Creating 3D Models of Buildings by Car-Mounted LIDAR

Dmitry Chetverikov and Iván Eichhard

Institute for Computer Science and Control, Budapest, Hungary
Eötvös Loránd University, Budapest, Hungary

Abstract

Automatic reconstruction of large-scale outdoor objects like house facades is an important component of mixed-reality systems that model and visualise real world at different level of detail. The authors are involved in a project that utilises a car-mounted LIDAR to acquire a sequence 3D point clouds representing facades in a street. No GPS or IMU is used. Hundreds of point clouds need to be automatically aligned to obtain a realistic surface model of facades. In this paper, we present and compare two solutions to this complex registration problem. Our methods are based on two different, widely used techniques for registering two partially overlapping point clouds in presence of outliers. The proposed algorithms are capable of automatically detecting occasional misalignments. We analyse the operation of the algorithms paying special attention to the robustness, speed and optimal parameter setting.

1. Introduction

The Integrated 4D (i4D) project [1] by Institute for Computer Science and Control (MTA SZTAKI) aims at reconstructing, editing and visualising dynamic real-world scenes at varying level of detail and by fusing different kinds of input data. An important ingredient of such mixed-reality systems is the unit that builds 3D models of large-scale environments and scenes.

We contributed to the project by developing and testing two methods for automatic alignment of a long sequence of measured 3D point clouds acquired by a car-mounted LIDAR device carried along a street. (LIDAR stands for Laser Imaging, Detection and Ranging.) The complete aligned point set represents the street facades. Good alignment results in a high-quality surface model that can be efficiently textured by facade images taken simultaneously or separately.

Point cloud sequence alignment is a popular task in field robotics, remote sensing, spatial information sciences and computer vision. This problem is also addressed in urban area reconstruction [9] when the input data is often provided by airborne or car-mounted LIDAR devices. Compared to image-based reconstruction, reconstruction from LIDAR measurements is more robust to changing illumination conditions and lack of surface texture.

Aerial LIDAR data is used to reconstruct complete residential areas [15], while the car-mounted devices are usually applied to smaller scenes such as houses in streets and squares. Depending on the conditions of measurement, auxiliary sensors such as GPS (Global Positioning System) or IMU (Inertial Measurement Unit) can be used to support the alignment.

Many algorithms have been developed to solve the problem of automatic *registration of two point sets*. Some of them have been successfully used to align a long sequence of point clouds. One of such algorithms is the Normal Distributions Transform (NDT) registration algorithm [7]. Its efficiency and accuracy were studied in [14]. Another popular tool for registration of two point sets is the Iterative Closest Point algorithm and its variants [2,13] whose performance was investigated in [10].

The study [8] evaluates and compares ICP and NDT on LIDAR data. Since the original version of ICP is not applicable to partially overlapping point sets, the authors use a robustified version of the algorithm. This version imposes an upper limit on the distance between the points that can be matched: larger distances are simply discarded. In practice, such limit is hard to set, especially when there is a relatively large rotation between scans, or when point density varies significantly within a scan. In our LIDAR application, the rotation is limited, but point density variation can be large, which would make such ICP-variant impractical.

The Trimmed Iterative Closest Point (TrICP) registration

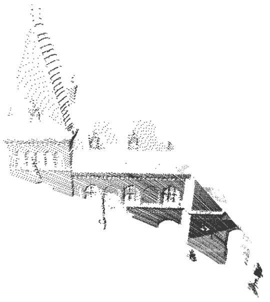


Figure 1: *A sample point cloud.*

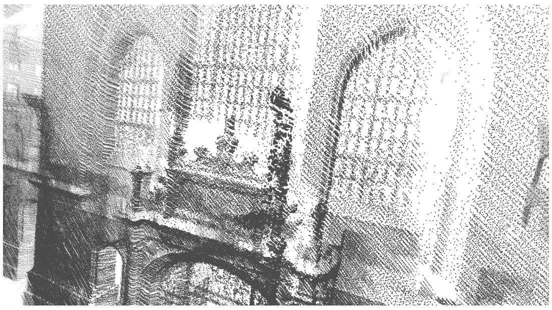


Figure 2: *A part of complete aligned point cloud.*

algorithm [2] is robust and free of such limitations. The comparative study [10] tests it as the standard, most widely applied robustification of ICP. TrICP has been successfully used to register different kinds of measured spatial data.

The main motivation of our study was to compare the two efficent point set registration techniques, NDT and TrICP, in the context of our application. In this paper, we present a point cloud sequence alignment method based on NDT and a method based on TrICP. The major contributions of the paper are the two methods and a discussion of our experience gained while applying the methods to real LIDAR sequences obtained for different streets and buildings.

The layout of the paper is as follows. In section 2, we describe the data acquisition procedure and introduce the alignment problem. Sections 3 and 4 are devoted to the proposed methods. Test results are presented in section 5, followed by discussion and conclusion in the final section 6.

2. Data Acquisition and the Alignment Problem

The 3D data is supplied by the Velodyne HDL-64E RMB-LIDAR device mounted on a car travelling along a street. The rotating multi-beam LIDAR device records 360°-view angle range data sequences of irregular clouds of unoriented points. Due to the intrinsic anisotropy of the data acquisition process, point density decreases with altitude and distance, which makes scanning and reconstruction of high or distant structures problematic.

When a wide facade or a short street is scanned, hundreds of 3D point clouds are stored. Figure 1 gives an example of point cloud. A cloud typically contains $30000-40000$ points. The measurements are noisy, and they are always spoiled by outliers that make the alignment more difficult. These data are to be registered and aligned in a single point set, as illustrated in figure 2. The complete aligned point set is then triangulated to obtain a surface mesh. The mesh makes much better visible the misalignments 'hidden' in the final point set. The misalignments must be automatically detected and corrected.

Currently, we do not use any auxiliary sensor (GPS, IMU) to support alignment. Before doing that, one has to carefully analyse the precision of the sensor. If the precision is not sufficiently high, adjusting point clouds according to the sensor data may be counterproductive. For example, a small angular error may result in a significant position error at large distance.

In the context of the car-mounted LIDAR data registration problem, the main sources of outliers are moving cars and pedestrians, surfaces with unstable reflectance, interior of buildings, as well as curtains and windows.

Data points resulting from moving objects deteriorate the reconstruction of static environments. Their number depends on the density of the traffic. Such objects can be detected easily as their height and size vary in a limited, well-defined range.

Outliers resulting from objects with unstable reflectance, such as trees in the wind, are relatively rare. However, vegetation can occlude significant portions of buildings and generate unstable point clouds.

Interiors of buildings supply data whose character can change suddenly. In one view the measurements may cover a large area and be useful for alignment, while in the subsequent scans the usable area may decrease or disappear entirely. Curtains and windows pose a similar problem as the laser beam may be reflected from them in one view and penetrate it in the next view. Usually, these two categories of outliers in LIDAR scans appear as remote points behind the actual visible surface, and they can be detected based on this property.

Pre-conditioning of the measured data is very desirable since it improves the robustness of alignment and enhances the quality of the resulting mesh. Both of our methods presented below try to detect and remove the outliers prior to point cloud registration.

3. NDT-based Method

Our first point cloud sequence alignment method is based on the 3D NDT registration algorithm [7]. Before registration, the acquired LIDAR data is pre-segmented by classifying each point as belonging to ground, short field object (vehicle, pedestrian), tall structure object (wall, roof, lamps post), or clutter. To achieve this, we use statistical features in the grid-based approach [4]. The procedure runs in real time as it uses only a few simple features of each grid cell such as the height difference between points within a particular cell.

The aim of the pre-segmentation is to select static, stable points that do not change their positions between scans. Such regions are, i.e., walls that are visible from large distances and appear in several scans as the car passes them by. The benefit of this is two-fold. First, the registration speeds up since less points are processed once the irrelevant points have been removed. Second, the preserved 3D data are the most useful regions of the scans. The regions that are hard or impossible to register, such as moving cars and pedestrians, are discarded.

The 3D NDT registration algorithm [7] uses a 3D voxel-based approach to match subsequent point clouds. We apply the 3D NDT to find the optimal rigid transformation between two neighbouring scans and validate the result since the registration may occasionally fail.

Since the 3D data is recorded in real-world environments, we can set up constraints on the transformation matrices. In typical urban traffic, the car carrying the LIDAR travels about 40−80 centimeters between consecutive scans; its speed cannot change drastically between scans. This limits the translation vector. Also, rotation should be reasonably small since the car stays on the road and turns only with a limited speed.

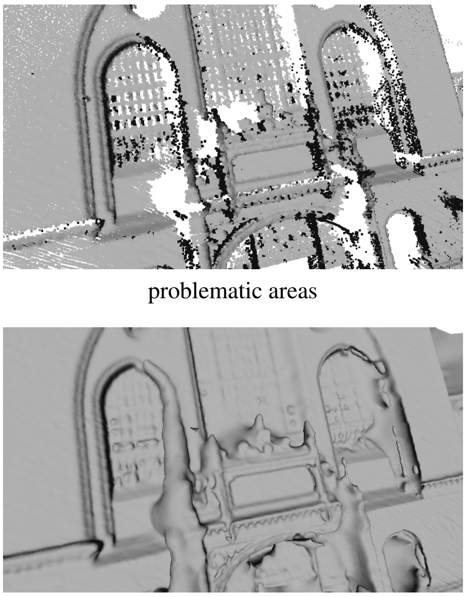
The validation procedure checks the obtained transformation matrix. If either the translation vector or the rotation matrix is unrealistic, the procedure rejects the result and skips the processed scan.

4. TrICP-based Method

Our second point cloud sequence alignment method is based on the TrICP registration algorithm [2]. Prior to point cloud registration, the input data are filtered to remove outliers in order to robustify and speed up the registration process. The procedure also provides surface normals that are used when a mesh is obtained from the point cloud sequence aligned by any of the two methods. In this section we, briefly discuss the main components of the TrICP-based method.

4.1. Data filtering and normal calculation

Similarly to the NDT-based method, the TrICP-based algorithm needs pre-processing to condition the input data, as discussed in section 2. The alignment block receives data processed by the pre-segmentation algorithm presented in section 3. In addition, we have developed a relatively simple but robust filtering procedure that removes other unreliable data such as occasionally measured interior structures of the buildings.


problematic areas

mesh errors

Figure 3: *Problematic areas and resulting mesh errors.*

The filtering procedure also assigns normal vectors to the remaining points in a consistent way, alleviating the problem of the 'flipped' normals typical for many normal calculation algorithms. The Poisson surface reconstruction algorithm [6] we use assumes consistent normals. In the flipped areas, the normals point in the opposite direction which results in mesh errors, as illustrated in figure 3. In the figure, flipped normal areas are shown in dark, unmeasured areas in light.

The data filtering method is based on the Hidden Point Removal (HPR) operator [5] that provides a theoretical basis for testing visibility on unordered, unoriented point sets. Although the name reminds the hidden surface removal, the HPR does not use surfaces or surface reconstruction. Instead, the operator applies a view-based point set inversion method and a convex hull calculation algorithm on the inverted point set. The triangle connectivity of the convex hull transferred to the original point set provides consistent surface normals. In our solution, we selected the 'spherical flipping' [5] as the inversion method. Figure 4 illustrates the operation of our filtering and normal calculation method.

4.2. Data alignment

In a long sequence of point clouds, we select each K-th cloud and apply TrICP to register each selected cloud to the next one in the subsequence. The typical values of the temporal step K are 2−4. Each partial registration is characterised by a Mean Square Error (MSE) value, the mean square distance

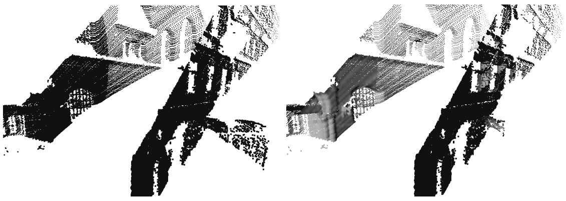


Figure 4: *Removing an outlier internal structure by the HPR operator.*

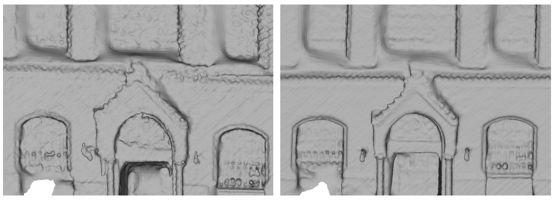


Figure 6: *Detail of Market Hall reconstruction.*

between the corresponding points in the two registered sets. Although the point-to-surface distance is often preferred to the point-to-point distance, in this particular application we still use the latter. The reason is that our data is dominated by flat surfaces of facades. As discussed in [11], the point-to-point metric is preferable for data with predominantly low or constant curvature.

A key parameter of the algorithm is the expected overlap of the two point sets. TrICP can search for the optimal value of this parameter and set it automatically, which needs additional computation. This makes sense when the overlap varies across the sequence. In the car-mounted LIDAR data, the overlap is sufficiently stable and easy to set. For this reason, we use a fixed overlap for each value of the parameter K.

The series of registrations results in a series of MSE values. We analyse these values and detect poor registrations as large outliers in the MSE array. A standard robust outlier detection rule [12] is used which is based on the median absolute deviation from the median value.

Each poor registration is discarded and substituted by a short sequence of registrations with the unit temporal step $K = 1$. In other words, a gap in the registration chain is bridged by aligning all clouds within the gap. The chance that this will repair the chain is high as we observed that misregistrations for different values of K do not correlate and appear in different places. At the same time, using $K = 1$ for the whole sequence is not a good solution as it is slower and often results in an even larger number of errors than with $K = 2$ or 3.

Finally, the transformation matrices of the complete repaired registration chain are multiplied, and each initial point cloud is registered to the reference cloud which is the last cloud of the sequence. In principle, it would be possible to ‘dissipate’ the registration errors in the chain [11] and smooth the overall alignment. However, given the large size of the data, this would require excessive computation, so we decided to omit this step. Currently, typical execution time of the algorithm is $20-30$ minutes for a sequence of 300 point clouds.

5. Test Results

In this section, we demonstrate and compare sample results of reconstruction from car-mounted LIDAR data. The aligned point sets are converted to triangular meshes using the Poisson Surface Reconstruction algorithm [3, 6] which works on oriented point sets. As discussed in section 4.1, the normals are assigned to points by our data filtering algorithm. Alternatively, we could use the normal calculation algorithm provided by the Meshlab package [3]. However, we have experienced that this algorithm is slower and less robust; in particular, it can produce flipped normals.

The test results form two distinct groups. Section 5.1 is devoted to the Market Hall data acquired along a facade of the Central Market Hall of Budapest (in Hungarian, Központi Vásárcsarnok). This building features a number of characteristic architectural elements, which can facilitate the alignment but makes the potential reconstruction errors more visible. The Kende Street data discussed in section 5.2 represents most of this short street and includes $7-8$ buildings on each side. Most of the buildings have simple, featureless facades that are significantly higher than those of the Market Hall.

5.1. Market Hall

Figure 5 shows the front facade of the Market Hall reconstructed by the two methods. Both textureless and textured versions are presented. The TrICP-based alignment algorithm processed each third of the original 300 point clouds ($K = 3$) with the overlap value of 85%. Five bad registrations were detected and successfully repaired using $K = 1$ and overlap 95%. The overall quality of the reconstruction is good as the global geometry and the fine details are correct.

For the Market Hall data, the NDT-based alignment method yields worse results. In addition to the global bendings and visible surface roughness, some structural elements and details appear distorted and blurred, as illustrated in figure 6 where a vicinity of a market entrance is shown enlarged. The TrICP alignment better preserves the geometry and the details.

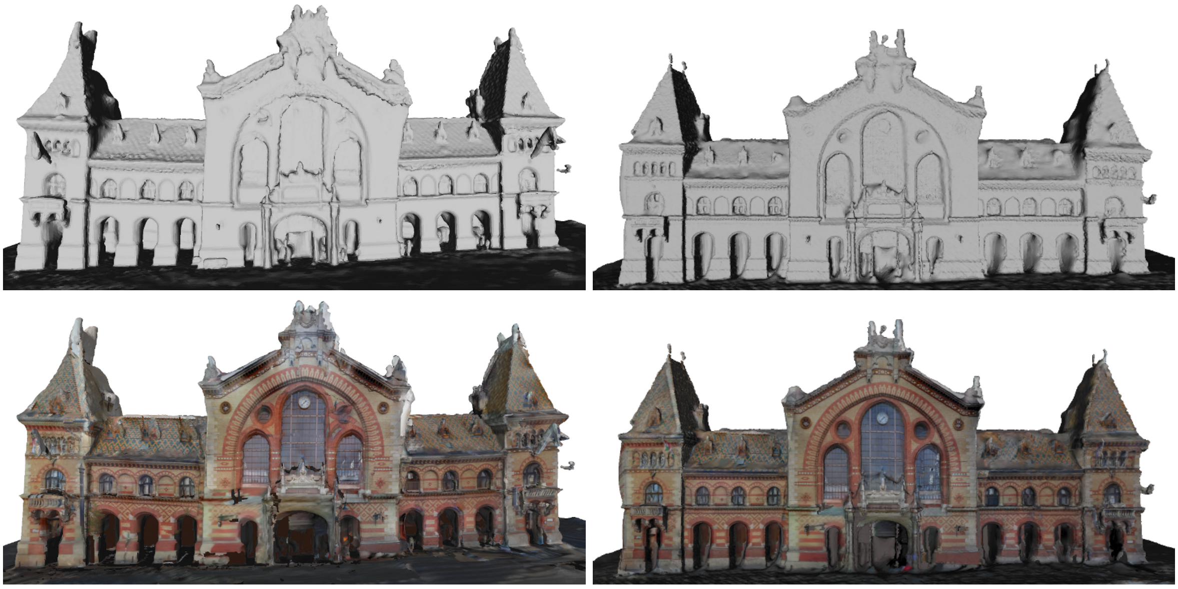


Figure 5: *Reconstruction of Market Hall front facade.*

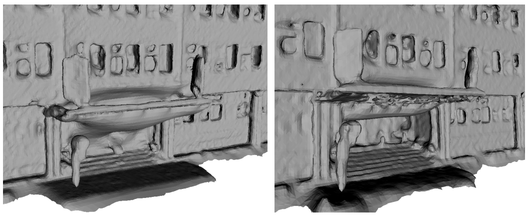


Figure 8: *Reconstruction of the entrance of MTA SZTAKI.*

5.2. Kende Street

For the Kende Street data, the outcome of the test is just the opposite. Here, the Trimmed ICP often fails to cope with featureless and sparse pieces of data, while NDT yields satisfactory results demonstrated in figure 7. The overall quality is nevertheless much lower than for the Market Hall data.

When the 3D data contains distinct features and is sufficiently dense (e.g., at low altitudes), TrICP can still produce reasonable output. Figure 8 compares the two results at the vicinity of the entrance of the main building of MTA SZTAKI situated in the street. There is no big difference in the quality.

However, in several other areas of the street the performance of TrICP is poor. Figure 9 gives an example when the reconstruction by TrICP exhibits severe global and local distortions, while the result by NDT is visibly better, although not perfect.

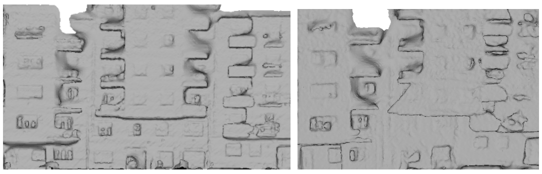


Figure 9: *Part of Kende Street reconstruction.*

6. Discussion and Conclusion

We have presented two methods for aligning long sequences of point clouds acquired by a car-mounted LIDAR device measuring facades in a street. Our current experience with the methods can be summarised as follows.

Due to nature of the LIDAR scans and the fact that the NDT algorithm works in discrete space, this registration method is sensitive to the discretisation parameter. For challenging scenes such as facades with repeating patterns (i.e., rows of similar windows), this parameter may need to be increased to 50−70 centimeters which can result in distorted registrations. Fine details can be blurred. Otherwise, the method is suitable for reconstructing single facades as well as sequences of facades. It is less sensitive to featureless and sparse areas than the TrICP-based method.

The execution times of the two algorithms are comparable. However, the speed of NDT decreases drastically as the spatial resolution increases. The speed of TrICP is less sensitive to the setting of its parameters.

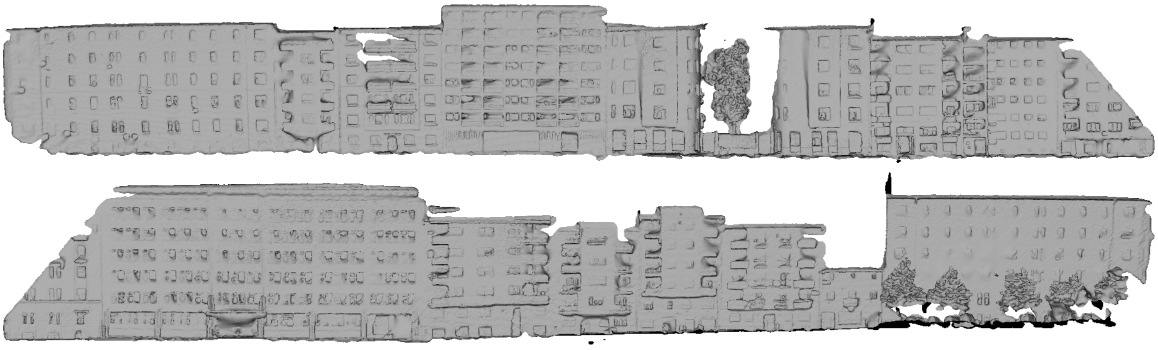


Figure 7: *NDT reconstruction of Kende Street facades. Top: East side. Bottom: West side.*

TrICP is applicable to, and produces superior results for, surfaces containing characteristic features. It can cope with larger rotations between scans then NDT. However, when featureless or sparse areas dominate, TrICP works at the limit of its capabilities and produces multiple misalignments that cannot be corrected by the proposed procedure.

In future, we plan to investigate the role of LIDAR sampling, in general, and its influence on the systematic registration errors, in particular. We have observed that both methods are sensitive to the order in which the point clouds are processed. This may result from the asymmetry of their cost functions w.r.t. the two clouds, as well as from the anisotropy of data sampling by LIDAR.

The possibility and efficiency of using auxiliary sensors in our street scenario will also be studied. If the answer is positive, such sensors will be applied. However, the precision of today's sensors does not seem to be sufficient for our purposes.

The TrICP-based method may profit from the prior knowledge of limited rotation and shift, which currently is not utilised. The NDT-based method can be enhanced by the filtering procedure introduced in section 4.1.

Both methods will be applied to the reconstruction of complete models of buildings and quarters. Such reconstruction will need fast and efficient algorithms for alignment error dissipation within a circular data sequence, to avoid error accumulation and global misalignment when the registration loop terminates.

Acknowledgment

This work was supported by an internal grant of the Institute for Computer Science and Control. The authors acknowledge the contribution of Oszkár Józsa who implemented and tested the NDT-based registration algorithm.

References

1. C. Benedek, Z. Jankó, C. Horváth, D. Molnár, D. Chetverikov, and T. Szirányi. An integrated 4D vision and visualisation system. In *Computer Vision Systems*, volume 7963 of *Lecture Notes in Computer Science*, pages 21–30. Springer, 2013.

2. D. Chetverikov, D. Stepanov, and P. Krsek. Robust Euclidean alignment of 3D point sets: the Trimmed Iterative Closest Point algorithm. *Image and Vision Computing*, 23:299–309, 2005.

3. P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, and G. Ranzuglia. Meshlab: an open-source mesh processing tool. In *Proc. Eurographics Italian Chapter Conference*, pages 129–136, 2008.

4. O. Józsa, A. Börcs, and C. Benedek. Towards 4D virtual city reconstruction from LiDAR point cloud sequences. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W1:15–20, 2013.

5. S. Katz, A. Tal, and R. Basri. Direct visibility of point sets. In *ACM SIGGRAPH*. ACM, 2007.

6. M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proc. Fourth Eurographics Symposium on Geometry Processing*, pages 61–70. Eurographics Association, 2006.

7. M. Magnusson. *The Three-Dimensional Normal-Distributions Transform – an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. PhD thesis, Örebro University, December 2009.

8. M. Magnusson, A. Nuchter, C. Lorken, A.J. Lilienthal, and J. Hertzberg. Evaluation of 3D registration reliability and speed – a comparison of ICP and NDT. In *Proc. IEEE International Conference on Robotics and Automation*, pages 3907–3912, 2009.

9. P. Musialski, P. Wonka, D.G. Aliaga, M. Wimmer, L. Gool, and W. Purgathofer. A survey of urban reconstruction. In *Computer Graphics Forum*, volume 32, pages 146–177, 2013.

10. F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat. Comparing ICP variants on real-world data sets. *Autonomous Robots*, pages 1–16, 2013.

11. K. Pulli. Multiview registration for large data sets. In *Proc. Second International Conference on 3-D Digital Imaging and Modeling*, pages 160–168, 1999.

12. P.J. Rousseeuw and A.M. Leroy. *Robust regression and outlier detection*. Wiley, 2005.

13. S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Proc. Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

14. T. Stoyanov, M. Magnusson, H. Almqvist, and A.J. Lilienthal. On the accuracy of the 3D normal distributions transform as a tool for spatial representation. In *Proc. IEEE International Conference onRobotics and Automation*, pages 4080–4085, 2011.

15. Q.-Y. Zhou and U. Neumann. Complete residential urban area reconstruction from dense aerial LiDAR point clouds. *Graphical Models*, 75:118–125, 2013.