BENCHMARK OF 6D SLAM (6D SIMULTANEOUS LOCALISATION AND MAPPING) ALGORITHMS WITH ROBOTIC MOBILE MAPPING SYSTEMS

Janusz BEDKOWSKI [*], Timo RÖHLING[†], Frank HOELLER[†],
Dirk SHULZ[†], Frank E. SCHNEIDER [†]

Abstract. This work concerns the study of 6DSLAM algorithms with an application of robotic mobile mapping systems. The architecture of the 6DSLAM algorithm is designed for evaluation of different data registration strategies. The algorithm is composed of the iterative registration component, thus ICP (Iterative Closest Point), ICP (point to projection), ICP with semantic discrimination of points, LS3D (Least Square Surface Matching), NDT (Normal Distribution Transform) can be chosen. Loop closing is based on LUM and LS3D. The main research goal was to investigate the semantic discrimination of measured points that improve the accuracy of final map especially in demanding scenarios such as multi-level maps (e.g., climbing stairs). The parallel programming based nearest neighborhood search implementation such as point to point, point to projection, semantic discrimination of points is used. The 6DSLAM framework is based on modified 3DTK and PCL open source libraries and parallel programming techniques using NVIDIA CUDA. The paper shows experiments that are demonstrating advantages of proposed approach in relation to practical applications. The major added value of presented research is the qualitative and quantitative evaluation based on realistic scenarios including ground truth data obtained by geodetic survey. The research novelty looking from mobile robotics is the evaluation of LS3D algorithm well known in geodesy.

Keywords: Mobile robot, mobile mapping system, Iterative Closest Point, Least Square Surface Matching, Normal Distribution Transform, LUM, 6DSLAM, CUDA.

[*]Institute of Fundamental Technological Research, Polish Academy of Science, Warsaw, Poland IPPT PAN{januszbedkowski}@gmail.com

[†]Fraunhofer-Institut für Kommunikation, Informationsverarbeitung und Ergonomie, Kognitive Mobile Systeme, Wachtberg, Germany FKIE, {timo.roehling, frank.hoeller, dirk.schulz, frank.schneider}@fkie.fraunhofer.de

1.   Introduction

The problem of data registration can be considered as solved since there are plenty of available algorithms in mobile robotics and geodesy domains. Obviously, it is worth to mention work [8] as the relevant source within the context of Iterative Closest Point (ICP) algorithm. ICP computes the transformation matrix between two point clouds, thus the sum of Euclidean distances between nearest neighbours is minimised. It can be used for odometry error correction. Unfortunately, the incremental error is increasing, thus loop closing has to be performed to optimise entire trajectory resulting more accurate 3D map. Instead of ICP which optimises error iteratively, works [14] and [15] introduce closed form solution giving result in the single optimisation step. An alternative approach to ICP is Normal Distribution Transform (NDT) that was introduced in [18]. Authors of NDT claim the advantage over ICP within the context of robustness in the case of large initial error of alignment. It is evident that ICP performs similarly to NDT for the cases were the initial trajectory is given. Conversely, NDT does not require accurate initial trajectory for most cases. The problem of aligning point clouds is also studied in geodesy domain. Here the relevant approach is Least Square Surface Matching [2] which, according to our best knowledge is not yet adapted for mobile robotics domain, thus this can be considered as added value of the paper. In comparison to ICP, LS3D uses the different representation of data. Instead of point cloud LS3D uses surface representation, thus it minimises the Euclidean distance between surfaces to find transformation matrix between point clouds. It is worth to comment why LS3D is not so popular in mobile robotics. Hence it requires additional computation for extracting surface representation from the point cloud. Additionally, the accuracy depends on the surface representation, thus the uncertainty of the measurement strongly affects the results. Moreover, the surface representation determines better accuracy in structured environments, thus a large number of trees or bushes could drastically affect the result. The advantage of LS3D over iterative ICP and NDT is faster convergence that typically requires up to six optimisation steps. Looking from the geodetic point of view the added value of LS3D is full information concerning the optimisation process. It is evident that many researchers are working on the improvement of those methods, mainly they are focused on computer science methodologies for efficient 3D data processing. For example work [13] introduces efficient 3D data decomposition (kd-tree) for faster nearest neighbourhood search (NNS). Additionally, work [24] introduces parallel programming approach for kd-tree. There are plenty of possible applications for data registration, for example, work [20] shows the underground mine mapping use case. Further works [21], [26], [12] introduce new optimisation techniques and robust 3D data processing methods.

It is evident that research community strongly supports 3D data processing domain, thus it is worth to mention some open-source frameworks: 3DTK (The 3D Toolkit) [1], PCL (Point Cloud Library) [23] and ROS (Robot Operating System) [25]. These software frameworks were the basis for the work presented in this paper. The motivation of our research was to chose and to combine relevant state of the art registration techniques into a common framework to find not always obvious

advantages and disadvantages that could help with the deployment of the proper algorithmic solution for the certain robotic mobile mapping application. The added value of our work is the improvement based on the parallel implementation, thus we reached on-line registration, which is important for moving robot that will use this information for the navigation.

The paper is organised as follows: *Methodology* introduces entire software framework evaluated in the *Qualitative evaluation* and *Quantitative evaluation* subsection. *Conclusion* finalises the paper and comments future work.
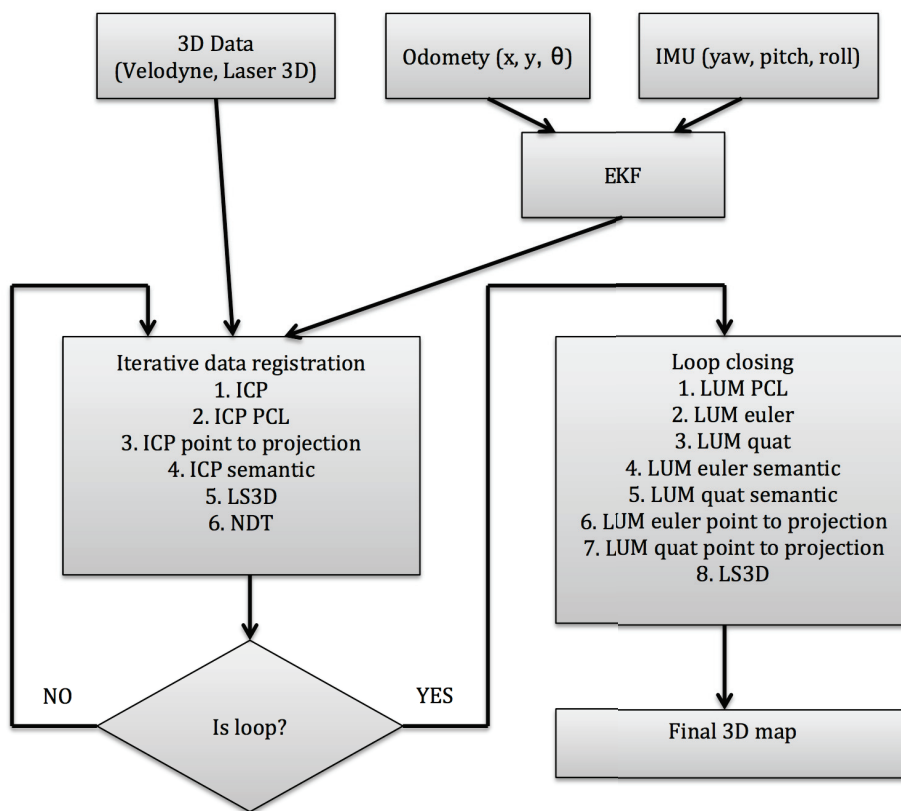
2. Methodology

Figure 1 shows the scheme of the framework concerning the 6D Simultaneous Localisation and Mapping Algorithms evaluated in this paper. The implementation is using open source components from 3DTK [1] (loop closing optimisation - LUM), PCL [23] (data structures, NDT, ICP, LUM), CUDA [11] (parallel implementation) and ROS [25] (integration with robotic platform). The input for 6DSLAM is 3D laser range finder data and pose obtained by Extended Kalman Filter - EKF fusing odometry (x, y, z) and inertial measurement unit readings (yaw, pitch, roll, accelerations, rotational velocities). It is considered stop-scan fashion of derived 3D laser range finder data. Due to a use of Velodyne HDL-64E 3D data are acquired during robot motion. Conversely, the use of rotated profile laser requires static 3D measurement (robot is not moving during the measurement). Initial trajectory and 3D data associated with different positions from this trajectory are the input for iterative data registration component. Thus, there are available six registration algorithms:

- ICP with parallel implementation of nearest neighbourhood procedure NNS.

- ICP - PCL implementation.

- ICP point to projection with parallel NNS.

- ICP semantic with parallel discrimination of points into four classes *plane*, *edge*, *floor/ground*, *ceiling*).

- LS3D Least Square Surface Matching with the parallel computation of surface representation.

- NDT - PCL implementation).

The output is the corrected trajectory with a cumulated iterative error. Thus, if the loop is considered to be closed the loop closing component performs final map relaxation, thus we suppose to obtain the accurate solution. The framework contains eight different loop closing implementations:

- LUM (1) based on PCL implementation .

- LUM euler (2) and quat (3) based on 3DTK improved by parallel implementation.

- LUM euler (4) and quat (5) semantic based on 3DTK improved by parallel implementation and semantic discrimination of points for better matching.

- LUM euler (6) and quat (7) point to projection based on 3DTK improved by parallel implementation.

- LS3D (8) multi view Least Square Surface Matching with parallel computation of surface representation.

The minor advantage of the framework is the possibility of building several combination of 6DSLAM algorithm. The major advantage is the freedom to chose relevant SLAM solution for a certain application.

Figure 1. The scheme of the framework concerning the 6D Simultaneous Localisation and Mapping Algorithms evaluated in this paper.

2.1. Iterative Closest Point (ICP)

The goal of ICP algorithm is to find the transformation matrix that minimises the sum of distances between the corresponding points in two different data sets, therefore two important aspects have to be solved:

- nearest neighbourhood search,

- choosing the proper optimisation technique for the minimisation of the mentioned function (estimation of the 3D rigid transformation).

Iteratively repeating these two steps results in convergence to the desired transformation. Semantic discrimination of these correspondences improves the convergence. Range images (scans) are defined as model set M where

$$|M| = N_m \tag{1}$$

and data set D where

$$|D| = N_d \tag{2}$$

The goal of semantic ICP is to minimise following cost function:

$$E\left(\mathbf{R}, \mathbf{t}\right) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} w_{ij} \left\| \mathbf{m}_i^c - \left(\mathbf{R}\mathbf{d}_j^c + \mathbf{t}\right) \right\|^2 \tag{3}$$

where w_{ij} is assigned 1 if the ith point of M correspond to the jth point in D in the sense of minimum distance and the same class. Otherwise $w_{ij}=0$. Class c discriminates points into wall, ceiling, floor or edge. $\mathbf{R}$ is the rotation matrix, $\mathbf{t}$ is the translation matrix, $\mathbf{m}_i^c$ corresponds to points of class c from the model set M, $\mathbf{d}_j^c$ corresponds to points of class c from the data set D. Obviously for the ICP without semantics class c = 0 for all points.

Choosing the proper optimisation technique in ICP like algorithms has been a research topic during the last decades. A comparison of four algorithms for estimating 3D rigid transformation is shown in [16]. The first algorithm proposed in [4] is using Singular Value Decomposition. The second approach is based on orthonormal matrices and computation of eigensystem of derived matrix and it is proposed in [15]. The third algorithm is shown in [14], it finds the transformation for the ICP algorithm by using unit quaternions. The forth algorithm shown in [27] is using so-called dual quaternions. Instead of these four closed-form solution methods, a novel linear solutions to the scan registration problem is shown in [22]. The advantage of these new linear solutions is that they can be extended straightforward to n-scan registrations. In presented framework, we use ICP from PCL library and ICP improved by parallel nearest neighbourhood search and SVD optimisation inspired by 3DTK framework. More information is available here [5].

2.2. Least Square Surface Matching (LS3D)

LS3D can be used both for iterative data registration and for loop closing. The implementation of iterative registration is based on work [2] and [3] for multiple surface matching. The novelty is parallel computing approach for normal vector computation as surface representation [6] and the use of CUSOLVER LIBRARY [11] for solving $\mathbf{Ax}=\mathbf{B}$ problem. The method assumes a set of n surfaces of an object: $g_1(x, y, z), ..., g_n(x, y, z)$. The object is defined in a 3D Cartesian coordinate

system and the surfaces are considered to be located in local coordinate systems. The surface is represented by local planes (normal vectors attached to point cloud).

There are m mutual spatial overlaps between the surfaces $g_i(x, y, z)$ thus, each overlap satisfies a pairwise matching:

$$g_i(x, y, z) - e_i(x, y, z) = g_j(x, y, z), i, j = 1, ..., n, i \neq j \tag{4}$$

where $e_i(x, y, z)$ is a true error vector. Each surface is associated with transformation in 3D:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_i = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}_i + m_i \mathbf{R}_i \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}_i \tag{5}$$

where $\mathbf{R}_i = \mathbf{R}_i(\omega, \varphi, \kappa)$ is the rotation matrix, $[t_x, t_y, t_z]_i^T$ is the translation vector, m_i is the uniform scale factor, (x_0, y_0, z_0) is the initial location of the surface. Equation 4 is nonlinear, thus it is linearised by Taylor series expansion:

$$-e_i(x, y, z) = g_j^0(x, y, z) + \frac{\partial g_j^0(x, y, z)}{\partial x_j} dx_j + \frac{\partial g_j^0(x, y, z)}{\partial y_j} dy_j + \frac{\partial g_j^0(x, y, z)}{\partial z_j} dz_j -$$
$$g_i^0(x, y, z) - \frac{\partial g_i^0(x, y, z)}{\partial x_i} dx_i - \frac{\partial g_i^0(x, y, z)}{\partial y_i} dy_i - \frac{\partial g_i^0(x, y, z)}{\partial z_i} dz_i \tag{6}$$

where dx, dy and dz correpond to differentiations of the transformation (equation 5)

$$dx = dt_x + a_{10} dm + a_{11} d\omega + a_{12} d\varphi + a_{13} d\kappa$$
$$dy = dt_y + a_{20} dm + a_{21} d\omega + a_{22} d\varphi + a_{23} d\kappa \tag{7}$$
$$dz = dt_z + a_{30} dm + a_{31} d\omega + a_{32} d\varphi + a_{33} d\kappa$$

with a_{pq} as the coefficient terms briefly described in [2], thus we obtain following form:

$$
\begin{aligned}
-e_i(x, y, z) = & g_{xj} dt_{xj} + g_{yj} dt_{yj} + g_{zj} dt_{zj} \\
& + (g_{xj} a_{10} + g_{yj} a_{20} + g_{zj} a_{30}) dm_j \\
& + (g_{xj} a_{11} + g_{yj} a_{21} + g_{zj} a_{31}) d\omega_j \\
& + (g_{xj} a_{12} + g_{yj} a_{22} + g_{zj} a_{32}) d\varphi_j \\
& + (g_{xj} a_{13} + g_{yj} a_{23} + g_{zj} a_{33}) d\kappa_j \\
& - g_{xi} dt_{xi} - g_{yi} dt_{yi} - g_{zi} dt_{zi} \\
& - (g_{xi} b_{10} + g_{yi} b_{20} + g_{zi} b_{30}) dm_i \\
& - (g_{xi} b_{11} + g_{yi} b_{21} + g_{zi} b_{31}) d\omega_i \\
& - (g_{xi} b_{12} + g_{yi} b_{22} + g_{zi} b_{32}) d\varphi_i \\
& - (g_{xi} b_{13} + g_{yi} b_{23} + g_{zi} b_{33}) d\kappa_i \\
& - (g_i^0(x, y, z) - g_j^0(x, y, z))
\end{aligned}
\tag{8}
$$

Equation 8 results in the following linear systems in matrix/vector form:

$$
\begin{aligned}
-\mathbf{e}_1 &= \mathbf{A}_1\mathbf{x} - \mathbf{l}_1, \mathbf{P}_1 \\
-\mathbf{e}_2 &= \mathbf{A}_2\mathbf{x} - \mathbf{l}_2, \mathbf{P}_2 \\
&\vdots \\
-\mathbf{e}_m &= \mathbf{A}_m\mathbf{x} - \mathbf{l}_m, \mathbf{P}_m
\end{aligned}
\tag{9}
$$

where $\mathbf{A}$ is the design matrix, $\mathbf{x}$ is the parameter vector containing n sets of transformation parameters, $\mathbf{P}$ is the weight matrix and $\mathbf{l} = g_i^0(x,y,z) - g_j^0(x,y,z)$ as the Euclidean distances between the corresponding elements of the overlapping surfaces. Solving this $\mathbf{Ax} = \mathbf{B}$ problem results n sets of transformation parameters.

2.3. Normal Distribution Transform (NDT)

The normal distributions transform (NDT) uses the different representation of the data [9]. Instead of using the individual points of the model point cloud like in ICP or surfaces as in LS3D, NDT uses a combination of normal distributions, describing the probability of finding part of the surface at any point in space. The normal distributions give a piecewise smooth representation of the model point cloud, with continuous first and second order derivatives. Using this representation, it is possible to apply standard numerical optimisation methods for registration. More information can be found in [19]. The NDT models the distribution of all reconstructed 3D-points of one laser scan by a collection of local normal distributions. First, the 3D space around the robot is subdivided into cells with constant size. The for each cell, that contains sufficient number of points, the following procedure is performed:

- Collect all 3D-Points $\mathbf{x}_{i=1..n}$ contained in this box.

- Calculate the mean $\mathbf{q} = \frac{1}{n}\sum_i \mathbf{x}_i$.

- Calculate the covariance matrix $\Sigma = \frac{1}{n-1}\sum_i(\mathbf{x}_i - \mathbf{q})(\mathbf{x}_i - \mathbf{q})^t$.

The probability of measuring a sample at 3D-point $\mathbf{x}$ contained in this cell is modelled by the normal distribution $N(q, \Sigma)$: $p(\mathbf{x}) = \frac{1}{C}exp(-\frac{(\mathbf{x}-\mathbf{q})^t\Sigma^{-1}(\mathbf{x}-\mathbf{q})}{2})$. Where $C = (2\Pi)^{\frac{3}{2}}\sqrt{|\Sigma|}$ is constant value that is used to normalise $p(\mathbf{x})$.

2.4. Loop closing - Lu and Milios style SLAM (LUM)

We use an extension of Lu and Milios [17] SLAM into 3D space presented in work [10] available in 3DTK open-source framework [1]. Once a loop is detected, i.e., the distance between the poses of two scans falls below a threshold, the adaptation of Lu/Milios style scan matching is applied. In the experiments, we set this threshold as 30 meters in case of robot equipped with Velodyne HDL-64E and 100 meters in case of a robot equipped with Z+F IMAGER 5010. For each iteration, a network of pose relations is built automatically. From the corresponding scans, a linear equation

system representing distance measurements is built and solved, resulting in optimised pose estimations.

3.   Qualitative evaluation

Figure 2:left shows the mobile robotic platform from Cognitive Mobile Systems Group, Fraunhofer Institute FKIE Wachberg, Germany equipped with laser system 3D Velodyne HDL-64E and other important sensors (odometry, IMU) looking from a mobile mapping point of view. Mobile platform produces relevant data for the 6DSLAM algorithm (3D point cloud typically of 10.000 points assuming downsampling using regular grid of 0.3 x 0.3 x 0.3 m, odometry, IMU), thus for certain positions we obtain full 3D measurement (e.g. stop-scan fashion) even for fast motion due to advantage of robust 3D laser. Figure 2:centre shows the 170x130m testing environment. 6DSLAM framework was integrated with ROS Indigo (Robot Operating System - Inigo distribution release), thus figure 2:right shows the data visualisation related with the evaluated 6DSLAM algorithm. The goal was to record so called ROSBAG data containing entire information during robotic trial and use it for postprocess evaluation, thus 3D scans, odometry and IMU readings were available to test different algorithms. Figure 3 shows the time of registrations (iterative data registration component of 6DSLAM - ICP, LS3D and NDT) in relation to two testing computers:

- Intel(R) Core(TM) i7-4820K CPU @ 3.70GHz, GeForce GTX TITAN X (CUDA cores: 3072, Graphics Clock: 1000 MHz, Graphics Card TDP: 250W)

- Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz, GeForce GT 650M (CUDA cores: 384, Graphics Clock: 900 MHz, Graphics Card TDP: 45W)

The computers were chosen to demonstrate most powerful GPGPU processor NVIDIA GeForce GTX TITAN X (MAXWELL architecture) available during experiments that requires 250W of energy and affordable mobile processor GeForce GT 650M consuming 45W. From the evaluation, we observed that the computing power of GeForce GTX TITAN X is sufficient for on-line data registration assuming a frame rate of five full 3D laser scans per second. GeForce GT 650M provides much less computing power, thus it requires more time for registration. Hence, it can not be considered for on-line mapping application with a fast moving robot. It can be used in stop-scan scenarios where there is the sufficient time between the acquisition of observations and the movement to the next goal. The resulting trajectories of iterative data registration component of 6DSLAM are shown in figure 4. Hence, we observed comparable results. We claim that in this particular scenario it is not evident which of the iterative data registration technique performs better. According to the performance with the GeForce GTX TITAN X all of methods provide on-line computations. Even NDT that is based on PCL CPU based implementation guarantee on-line computation, but it is up to 10x slower that GPU based algorithms.

For the second phase of the experiment, the ICP-based trajectory was chosen as reference data for loop closing. Hence, eight different loop-closing implementations

were compared (LUM - PCL implementation, LUM - euler, LUM - quat, LUM - euler semantic, LUM - quat semantic, LUM - euler point to projection, LUM - quat point to projection, LS3D). The time of loop closure computation for certain method is given in table 1 and the resulting maps are given in figure 5. Hence, it is evident that there is a difference between methods looking from the final map accuracy. Algorithms: LUM PCL, LUM quat, LUM quat semantic are producing comparable maps that are less accurate than other methods. It is worth to mention that semantic approach and LS3D methods are in our opinion most promising algorithms.

To convince Reader about the advantage of the semantic approach we decided to introduce another experiment with the different robot (figure 6) equipped with the 3D laser measurement unit based on the rotated LMS SICK TIM500 moving down the stairs. The robot was acquiring data in stop-scan fashion. Thus, figure 6 shows the input data (3D local measurements transformed via odometry fused with IMU). Figure 7 shows the result of iterative data registration component of 6DSLAM. This particular use case demonstrates the advantage of semantic approach in relation to other techniques. Hence, other techniques tend to match floor with the ceiling. ICP semantic uses discrimination of points, thus matches between points classified as floor and points classified as ceiling are not considered in the optimisation. Thus, it improves the accuracy of the final map.
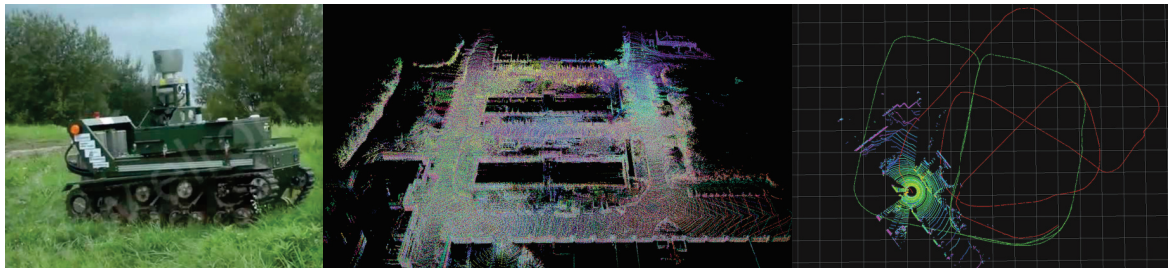
Figure 2. Mobile robotic platform from Cognitive Mobile Systems Group, Fraunhofer Institute FKIE Wachberg, Germany. Testing environment and plot of acquired data.

4.   Quantitative evaluation

The quantitative evaluation was performed based on ground truth data obtained by the geodetic survey. The mapping robotic system (mobile robot equipped with accurate 3D laser scanner Z+F IMAGER 5010 producing even 100.000.000 raw points, thus 300.000 points are used in average assuming downsampling based on regular grid 0.3 x 0.3 x 0.3 m), realistic environment, geodetic survey equipment and accuracy plot are shown in figure 8. The geodetic survey gave us the opportunity to collect ground truth data with accuracy of the final control network approximately 0.3mm [7]. The goal of quantitative evaluation is to compare automatic registration techniques with manual geodetic survey, thus we show relevant plots of X, Y, Z, X angle, Y angle, Z angle errors as difference between ground truth data and relevant
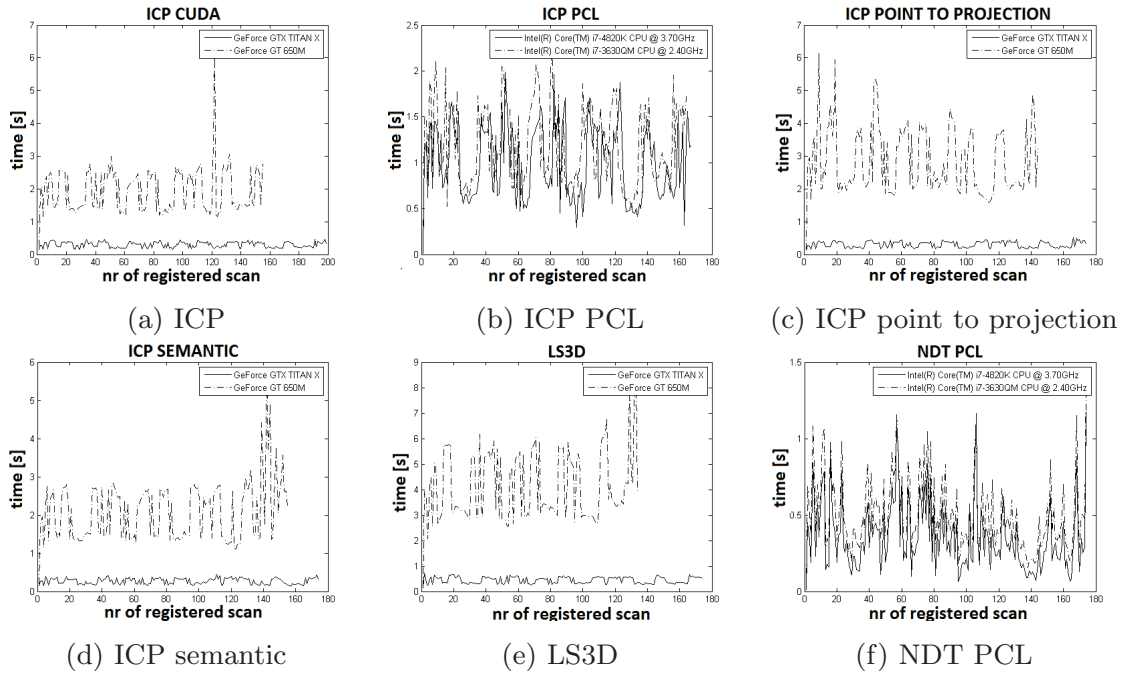
(a) ICP   (b) ICP PCL   (c) ICP point to projection

(d) ICP semantic   (e) LS3D   (f) NDT PCL

Figure 3. Time of iterative registrations for certain method using two different PCs: 1. Intel(R) Core(TM) i7-4820K CPU @ 3.70GHz, GeForce GTX TITAN X (CUDA cores: 3072, Graphics Clock: 1000 MHz, Graphics Card Power: 250W), 2. Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz, GeForce GT 650M (CUDA cores: 384, Graphics Clock: 900 MHz, Graphics Card Power: 45W).
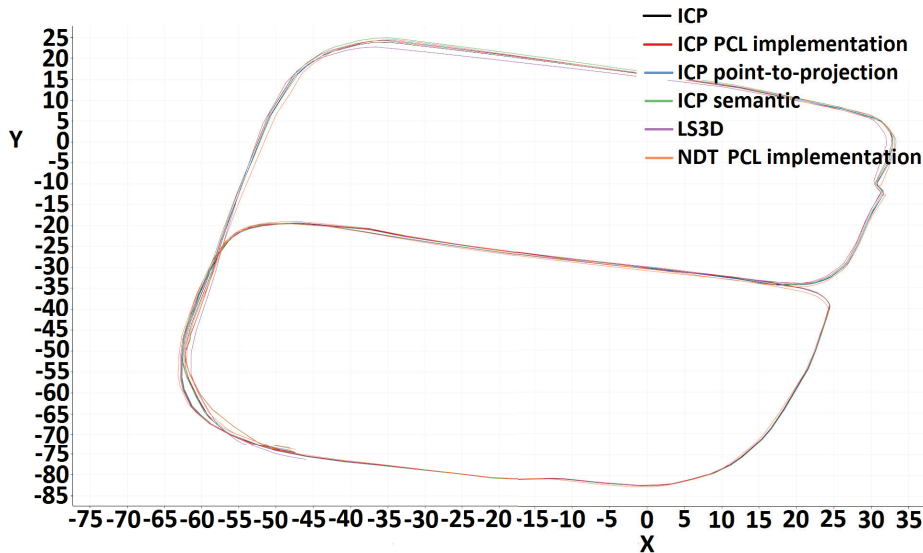

Figure 4. Visualisation of the trajectories obtained by different iterative registration methods.
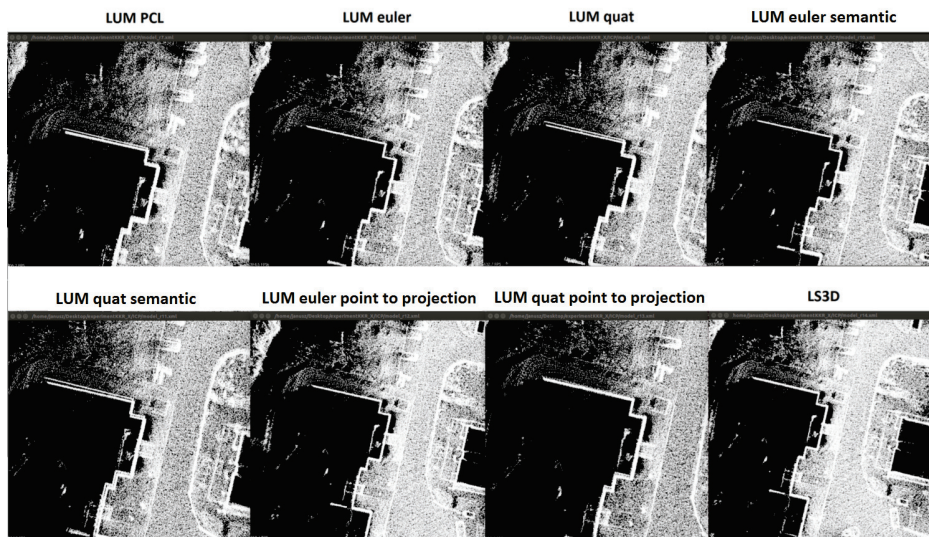
Figure 5. Visualisation of the final 3D maps obtained by different loop closing methods. Data were initially registered by iterative registration method ICP.
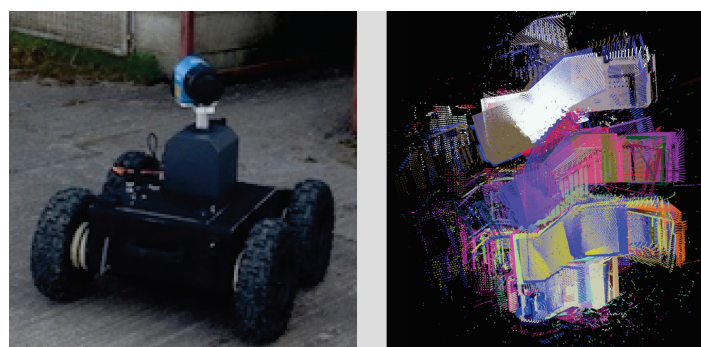
Figure 6. Left: drRobot mobile robotic platform equipped with 3D laser system, right: raw data from stairs experiment.

Table 1: Computation time of loop closing.

Method	Intel(R) Core(TM) i7-4820K CPU @ 3.70GHz, GeForce GTX TITAN X (CUDA cores: 3072, Graphics Clock: 1000 MHz, Graphics Card Power: 250W)	Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz, GeForce GT 650M (CUDA cores: 384, Graphics Clock: 900 MHz, Graphics Card Power: 45W)
LUM - PCL (25 iterations)	50.1 [s]	53.66 [s]
LUM euler (25 iterations)	76.3 [s]	145.1 [s]
LUM quat (25 iterations)	67.7 [s]	127,3 [s]
LUM euler semantic (25 iterations)	69.4 [s]	138.6 [s]
LUM quat semantic (25 iterations)	62.6 [s]	121.3 [s]
LUM euler point to projection (25 iterations)	118.8 [s]	255.2 [s]
LUM quat point to projection (25 iterations)	114.8 [s]	246 [s]
LS3D (25 iterations)	142.3 [s]	327 [s]
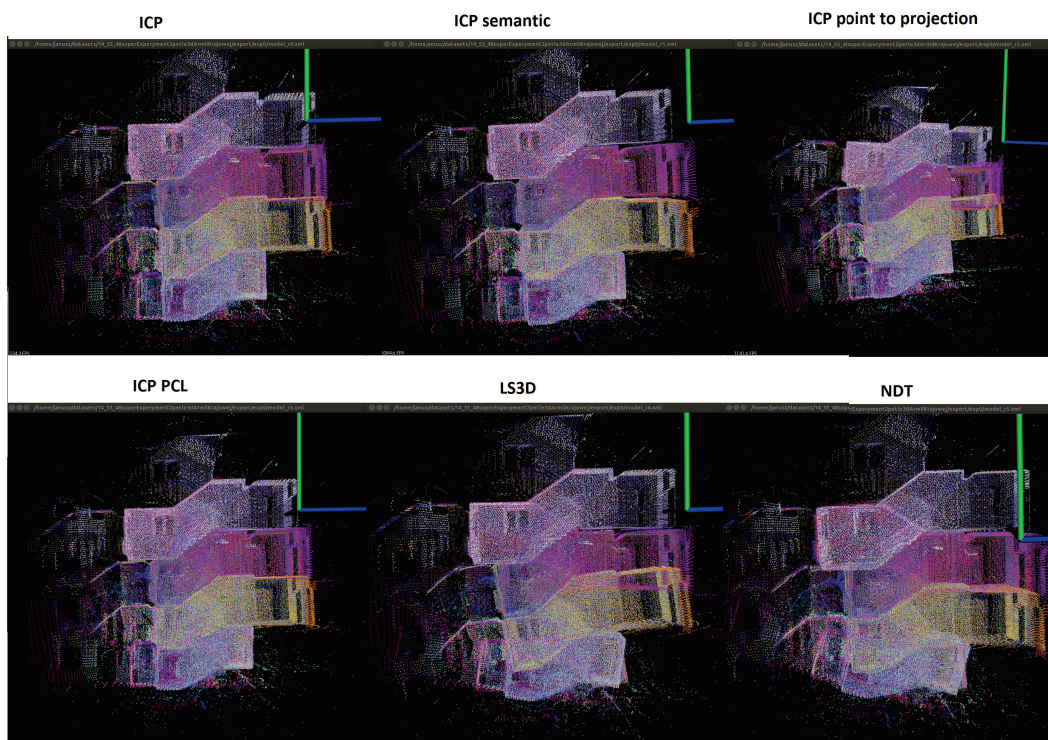
Figure 7. The result of iterative stairs data registration with different methods.

one for each of poses shown in figure 8. The parameters for each evaluated method were chosen to reach comparative results: nearest neighbourhood search radius: 1m, downsampling bucket size: 0.3m, number of iterations: 100, projection radius: 1m. The input data set is composed of 18 3D scans with initial poses given manually. This data was registered iteratively with ICP, ICP PCL (Point Cloud Library implementation), ICP point to projection, ICP semantic, LS3D, NDT (Point Cloud Library implementation). Then, the result of LS3D was chosen as input for final loop closing with LS3D, LUM euler, LUM euler point to projection, LUM euler semantic, LUM PCL (Point Cloud Library implementation), LUM quat, LUM quat point to projection, LUM quat semantic methods. Thus, figure 9 shows plots of errors: angle X, angle Y, angle Z for ICP, ICP PCL (Point Cloud Library implementation), ICP point to projection, ICP semantic, LS3D, NDT (Point Cloud Library implementation). Figure 10 shows plots of errors: X, Y, Z for ICP, ICP PCL (Point Cloud Library implementation), ICP point to projection, ICP semantic, LS3D, NDT (Point Cloud Library implementation). Figure 11 shows plots of errors: angle X, angle Y, angle Z for LS3D, LUM euler, LUM euler point to projection, LUM euler semantic, LUM PCL (Point Cloud Library implementation), LUM quat, LUM quat point to projection, LUM quat semantic. Figure 12 shows plots of errors: X, Y, Z for LS3D, LUM euler, LUM euler point to projection, LUM euler semantic, LUM PCL (Point Cloud Library implementation), LUM quat, LUM quat point to projection, LUM quat semantic. The observation is that Point Cloud Library implementation perform worse. Least Square Surface Matching performs well, thus it should be investigated

more for mobile robotic applications. An important conclusion is that the accuracy of automatic registration is close to the manual one.

5.   Conclusion

In this work, a set of 6DSLAM algorithms were evaluated with the focus on robotic mobile mapping system applications. The architecture of 6DSLAM algorithm is given thus, it makes possible to choose different registration and loop closing techniques. The algorithm is composed of the iterative registration component, hence ICP (Iterative Closest Point), ICP (point to projection), ICP with semantic discrimination of points, LS3D (Least Square Surface Matching), NDT (Normal Distribution Transform) can be chosen. Loop closing is based on different variations of LUM (quaternion, euler, semantic) and LS3D. The main focus was to evaluate the parallel programming approach and the semantic discrimination of points to improve the matching and the final 3D maps. The implementation is based on the state of the art 3DTK, PCL, and ROS frameworks improved by the novel parallel programming components. The experiments were discussed to show the advantages and disadvantages of certain methods in relation to practical mobile robotic applications. In the first experiment it was shown that the use of GeForce GTX TITAN X enables the on-line data registration by all evaluated methods. It is worth to mention that NDT based on PCL-CPU implementation can also produce on-line results. Algorithms: LUM PCL, LUM quat, LUM quat semantic are producing comparable maps that are less accurate than other methods. It is worth to mention that semantic approach and LS3D methods are in our opinion most promising algorithms for real-world tasks. Second experiment (robot moving down the stairs) demonstrates the advantage of semantic approach in relation to other techniques. It is important to emphasise that the other techniques tend to match floor with the ceiling. ICP semantic uses discrimination of points, thus matches between points classified as floor and points classified as ceiling are not considered in the optimisation. Thus, it improves the accuracy of the final map. The quantitative evaluation shows that the automatic data registration is close to the accuracy of the manual geodetic survey, thus autonomous mobile mapping systems can reach satisfactory performance and end-user acceptance in near future. To conclude, presented framework can help in solving the issue concerning which 6DSLAM method should be used for the certain application.

(a) Location of scenario.

(b) Object of interest.

(c) Mobile robot and geodetic equipment.
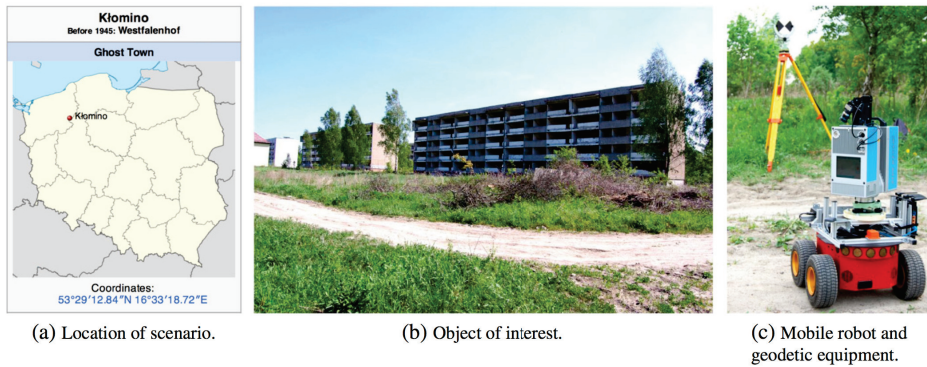
(a) Location of the experiment, photo of the realistic environment, mapping robotic system (mobile robot equipped with accurate 3D laser scanner Z+F IMAGER 5010).
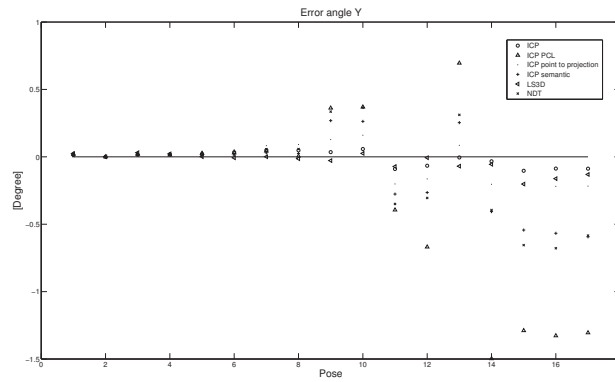

(a) Geodetic network.

(b) Numbered poses of registered scans
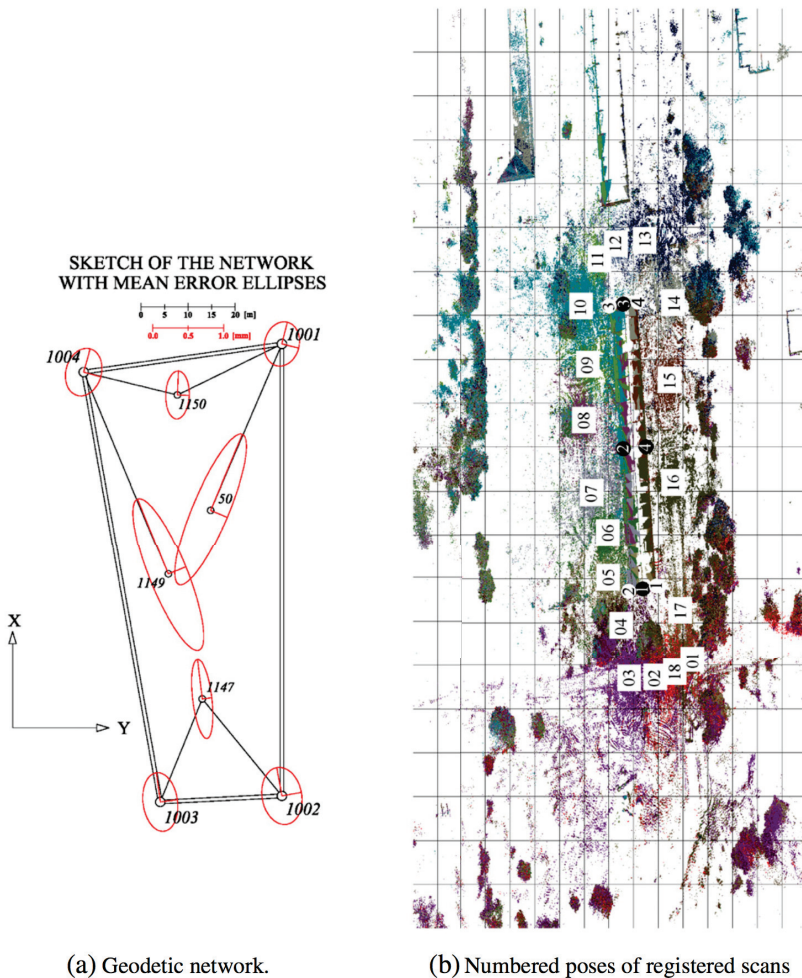
(b) Final control network with ellipses of accuracy, point cloud with 18 ground truth poses.

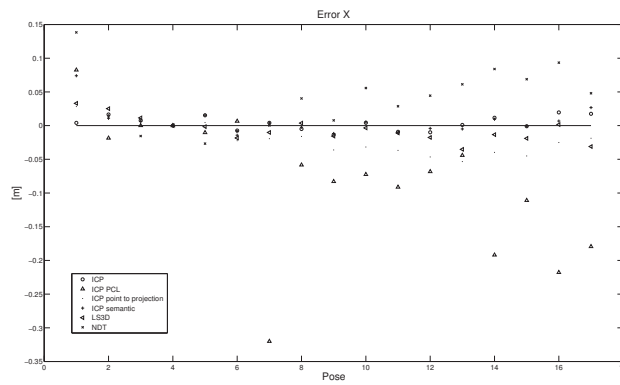**Figure 8**. Ground truth data set obtained with geodetic survey.
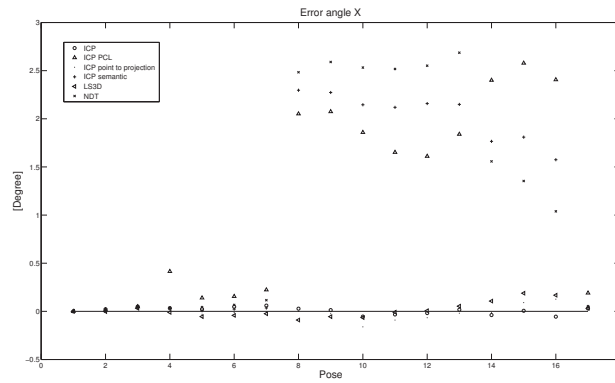
(a) Error angle X.

(b) Error angle Y.
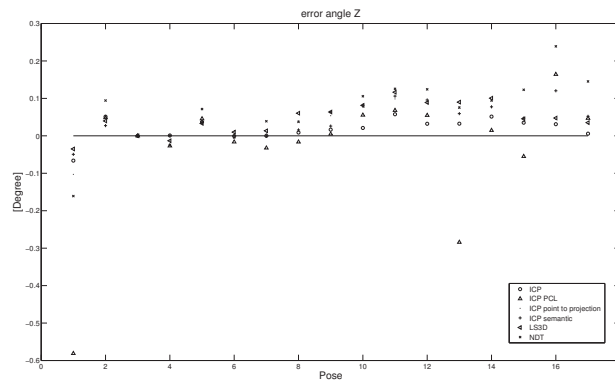
(c) Error angle Z.

Figure 9. Plots of errors: angle X, angle Y, angle Z for ICP, ICP PCL (Point Cloud Library implementation), ICP point to projection, ICP semantic, LS3D, NDT (Point Cloud Library implementation).
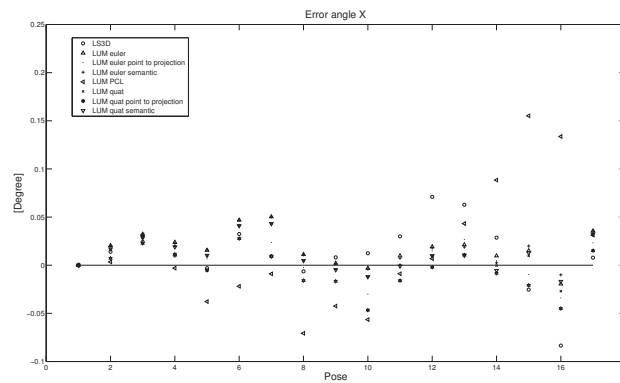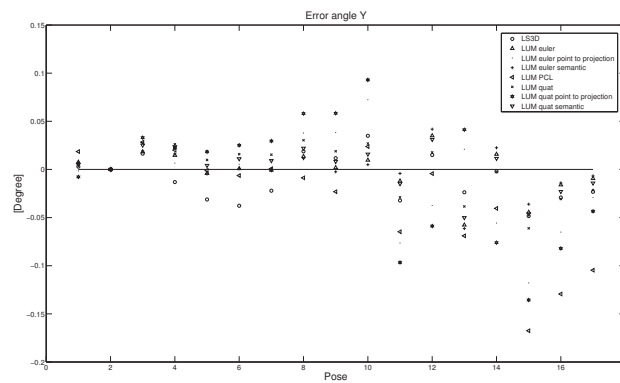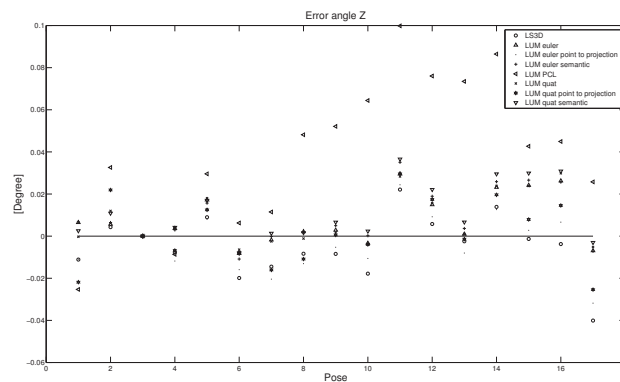
(a) Error X.
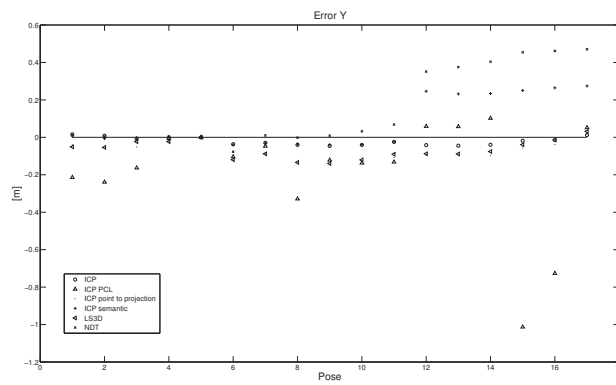
(b) Error Y.
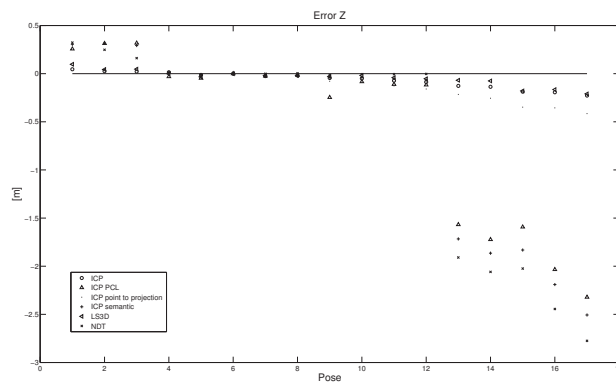
(c) Error Z.

Figure 10. Plots of errors: X, Y, Z for ICP, ICP PCL (Point Cloud Library implementation), ICP point to projection, ICP semantic, LS3D, NDT (Point Cloud Library implementation).
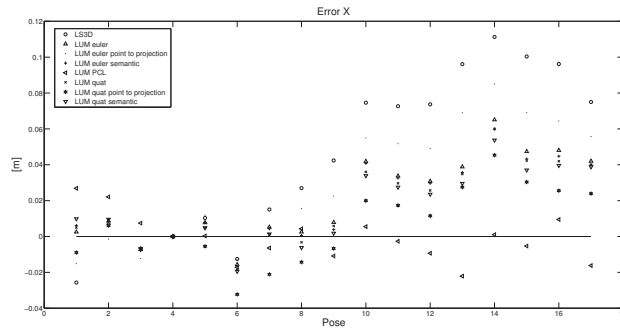
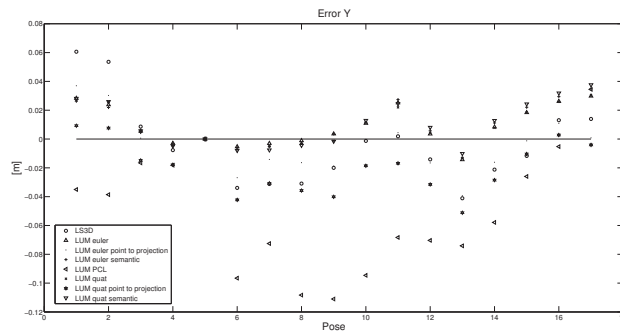(a) Error angle X.

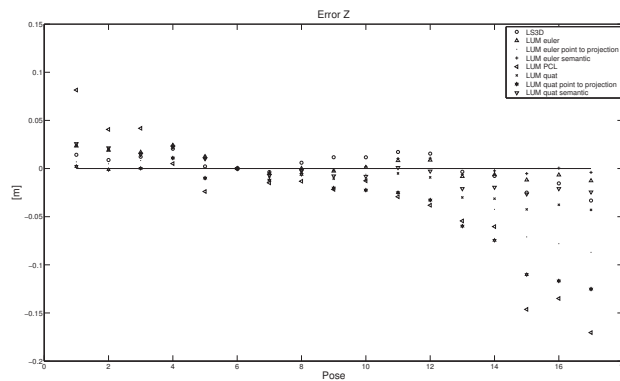
(b) Error angle Y.

(c) Error angle Z.

Figure 11. Plots of errors: angle X, angle Y, angle Z for LS3D, LUM euler, LUM euler point to projection, LUM euler semantic, LUM PCL (Point Cloud Library implementation), LUM quat, LUM quat point to projection, LUM quat semantic.

(a) Error X.

(b) Error Y.

(c) Error Z.

Figure 12. Plots of errors: X, Y, Z for LS3D, LUM euler, LUM euler point to projection, LUM euler semantic, LUM PCL (Point Cloud Library implementation), LUM quat, LUM quat point to projection, LUM quat semantic.

References

[1] *3DTK – The 3D Toolkit*, http://slam6d.sourceforge.net

[2] Akca D., Least Squares 3D surface matching. Ph.D. thesis, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, ISBN 3-906467-63-5, Mitteilungen Nr. 92, 78 pages, 2007.

[3] Akca D., Gruen A., Generalized Least Squares Multiple 3D Surface Matching, IAPRS Volume XXXVI, Part 3 / W52, 2007.

[4] Arun K. S., Huang T. S., Blostein S. D., Least-Squares Fitting of Two 3-D Point Sets, IEEE Trans. Pattern Anal. Mach. Intell., (9)5, pp. 698-700, 1987.

[5] Bedkowski J., Masłowski A., de Cubber G., Real time 3D localization and mapping for USAR robotic application, Industrial Robot, (39)5, pp. 464-474, 2012.

[6] Bedkowski J., Majek K., Nüchter A., General purpose computing on graphics processing units for robotic applications, Journal of Software Engineering for Robotics (JOSER) 4 (1), pp. 23-33, 2013.

[7] Bedkowski J., Majek K., Musialik P., Adamek A., Andrzejewski D., Czekaj D., Towards terrestrial 3D data registration improved by parallel programming and evaluated with geodetic precision, Automation in Construction Volume 47, pp. 78-91, 2014.

[8] Besl P. J., McKay N. D., A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14 (2), pp. 239-256, 1992.

[9] Biber P., Straßer W., The normal distributions transform: A new approach to laser scan matching. In Proc. IROS, volume 3, pp. 2743-2748, 2003.

[10] Borrmann D., Elseberg J., Lingemann K., Nüchter A., Hertzberg J., Globally Consistent 3D Mapping with Scan Matching. J. Robotics and Autonomous Sytems, 65(2), pp. 130-142, 2008.

[11] *CUDA – Compute Unified Device Architecture*, https://developer.nvidia.com/cuda-zone

[12] Elseberg J., Borrmann D., Nüchter A., Efficient Processing of Large 3D Point Clouds, in: Proceedings of the XXIII International Symposium on Information, Communication and Automation Technologies (ICAT11), Sarajevo, Bosnia, 2011.

[13] Elseberg J., Magnenat S., Siegwart R., Nüchter A., Comparison of nearest-neighbor-search strategies and implementations for efficient shape registration, Journal of Software Engineering for Robotics (JOSER) 3 (1), pp. 2-12, 2012.

[14] Horn B. K. P., Closed-form solution of absolute orientation using unit quaternions, Journal of the Optical Society of America, 4 (4), pp. 629-642, 1987.

[15] Horn B. K. P., Hilden H.M., Negahdaripour S., Closed-Form Solution of Absolute Orientation using Orthonormal Matrices, Journal of the Optical Society America, 5 (7), pp. 1127-1135, 1988.

[16] Lorusso A., Eggert D.W., Fisher R. B., A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations, 1995.

[17] Lu F., Milios E., Globally Consistent Range Scan Alignment for Environment Mapping. Autonomous Robots, 4, pp. 333-349, 1997.

[18] Magnusson M., Duckett T., A Comparison of 3D Registration Algorithms for Autonomous Underground Mining Vehicles, in: Proceedings of Eeuropean Conference on Mobile Robots ECMR, pp. 86-91, 2005.

[19] Magnusson M., Lilienthal A., Duckett T., Scan registration for autonomous mining vehicles using 3D-NDT. Journal of Field Robotics, 24(10), pp.803-827, 2007.

[20] Nüchter A., Surmann H., Lingemann K., Hertzberg J., Thrun S., 6D SLAM with an Application in Autonomous Mine Mapping, in: In Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1998-2003, 2004.

[21] Nüchter A., Lingemann K., Hertzberg J., Surmann H., 6D SLAM - 3D mapping outdoor environments, Journal of Field Robotics 24 (8-9), pp. 699-722, 2007.

[22] Nüchter A., Elseberg J., Schneider P., Paulus D., Study of parameterizations for the rigid body transformations of the scan registration problem, Comput. Vis. Image Underst., (114)8, pp. 963-980, 2010.

[23] *PCL - Point Cloud Library*, http://pointclouds.org

[24] Qiu D., May S., Nuechter A., GPU-Accelerated Nearest Neighbor Search for 3D Registration, in: Proceedings of the 7th International Conference on Computer Vision Systems, ICVS09, Springer-Verlag, Berlin, Heidelberg, pp. 194-203, 2009.

[25] *ROS - Robot Operating System*, http://www.ros.org

[26] Sprickerhof J., Nüchter A., Lingemann K., Hertzberg J., An Explicit Loop Closing Technique for 6D SLAM, in: I. Petrovic, A. J. Lilienthal (Eds.), Proceedings of the 4th European Conference on Mobile Robots, ECMR 09, September 23-25, 2009, Mlini/Dubrovnik, Croatia, KoREMA, pp. 229-234, 2009.

[27] Walker M. W., Shao L., Richard A. Volz, Estimating 3-D location parameters using dual number quaternions, CVGIP: Image Underst., (54)3, pp. 358-367, 1991.