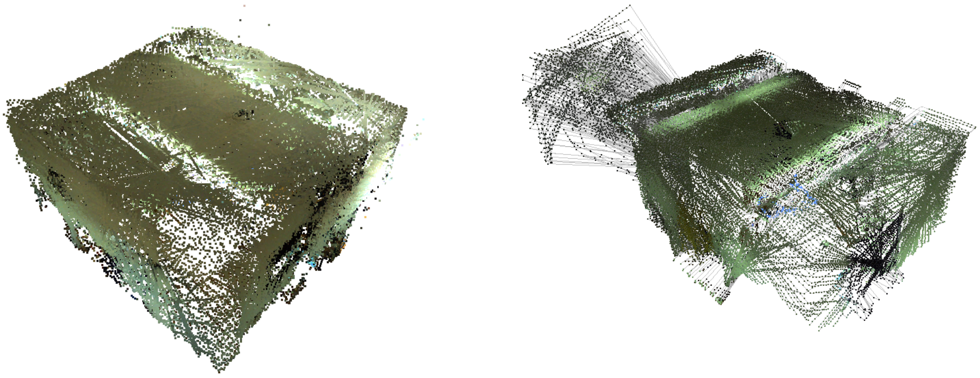
(a) Office Room I (initial configuration, simplified)

(b) Office Room I (simplified)

(c) Office Room II (initial configuration)

(d) Office Room II (simplified)

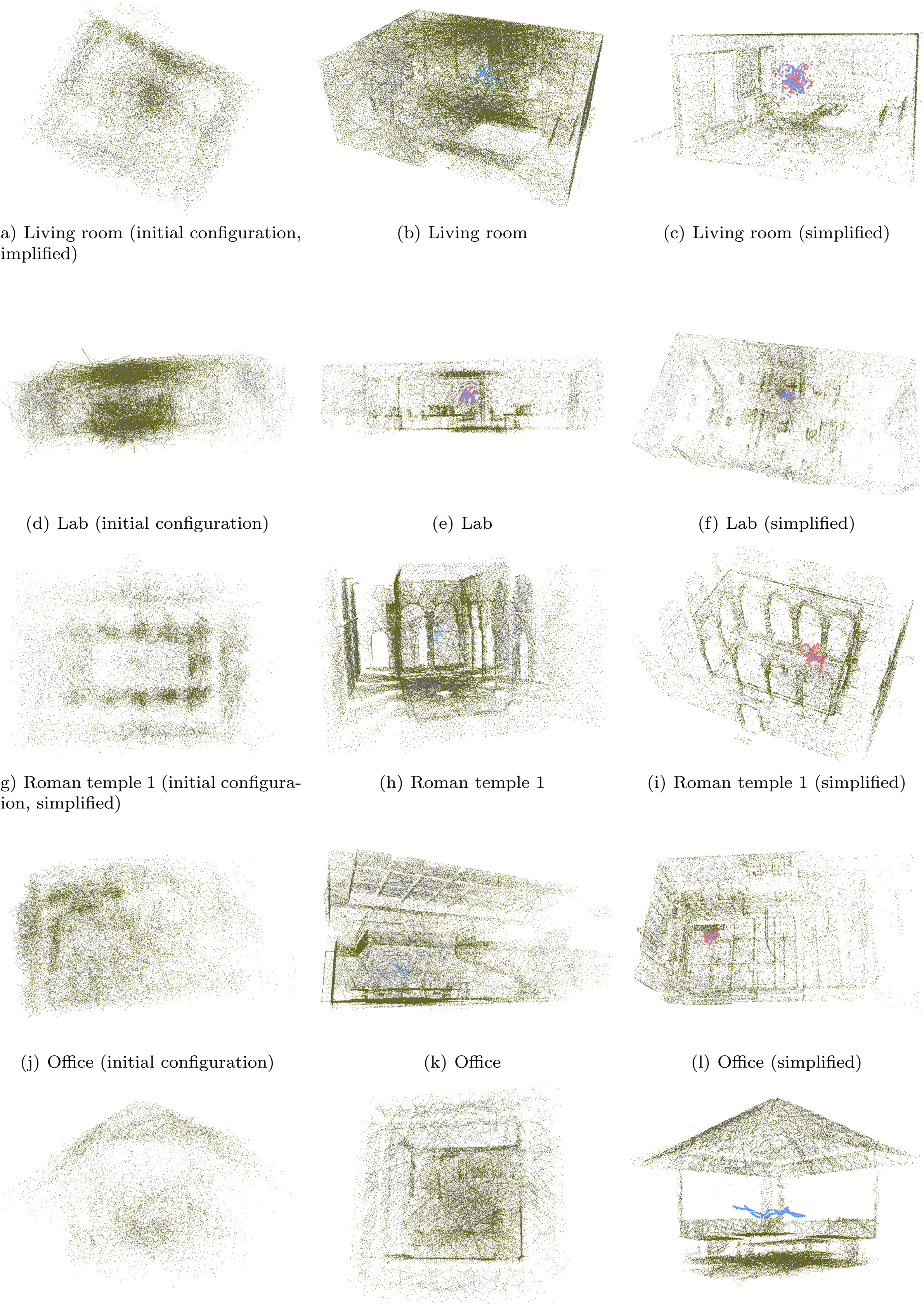
Fig. 9 Reconstructed point clouds from real range data. Blue and red dots indicate the reconstructed scan origins. The *simplified* version of the point clouds is the same as is used for the registration.



(a) Office Room I (textured)

(b) Office Room II (textured)

Fig. 10 Reconstructed point clouds from real range data, textured.



(a) Living room (initial configuration, simplified)

(b) Living room

(c) Living room (simplified)

(d) Lab (initial configuration)

(e) Lab

(f) Lab (simplified)

(g) Roman temple 1 (initial configuration, simplified)

(h) Roman temple 1

(i) Roman temple 1 (simplified)

(j) Office (initial configuration)

(k) Office

(l) Office (simplified)

(m) Kyoto shelter (initial configuration, simplified)

(n) Kyoto shelter

(o) Kyoto shelter

Fig. 11 Reconstructed point clouds from simulated range data. Blue and red dots indicate the reconstructed scan origins. The *simplified* version of the point clouds is the same as is used for the registration.

digitized line or its caricature. The Canadian Cartographer **10**, 112–122 (1973)
3. Paul J. Besl and Neil D. McKay: A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **14**(2), 239–256 (1992). DOI http://dx.doi.org/10.1109/34.121791
4. Yang Chen and Gérard Medioni: Object modelling by registration of multiple range images. Image and Vision Computing **10**(3), 145–155 (1992)
5. David Baraff: An introduction to physically based modeling: Rigid body simulation i – unconstrained rigid body dynamics. In: SIGGRAPH Course Notes (1997)
6. Andrew Edie Johnson: Spin images: A representation for 3-D surface matching. Ph.D. thesis, Carnegie Mellon University (1997)
7. X. Pennec: Computing the mean of geometric features – application to the mean rotation. Tech. rep., Institut National de Recherche en Informatique et en Automatique (1998)
8. Kiriakos N. Kutulakos and Steven M. Seitz: A Theory of Shape by Space Carving. International Journal of Computer Vision **38**(3), 199–218 (2000)
9. Szymon Marek Rusinkiewicz: Efficient variants of the ICP algorithm. pp., 145–152 (2001)
10. Hartmut Surmann and Andreas Nüchter and Joachim Hertzberg: An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. Robotics and Autonomous Systems **45**, 181–198 (2003)
11. Christian Früh and Avideh Zakhor: Constructing 3D City Models by Merging Ground-Based and Airborne Views. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2003)
12. Peter Biber and Wolfgang Straßer: The Normal Distributions Transform: A New Approach to Laser Scan Matching. In: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2003)
13. Andreas Nüchter and Hartmut Surmann and Kai Lingemann and Joachim Hertzberg and Sebastian Thrun: 6D SLAM with Application in Autonomous Mine Mapping. pp., 1998–2003 (2004)
14. G. H. Bendels and P. Degener and R. Wahl and M. Körtgen and R. Klein: Image-Based Registration of 3D-Range Data Using Feature Surface Elements. In: Proc. Int. Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST) (2004)
15. Natasha Gelfand and Niloy J. Mitra and Leonidas J. Guibas and Helmut Pottmann: Robust global registration. In: Symposium on Geometry Processing (SGP) (2005)
16. D. Scaramuzza and A. Martinelli and R. Siegwart: A Toolbox for Easy Calibrating Omnidirectional Cameras. In: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) (2006)
17. David M. Cole and Paul M. Newman: Using Laser Range Data for 3D SLAM in Outdoor Environments. In: Proc. of IEEE International Conference on Robotics and Automation (ICRA) (2006)
18. Magnusson, Martin and Duckett, Tom and Lilienthal, Achim J.: 3d scan registration for autonomous mining vehicles. Journal of Field Robotics **24**(10), 803–827 (2007)
19. Alastair Harrison and Paul Newman: High Quality 3D Laser Ranging Under General Vehicle Motion. In: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA). Pasadena, California (2008)
20. Martin Magnusson and Andreas Nüchter and Christopher Lörken and Achim J. Lilienthal and Joachim Hertzberg: Evaluation of 3D Registration Reliability and Speed A Comparison of ICP and NDT. In: Proc. of IEEE International Conference on Robotics and Automation (ICRA) (2009)
21. Todor Stoyanov and Achim J. Lilienthal: Maximum likelihood point cloud acquisition from a mobile platform. In: Proc. Int. Conf. on Advanced Robotics (ICAR) (2009)
22. Henrik Andreasson and Achim J. Lilienthal: 6D Scan Registration using Depth-Interpolated Local Image Features. Robotics and Autonomous Systems **58**(2) (2010)
23. SICK AG, Waldkirch, Germany: Technical Description (2006)
24. SICK AG, Waldkirch, Germany: Telegram Listing (2006)