Benchmark of multi-view Terrestrial Laser Scanning Point Cloud data registration algorithms

Janusz Będkowski

Institute of Fundamental Technological Research, Polish Academy of Science, ul. Pawińskiego 5b, Warsaw, 02-106, Poland

ARTICLE INFO

ABSTRACT

This study addresses multi-view Terrestrial Laser Scanning Point Cloud data registration methods. Multiple rigid point cloud data registration is mandatory for aligning all scans into a common reference frame and it is still considered a challenge looking from a large-scale surveys point of view. The goal of this work is to support the development of cutting-edge registration methods in geoscience and mobile robotics domains. This work evaluates 3 data sets of total 20 scenes available in the literature. This paper provides a novel open-source framework for multi-view Terrestrial Laser Scanning Point Cloud data registration benchmarks. The goal was to verify experimentally which registration variant can improve the open-source data looking from the quantitative and qualitative points of view. In particular, the following scanners provided measurement data: Z+F TLS Imager 5006i, Z+F TLS Imager 5010C, Leica ScanStation C5, Leica ScanStation C10, Leica P40 and Riegl VZ-400. The benchmark shows an impact of the metric e.g. point to point, point to projection onto a plane, plane to plane etc..., rotation matrix parameterization (Tait–Bryan, quaternion, Rodrigues) and other implementation variations (e.g. multi-view Normal Distributions Transform, Pose Graph SLAM approach) onto the multi-view data registration accuracy and performance. An open-source project is created and it can be used for improving existing data sets reported in the literature, it is the added value of the presented research. The combination of metrics, rotation matrix parameterization and optimization algorithms creates hundreds of possible approaches. It is shown that chosen metric is a dominant factor in data registration. The rotation parameterization and other degrees of freedom of proposed variants are rather negligible compared with chosen metric. Most of the proposed approaches improve registered reference data provided by other researchers. Only for 2 from 20 scenes it was not possible to provide significant improvement. The largest improvements are evident for large-scale scenes. The project is available and maintained at https://github.com/MapsHD/HDMapping.

1. Introduction

Multi-view TLS (Terrestrial Laser Scanner) data registration is essential for many applications and domains such as geoscience, mobile robotics and computer vision. It provides aligned point clouds. A gap between these domains is evident since geo-science claims having already mature technology for providing accurate TLS. Mobile robotics is rather focused on the high performance of real-time data registration systems than offline calculations. Many researchers from the mobile robotics domain [1] claim high accuracy of delivered maps, thus it was a prerequisite for conduced research closing the gap between those domains. Terrestrial Laser Scanning data is widely disseminated by many vendors and organizations [2,3]. An interesting data set [4] relates with 3D woody structure of large tropical trees. Recent work on large-scale TLS registration [5] and forest terrestrial point clouds [6] shows still the great interest in this topic since many challenges are still evident such

as large-scale problems and scanning non-urban environments. For this purpose proposed novel open-source framework provides all necessary tools for conducting large-scale multi-view TLS benchmarks.

This paper shows the novel complementary open-source framework that was not yet proposed nor elaborated on in the literature. This paper organizes state-of-the-art multi-view TLS data registration knowledge and extends it by the novel implementation of multi-view Normal Distributions Transform using regular grid decomposition and multi-threading for efficient calculations. Also, it provides a possibility to incorporate a general and adaptive robust loss function [7] for all multi-view registration variants. Thus, it is considered as a contribution of this work. It is evident that an alternative solution could be the K-means algorithm utilized to divide all data points into different clusters [8]. But, the proposed approach is based on lightweight implementation without a need of incorporating additional
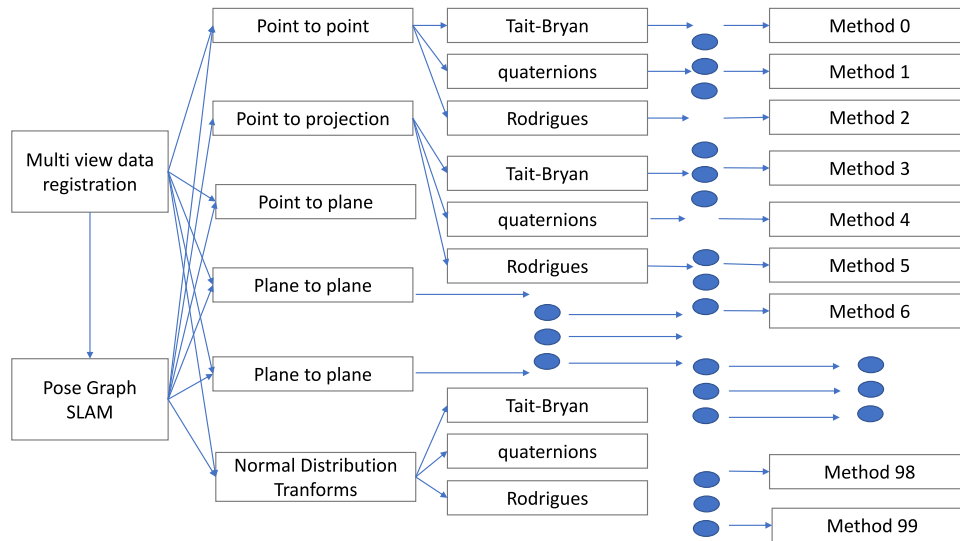
Fig. 1. The overview of the complementary multi-view TLS data registration benchmark. The combination of metrics, rotation matrix parameterization and optimization algorithms create hundreds of possible approaches for multi-view TLS (Terrestrial Laser Scanner) data registration.

programming libraries. The framework provides metrics such as point-to-point, point-to-projection onto a plane, and plane to plane etc..., thus the sum of the distances between basic geometric primitives (or their parameterized form) is minimized. Moreover, it is shown that it is possible to express SO(3) using Euler, Tait–Bryan, quaternion or Rodrigues' rotation formula, thus it provides for the first time in literature the possibility of observing the potential impact on the data registration. The optimization can be performed locally or globally using the weighted nonlinear least squares method, Gauss–Newton, Levenberg–Marquardt algorithms including robust variants and based on probabilistic approaches e.g. Normal Distributions Transform, Generalized Iterative Closest Point and Pose Graph Simultaneous Localization and Mapping. The combination of metrics, rotation matrix parameterization and optimization algorithms create hundreds of possible variants that are shown in Fig. 1. Thus, it is the most complementary open-source multi-view TLS data registration framework [9]. The goal of the conducted benchmark was to improve data provided by [10–12]. We consider a scenario when data are initially registered. For this reason, application with large initial errors and small overlap between scan [13] is rather out of the scope of this research. The results of this research can improve many applications such as accurate documentation in cultural heritage [14], environmental management [15], TLS performance evaluation [16], geology [17], forest measuring and modeling [18], forensics [19,20], mountain terrain mapping [21], space exploration [22] and protection systems [23]. It is possible to enumerate numerous other applications since TLS is with us for more than a decade. Numerous algorithms appeared [24], thus many researchers provide mature complementary surveys such as [25]. Providing registered reference data sets is a concern for many researchers [26] to conduct their research [27]. It is more and more popular in mobile robotics to provide accurate reference data for evaluation mapping capabilities [28]. For these purposes, this paper proposes a novel complementary open-source TLS data registration framework. It provides software tools for improving TLS data provided by other researchers. In that sense, this paper extends the state of the art by novel benchmark tools.

The importance of the work can be summarized as follows:

- it aggregates relevant multi-view TLS data registration methods in a common framework,
- the implementation does not require sophisticated programming skills, thus larger audience can benefit from this study,
- this work shows how to improve state-of-the-art data sets considered by other researchers as ground truth,

- it shows that the significant impact is related to metric (e.g. point to point, point to projection ...),
- this work shows that large-scale surveys are still a challenge, thus future work should consider scalable implementation.

The novelty of the paper is a complementary multi-view TLS data registration benchmark. This paper introduces a novel multi-view normal distributions transform algorithm with all necessary variations of rotation matrix parameterization. Most variants of lidar metrics (point to point, point to plane, plane to plane etc...) are incorporated into this benchmark. The quantitative and qualitative benchmark of a such complementary set of TLS data registration variants was not elaborated in the literature. The motivation behind this research was, on one hand, to provide an objective tool for TLS benchmark, on the other hand, to provide a tool for improving state-of-the-art data sets. The contribution of the paper is an evaluation based on 20 open-source datasets from 3 independent research sources. The following scanners provided data: Z+F TLS Imager 5006i, Z+F TLS Imager 5010C, Leica ScanStation C5, Leica ScanStation C10, Leica P40 and Riegl VZ-400. Such complementary and large-scale benchmark was not yet elaborated on in the literature. The novelty of the work can be summarized as follows:

- it introduces a scalable solution for large-scale surveys based on novel multi-view normal distributions transform algorithm,
- it provides relevant metrics for point cloud data registration,
- it elaborates on most significant data registration variants including different rotation matrix parameterization,
- this work was evaluated based on state-of-the-art data sets incorporating relevant TLS,
- the results are supported by 2000 experiments that can be easily reproduced by provided open-source implementation,
- this open-source implementation can be used for accurate reference data creation.

The paper is organized as follows: Section 2 encapsulates all necessary methods to build all TLS data registration algorithms. Section 3 shows the result of 2000 experiments performed on publicly available datasets and demonstrates the fact that in most cases proposed algorithms improve state of the art. Data sets are composed of challenging indoor and outdoor scenarios. The justification of the results is based on quantitative and qualitative measures widely used in literature. These datasets are widely used by other researchers as ground truth, therefore the impact of this research will improve the overall state of the art. The

Fig. 2. Three publicly available data sets are incorporated into the benchmark. First row: ETH [11], second row: RESSO [10], third and fourth rows: WHU_TLS [12].

main contribution of this research is an open-source project available and maintained at https://github.com/MapsHD/HDMapping capable of improving existing open-source large-scale data sets. Such wide scope TLS data registration implementation extends state of the art and has research and didactic potential. Section 4 summarizes the research.

2. Methodology

2.1. Manipulation of basic geometric primitives in 3D space

We consider points and planes as the basic primitives in this paper. Obviously, it is possible to extend this in future be a set of e.g. lines. To build an optimization system it is necessary to manipulate these geometric primitives in 3D space. For this reason we introduce WORLD to TLS $[\boldsymbol{R}, \boldsymbol{t}]_{W \rightarrow TLS}$ and TLS to WORLD $[\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}$ transformation matrices. These are 4×4 matrices $\in$ SE(3), where $\boldsymbol{R}$ is rotation matrix $\in$ SO(3) and $\boldsymbol{t} \in \mathbb{R}^3$ is a translation vector and an Eq. (1) shows an important relationship,

$$
\begin{bmatrix} \boldsymbol{R}_{W \leftarrow TLS} & \boldsymbol{t}_{W \leftarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix}
$$
$$
= \begin{bmatrix} \boldsymbol{R}_{W \rightarrow TLS} & \boldsymbol{t}_{W \rightarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix}^{-1}
$$
$$
= \begin{bmatrix} \boldsymbol{R}_{W \rightarrow TLS}^{\mathsf{T}} & -\boldsymbol{R}_{W \rightarrow TLS}^{\mathsf{T}} \boldsymbol{t}_{W \rightarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix} \quad (1)
$$

where 'T' is an operator of a transposed matrix. This relationship will play an important role within the fusion of our basic geometric primitives with the observation equations known from Computer Vision e.g. pinhole camera model re-projection error in Bundle Adjustment [29]. This information fusion is an interesting future research direction, for this reason, its impact on point cloud data registration is investigated in this paper.

The coordinates of the 3D point in Euclidean space are give as $\boldsymbol{P}^l = [x^l, y^l, z^l, 1]^{\mathsf{T}}$. Eq. (2) shows how to transform this point into a global reference frame as point $\boldsymbol{P}^g = [x^g, y^g, z^g, 1]^{\mathsf{T}}$. This is a fundamental concept to building an Iterative Closest Point algorithm [31].

$$
\boldsymbol{P}^g = [\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS} \boldsymbol{P}^l \quad (2)
$$

The plane in 3D space is represented uniquely by four parameters $[a, b, c, d]$, where $[a, b, c]$ is a unit vector orthogonal to the plane and

d is the distance from the origin. Eq. (3) shows how to transform plane expressed in local reference frame as $[a^l, b^l, c^l, d^l]$ to obtain parameters $[a^g, b^g, c^g, d^g]$ expressed in a global reference frame.

$$
\begin{bmatrix} a^g & b^g & c^g & d^g \end{bmatrix}
$$
$$
= \begin{bmatrix} a^l & b^l & c^l & d^l \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} & -\boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} \boldsymbol{t}_{W \leftarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix} \quad (3)
$$

Eqs. (2), (3) are fundamental concepts of transforming basic primitives in 3D space. The last important concept is the source point $\boldsymbol{P}^{src,g}$ to plane $[a, b, c, d]$ projection resulting $\boldsymbol{P}^{proj,g}$ given by Eq. (4),

$$
\boldsymbol{P}^{proj,g} = \begin{bmatrix} x^{src,g} \\ y^{src,g} \\ z^{src,g} \end{bmatrix} - \left(\begin{bmatrix} x^{src,g} & y^{src,g} & z^{src,g} & 1 \end{bmatrix} \cdot \begin{bmatrix} a^g & b^g & c^g & d^g \end{bmatrix} \right) \begin{bmatrix} a^g \\ b^g \\ c^g \end{bmatrix} \quad (4)
$$

where $(\cdot)$ is a dot product. Having in mind previously described concepts of basic geometric primitives manipulation in 3D space it is possible to build point-to-point, point-to-projection and plane-to-plane metrics. Plenty of other metrics exist and they will be discussed in Section 2.2.

2.2. Observation equations

An observation Eq. (5) relates a target value y_i, a model function $\boldsymbol{\Psi}_{[\beta]}(\boldsymbol{x}_i)$ and its *residual* r_i. Minimizing a difference between the target value and the model function allows defining the optimization problem as (6) that is described in detail in Section 2.4.

$$
\underbrace{r_i}_{residual} = \underbrace{y_i}_{target\ value} - \underbrace{\boldsymbol{\Psi}_{[\beta]}(\boldsymbol{x}_i)}_{model\ function} \quad (5)
$$
$$
\underbrace{\phantom{r_i = y_i - \boldsymbol{\Psi}}}_{observation\ equation}
$$

$$
\underbrace{\beta^* = \min_{\beta} \sum_{i=1}^{C} \left(y_i - \boldsymbol{\Psi}_{[\beta]}(\boldsymbol{x}_i) \right)^2}_{optimization\ problem} \quad (6)
$$

where, β is the vector of n adjustable parameters by the optimization process resulting β^* and $\boldsymbol{x}$ is the vector of input variables of the model function.

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 3. ETH data set [11], scene: arch. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is difficult to improve.

Table 1

Barron's adaptive robust loss function [7], $\rho(r, \alpha, c)$, $Y(r)$ - influence function, $w(r)$ - weight and reproduced loss function, (*) singularity.

Condition	$\rho(r)$	$Y(r)$	$w(r)$	Reproduced loss function
$\alpha = -\infty^*$	$1 - e^{-\frac{1}{2}\left(\frac{r}{c}\right)^2}$	$\frac{r}{c^2} e^{-\frac{1}{2}\left(\frac{r}{c}\right)^2}$	$\frac{1}{c^2} e^{-\frac{1}{2}\left(\frac{r}{c}\right)^2}$	Welsch
$\alpha = -2$	$\frac{2\left(\frac{r}{c}\right)^2}{\left(\frac{r}{c}\right)^2 + 4}$	$\frac{r}{c^2}\left(\frac{\left(\frac{r}{c}\right)^2}{4} + 1\right)^{-2}$	$\frac{1}{c^2}\left(\frac{\left(\frac{r}{c}\right)^2}{4} + 1\right)^{-2}$	Geman-McClure
$\alpha = 0^*$	$\log\left(\frac{1}{2}\left(\frac{r}{c}\right)^2 + 1\right)$	$\frac{2r}{r^2 + 2c^2}$	$\frac{2}{r^2 + 2c^2}$	Cauchy
$\alpha = 1$	$\sqrt{1 + \left(\frac{r}{c}\right)^2} - 1$	$\frac{r}{c^2}\left(\left(\frac{r}{c}\right)^2 + 1\right)^{-\frac{1}{2}}$	$\frac{1}{c^2}\left(\left(\frac{r}{c}\right)^2 + 1\right)^{-\frac{1}{2}}$	$L1 - L2$
$\alpha = 2^*$	$\frac{1}{2}\left(\frac{r}{c}\right)^2$	$\frac{r}{c^2}$	$\frac{1}{c^2}$	$L2$
otherwise	$\frac{|\alpha-2|}{\alpha}\left(\left(\frac{(\frac{r}{c})^2}{|\alpha-2|}\right)^{\frac{\alpha}{2}} - 1\right)$	$\frac{r}{c^2}\left(\frac{\left(\frac{r}{c}\right)^2}{|\alpha-2|} + 1\right)^{\left(\frac{\alpha}{2}-1\right)}$	$\frac{1}{c^2}\left(\frac{\left(\frac{r}{c}\right)^2}{|\alpha-2|} + 1\right)^{\left(\frac{\alpha}{2}-1\right)}$	generic

Table 2

The characteristics of conducted data sets ETH [11], RESSO [10] and WHU_TLS [12]. 'Name' is an unique identification of the data set, 'Source' indicates the publication with data set description, 'Scanner' indicates TLS type, 'Registration' indicates the method used for obtaining reference data, 'Nr of scans' denotes number of TLS stations, 'Nr of points' denotes summarized number of points for current data set, 'Nr of points after down-sampling' gives number of remaining points after down-sampling procedure, 'Dimensions (m)' describes scanned scene boundaries in meters, 'Environment organization' classifies scenes into structured or unstructured and give information concerning vegetation or other movable obstacles, 'Environment location' classifies scenes into indoor and outdoor, 'Overlap min (%)' denotes the minimum overlap between two scans, 'Overlap max (%)' denotes the maximum overlap between scans.

Name	Source	Scanner	Registration	Nr of scans	Nr of points	Nr of points after down-sampling	Dimensions (m)	Environment organization	Environment locations
exp1_arch	[11]	Z+F TLS Imager 5006i	ICP	5	137.114.356	2.678.325	x: 143, y: 165, z: 111	Structured (+vegetation)	Outdoor
exp2_courtyard	[11]	Z+F TLS Imager 5006i	ICP	8	108.348.634	2.737.986	x: 162, y: 147, z: 47	Unstructured	Outdoor
exp3_facade	[11]	Z+F TLS Imager 5006i	ICP	7	138.587.422	427.405	x: 54, y: 76, z: 21	Structured (+vegetation)	Outdoor
exp4_ office	[11]	Z+F TLS Imager 5006i	ICP	5	53.605.691	257.462	x: 20, y: 21, z: 25	Structured	Indoor
exp5_trees	[11]	Z+F TLS Imager 5006i	ICP	6	121.489.966	2.822.654	x:121, y: 122, z: 47	Unstructured (vegetation)	Outdoor
exp6_figure_7a	[10]	Leica ScanStation C10	ICP	12	6.416.789	4.987.983	x: 321, y: 299, z: 115	Structured (+vegetation)	Outdoor
exp7_figure_7b	[10]	Leica ScanStation C10	ICP	6	1.346.690	1.332.962	x: 426, y: 427, z: 176	Structured (+vegetation)	Indoor-Outdoor
exp8_figure_7c	[10]	Leica ScanStation C10	ICP	5	618.007	607.113	x: 240, y: 339, z: 22	Structured	Outdoor
exp9_figure_7d	[10]	Leica ScanStation C10	ICP	3	712.220	589.245	x: 334, y: 245, z: 151	Structured (+vegetation)	Outdoor
exp10_figure_7e	[10]	Leica ScanStation C10	ICP	26	4.192.172	3.808.582	x: 549, y: 563, z: 84	Structured(+vegetation)	Outdoor
exp11_1-SubwayStation	[12]	Z+F TLS Imager 5010C	[30]	6	237.573.133	654.433	x: 558, y: 270, z: 12	Structured	Indoor
exp12_10-UndergroundExcavation	[12]	VZ-400	[30]	12	311.349.879	9.081.757	x: 270, y: 309, z: 71	Unstructured	Indoor
exp13_11-Tunnel	[12]	VZ-400	[30]	6	157.018.478	1.255.305	x: 483, y: 308, z: 76	Structured	Indoor
exp14_2-HighSpeedRailway	[12]	VZ-400	[30]	8	49.863.303	4.288.896	x: 1095, y: 1193, z: 166	Structured (+moving cranes)	Outdoor
exp15_3-Mountain	[12]	Leica ScanStation C5	[30]	6	19.612.517	4.010.368	x: 349, y: 340, z: 162	Unstructured	Outdoor
exp16_5-Park	[12]	VZ-400	[30]	32	160.242.854	22.835.776	x: 886, y: 835, z: 166	Structured(+vegetation)	Outdoor
exp17_6-Campus	[12]	VZ-400	[30]	10	109.045.594	15.510.827	x: 799, y: 862, z: 219	Structured(+vegetation)	Outdoor
exp18_7-Residence	[12]	Leica P40	[30]	7	43.700.787	5.008.914	x: 482, y: 500, z: 102	Structured(+vegetation)	Outdoor (high buildings)
exp19_8-RiverBank	[12]	VZ-400	[30]	7	93.108.323	12.909.827	x: 694, y: 554, z: 166	Structured(+vegetation)	Outdoor (water)
exp20_9-HeritageBuilding	[12]	VZ-400	[30]	9	238.156.561	14.786.624	x: 646, y: 500, z: 163	Structured(+vegetation)	Outdoor

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
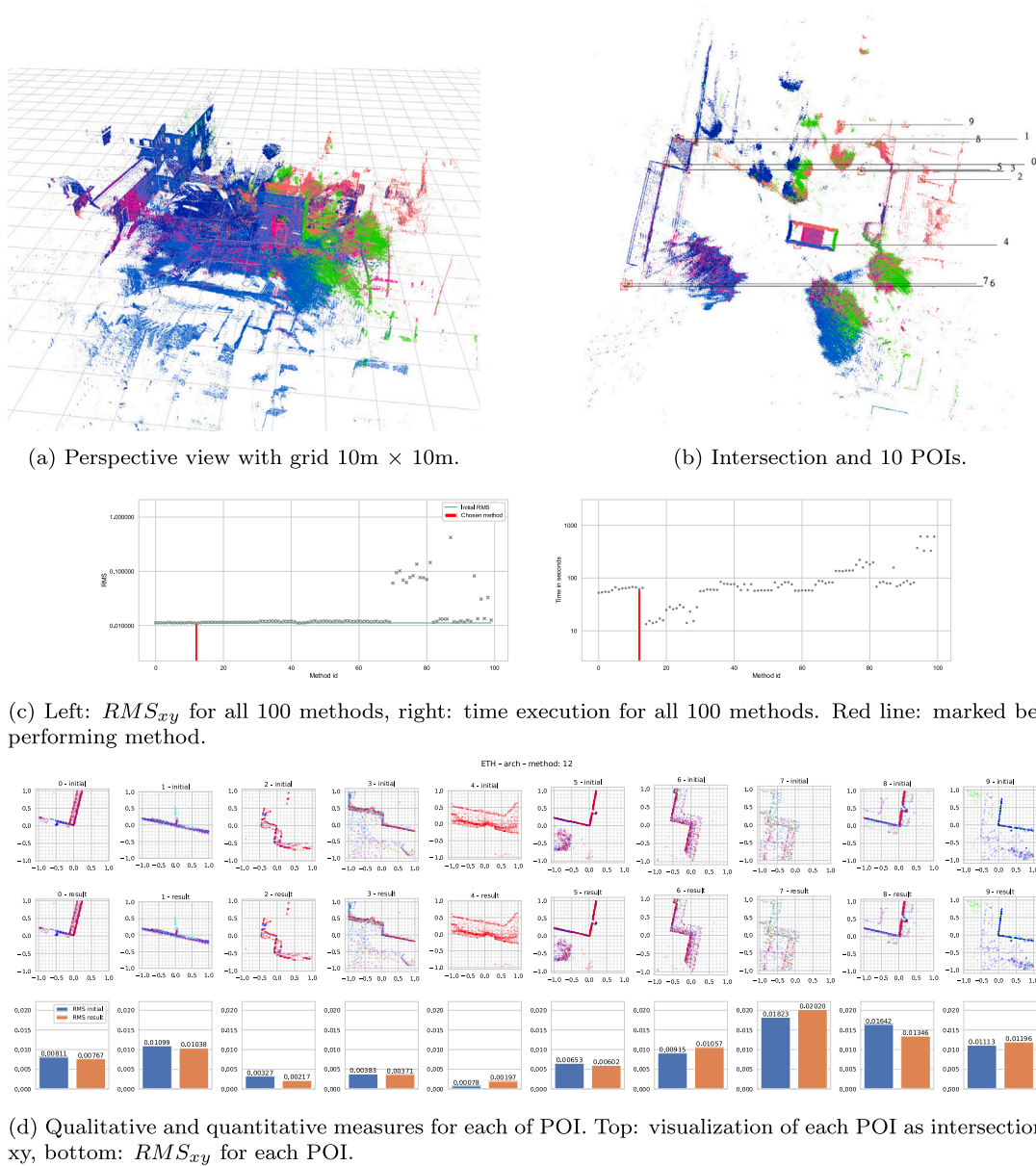
Fig. 4. ETH data set [11], scene: courtyard. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

Table 3
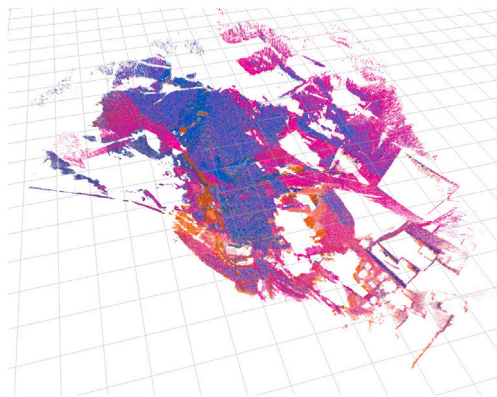Results for ETH dataset [11].

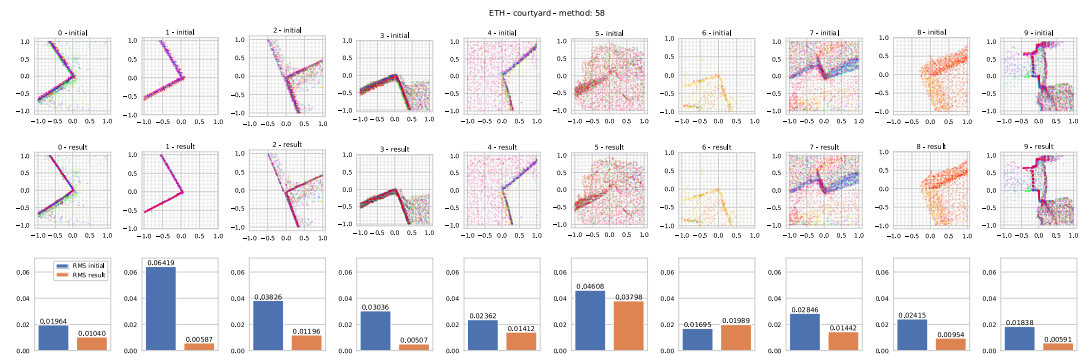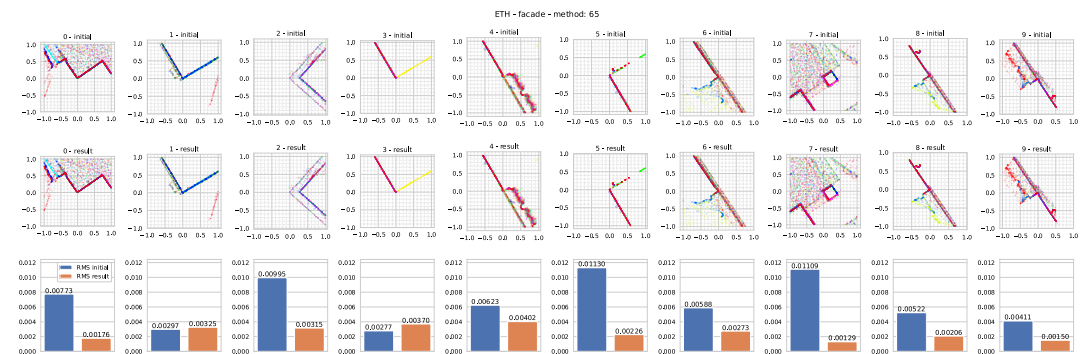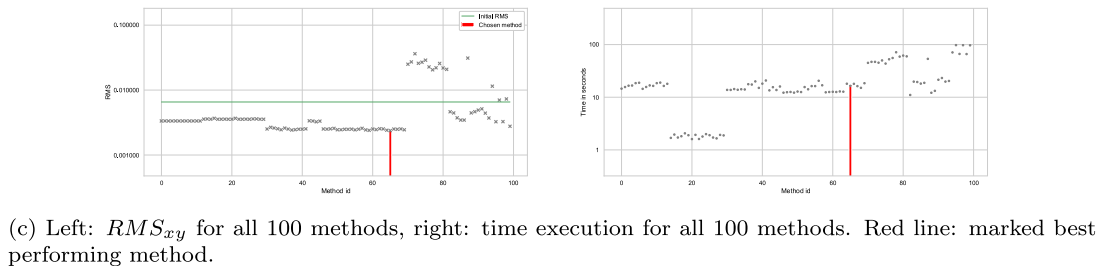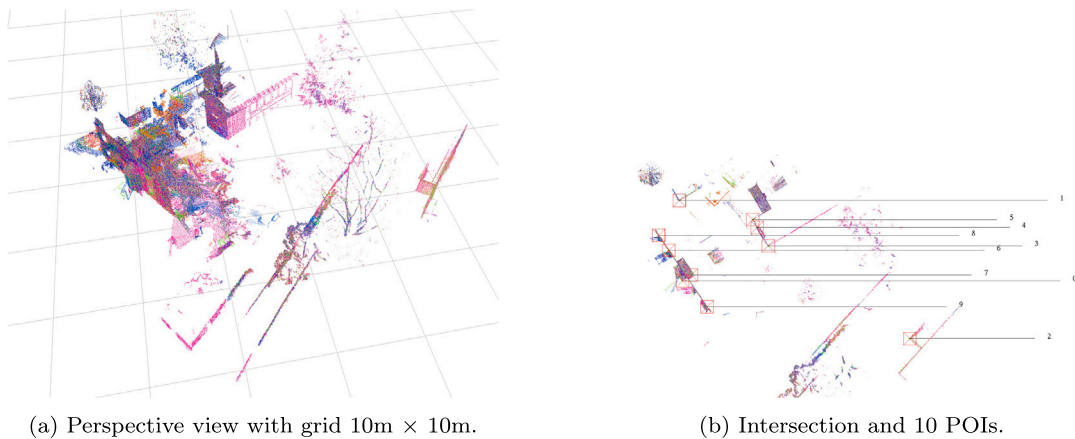| id	method	GN	LM	W ← TLS	W →TLS	TB	Q	R	leftJ	rightJ	arch RMS_{xy}	courtyard RMS_{xy}	facade RMS_{xy}	office RMS_{xy}	trees RMS_{xy}
-	initial[11]										0.011186	0.031796	0.006555	0.007657	0.026556
0	p2p	✓		✓			✓				0.011272	**0.023478**	**0.003345**	**0.007108**	**0.023778**
1	p2p	✓		✓		✓					0.011283	**0.023479**	**0.003350**	**0.007136**	**0.023782**
2	p2p	✓		✓				✓			0.011269	**0.023479**	**0.003344**	**0.007109**	**0.023778**
3	p2p	✓			✓	✓					0.011272	**0.023479**	**0.003345**	**0.007108**	**0.023778**
4	p2p	✓			✓		✓				0.011422	**0.023479**	**0.003349**	**0.007178**	**0.023786**
5	p2p	✓			✓			✓			0.011269	**0.023479**	**0.003344**	**0.007109**	**0.023778**
6	p2p		✓	✓		✓					0.011272	**0.023479**	**0.003345**	**0.007108**	**0.023778**
7	p2p		✓	✓			✓				0.011283	**0.023479**	**0.003350**	**0.007136**	**0.023782**
8	p2p		✓	✓				✓			0.011269	**0.023479**	**0.003344**	**0.007109**	**0.023778**
9	p2p		✓		✓	✓					0.011272	**0.023479**	**0.003345**	**0.007108**	**0.023778**
10	p2p		✓		✓		✓				0.011422	**0.023479**	**0.003349**	**0.007178**	**0.023786**
11	p2p		✓		✓			✓			0.011269	**0.023479**	**0.003344**	**0.007109**	**0.023778**
12	p2p	✓		✓				✓	✓		**0.011116**	0.021706	0.003567	0.007326	0.023643
13	p2p	✓		✓				✓		✓	0.011203	0.021853	0.003590	0.007122	0.023698
14	ndt	✓		✓		✓					0.011519	**0.016503**	**0.003555**	0.007657	**0.023922**
15	ndt	✓		✓			✓				0.011473	1.626930	**0.003670**	**0.007645**	**0.023915**
16	ndt	✓		✓				✓			0.011512	**0.016504**	**0.003552**	**0.007623**	**0.023919**
17	ndt	✓			✓	✓					0.011518	**0.016502**	**0.003555**	0.007657	**0.023921**
18	ndt	✓			✓		✓				0.011479	2.271770	**0.003566**	0.007683	**0.023897**
19	ndt	✓			✓			✓			0.011511	**0.016504**	**0.003552**	**0.007623**	**0.023919**
20	ndt		✓	✓		✓					0.011519	**0.016503**	**0.003555**	**0.007646**	**0.023922**
21	ndt		✓	✓			✓				0.011476	1.669740	**0.003670**	**0.007655**	**0.023915**
22	ndt		✓	✓				✓			0.011513	**0.016504**	**0.003552**	**0.007616**	**0.023919**
23	ndt		✓		✓	✓					0.011519	**0.016502**	**0.003555**	0.007657	**0.023921**
24	ndt		✓		✓		✓				0.011479	2.724490	**0.003566**	0.007674	**0.023897**
25	ndt		✓		✓			✓			0.011511	**0.016504**	**0.003552**	0.007617	**0.023919**
26	ndt	✓		✓				✓	✓		0.011515	**0.016514**	**0.003599**	0.008061	**0.023914**
27	ndt		✓	✓				✓	✓		0.011515	**0.016514**	**0.003599**	0.008061	**0.023914**
28	ndt	✓		✓				✓		✓	0.011512	**0.016504**	**0.003552**	**0.007625**	**0.023919**
29	ndt		✓	✓				✓		✓	0.011511	**0.016504**	**0.003552**	**0.007623**	**0.023919**
30	p2proj	✓		✓		✓					0.011613	**0.016540**	**0.002516**	**0.007356**	**0.023309**
31	p2proj	✓		✓			✓				0.012098	**0.016542**	**0.002664**	**0.007180**	**0.023352**
32	p2proj	✓		✓				✓			0.012030	**0.016801**	**0.002601**	**0.007425**	**0.023083**
33	p2proj	✓			✓	✓					0.011915	**0.016590**	**0.002537**	0.007661	**0.023334**
34	p2proj	✓			✓		✓				0.012157	**0.016704**	**0.002455**	**0.007430**	**0.023216**
35	p2proj	✓			✓			✓			0.012095	**0.016626**	**0.002596**	**0.007116**	**0.023174**
36	p2proj		✓	✓		✓					0.012112	**0.016567**	**0.002508**	**0.007143**	**0.023274**
37	p2proj		✓	✓			✓				0.011637	**0.016602**	**0.002424**	**0.007342**	**0.023289**
38	p2proj		✓	✓				✓			0.012046	**0.016504**	**0.002460**	**0.007214**	**0.023243**
39	p2proj		✓		✓	✓					0.011913	**0.016523**	**0.002495**	**0.007197**	**0.023398**
40	p2proj		✓		✓		✓				0.011983	**0.016571**	**0.002499**	**0.007223**	**0.023218**
41	p2proj		✓		✓			✓			0.011787	**0.016574**	**0.002531**	**0.007279**	**0.023241**
42	p2proj	✓		✓				✓	✓		**0.011156**	0.022832	0.003363	0.007022	0.023584
43	p2proj		✓	✓				✓	✓		**0.011158**	0.023105	0.003337	0.007496	0.023498
44	p2proj	✓		✓				✓		✓	0.011288	0.022592	0.003249	0.007169	0.023689
45	p2proj		✓	✓				✓		✓	0.011521	0.023268	0.003323	0.007094	0.023716
46	p2pl	✓		✓		✓					0.011894	**0.016566**	**0.002517**	**0.007245**	**0.023301**
47	p2pl	✓		✓			✓				0.012316	**0.016621**	**0.002500**	0.007649	**0.023289**
48	p2pl	✓		✓				✓			0.011751	**0.016537**	**0.002527**	**0.007054**	**0.023248**
49	p2pl	✓			✓	✓					0.011982	**0.016557**	**0.002566**	**0.007484**	**0.023172**
50	p2pl	✓			✓		✓				0.012134	**0.016590**	**0.002452**	**0.007300**	**0.023273**
51	p2pl	✓			✓			✓			0.011989	**0.016685**	**0.002453**	**0.007095**	**0.023046**
52	p2pl		✓	✓		✓					0.011854	**0.016567**	**0.002487**	0.007628	**0.023276**
53	p2pl		✓	✓			✓				0.012061	**0.016598**	**0.002484**	**0.007185**	**0.023250**
54	p2pl		✓	✓				✓			0.012067	**0.016628**	**0.002502**	**0.007441**	**0.023269**
55	p2pl		✓		✓	✓					0.011793	**0.016620**	**0.002442**	**0.007432**	**0.023220**
56	p2pl		✓		✓		✓				0.012270	**0.016649**	**0.002497**	**0.007331**	**0.023263**
57	p2pl		✓		✓			✓			0.012187	**0.016600**	**0.002574**	**0.007164**	**0.023252**
58	dp2pl	✓		✓		✓					0.011877	**0.016441**	**0.002439**	**0.007141**	**0.023307**
59	dp2pl	✓		✓			✓				0.011892	**0.016577**	**0.002393**	**0.007425**	**0.023089**
60	dp2pl	✓		✓				✓			0.012142	**0.016642**	**0.002515**	**0.007086**	**0.023313**
61	dp2pl	✓			✓	✓					0.011878	**0.016530**	**0.002467**	**0.007212**	**0.023238**
62	dp2pl	✓			✓		✓				0.012030	**0.016641**	**0.002544**	**0.007392**	**0.023121**
63	dp2pl	✓			✓			✓			0.011856	**0.016564**	**0.002522**	**0.007495**	**0.023210**
64	dp2pl		✓	✓		✓					0.011797	**0.016581**	**0.002408**	0.007762	**0.023255**
65	dp2pl		✓	✓			✓				0.012031	**0.016592**	**0.002383**	**0.007024**	**0.023391**
66	dp2pl		✓	✓				✓			0.011724	**0.016521**	**0.002511**	**0.007402**	**0.023239**
67	dp2pl		✓		✓	✓					0.011761	**0.016580**	**0.002488**	**0.007409**	**0.023206**
68	dp2pl		✓		✓		✓				0.012161	**0.016490**	**0.002522**	**0.007208**	**0.023246**
69	dp2pl		✓		✓			✓			0.011664	**0.016566**	**0.002448**	**0.007118**	**0.023270**
70	pl2pl	✓		✓		✓					0.060680	0.267633	0.025003	0.037953	0.131051
71	pl2pl	✓		✓			✓				0.094475	0.281467	0.027158	0.024108	0.194567
72	pl2pl	✓		✓				✓			0.101896	0.218077	0.036335	0.034041	0.166269
73	pl2pl	✓			✓	✓					0.068895	0.308047	0.025806	0.045773	0.152414
74	pl2pl	✓			✓		✓				0.062238	0.283095	0.026838	0.039157	0.163441
75	pl2pl	✓			✓			✓			0.077554	0.234948	0.028862	0.039278	0.137703
76	pl2pl		✓	✓		✓					0.083122	0.239665	0.022746	0.054528	0.110537
77	pl2pl		✓	✓			✓				0.135797	0.291295	0.020539	0.042264	0.200097
78	pl2pl		✓	✓				✓			0.077085	0.251072	0.022074	0.048644	0.140942
79	pl2pl		✓		✓	✓					0.075692	0.254073	0.025667	0.038568	0.128421
80	pl2pl		✓		✓		✓				0.070831	0.246029	0.021881	0.034479	0.098260
81	pl2pl		✓		✓			✓			0.144215	0.242653	0.020809	0.032035	0.110849
82	pgslam(ndt)	✓		✓		✓					0.011512	**0.019973**	**0.004639**	0.009282	**0.024781**
83	pgslam(p2p)	✓		✓		✓					0.011976	**0.024218**	**0.004442**	**0.007177**	**0.023983**
84	pgslam(p2proj)	✓		✓		✓					0.013214	**0.017919**	**0.003740**	**0.007645**	**0.024090**
85	pgslam(p2pl)	✓		✓		✓					0.013165	**0.019874**	**0.003454**	0.007674	**0.024233**
86	pgslam(dp2pl)	✓		✓		✓					0.013144	**0.017546**	**0.003451**	0.007674	**0.023677**
87	pgslam(pl2pl)	✓		✓		✓					0.418844	0.140561	0.031086	0.029753	0.438444
88	pgslam(ndt, left-J)	✓		✓		✓			✓		0.011721	**0.019911**	**0.004450**	0.009359	**0.024517**
89	pgslam(ndt, right-J)	✓		✓		✓				✓	0.011527	**0.019964**	**0.004652**	0.009270	**0.024797**
90	pgslam(p2p, left-J)	✓		✓		✓			✓		0.012148	**0.022227**	**0.004916**	0.007659	**0.023223**
91	pgslam(p2p, right-J)	✓		✓		✓				✓	0.011701	**0.022411**	**0.005136**	**0.007183**	**0.023928**
92	pgslam(p2proj, left-J)	✓		✓		✓			✓		0.012551	**0.026863**	**0.004412**	**0.007364**	**0.022992**
93	pgslam(p2proj, right-J)	✓		✓		✓				✓	0.011916	**0.025033**	**0.003715**	**0.007337**	**0.023953**
94	pgslam(ndt, pcl)	✓		✓				✓			0.082232	**0.020760**	0.011473	0.014862	0.054945
95	pgslam(icp, pcl)	✓		✓				✓			0.013413	**0.019929**	**0.003259**	**0.007091**	**0.024220**
96	pgslam(ndt, pcl, GTSAM)	✓		✓				✓		✓	0.030720	**0.026071**	0.007008	0.010001	0.041788
97	pgslam(icp, pcl, GTSAM)	✓		✓				✓		✓	0.013443	**0.019926**	**0.003276**	**0.007091**	**0.024221**
98	pgslam(ndt, pcl, manif)	✓		✓				✓		✓	0.032996	**0.023742**	0.007299	0.008524	0.034447
99	pgslam(icp, pcl, manif)	✓		✓				✓		✓	0.012644	**0.020485**	**0.002767**	**0.006965**	**0.023551**

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 5. ETH data set [11], scene: facade. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

2.2.1. Point to point

Point to point observation equation incorporates model function given by (7),

$$\Psi_{[R,t]_{W \leftarrow TLS}}(R_{W \leftarrow TLS}, t_{W \leftarrow TLS}, P^l, P^{nn,g}) = [R,t]^{3 \times 4}_{W \leftarrow TLS} P^l - P^{nn,g} \quad (7)$$

where $P^{nn,g}$ is a nearest neighbor to P^l expressed in global reference frame as $[x^{nn,g}, y^{nn,g}, z^{nn,g}]^\top$. Thus, point to point observation equation is given in form of (8),

$$\underbrace{[\delta^{3 \times 1}]}_{residuals} = \underbrace{[0^{3 \times 1}]}_{target\ values} - \underbrace{([R,t]^{3 \times 4}_{W \leftarrow TLS} P^l - P^{nn,g})}_{model\ function} \quad (8)$$

where target value (our expectation) is $[0^{3 \times 1}]$ interpreted as no difference between coordinates of nearest neighbors. This difference is denoted as residuals $[\delta^{3 \times 1}]$.

2.2.2. Point to projection

Point to projection observation equation incorporates $P^{proj,g}$ from Eq. (4) as replacement of nearest neighbor $P^{nn,g}$ from (8), thus it

become (9).

$$\underbrace{[\delta^{3 \times 1}]}_{residuals} = \underbrace{[0^{3 \times 1}]}_{target\ values} - \underbrace{([R,t]^{3 \times 4}_{W \leftarrow TLS} P^l - P^{proj,g})}_{model\ function} \quad (9)$$

2.2.3. Point to plane
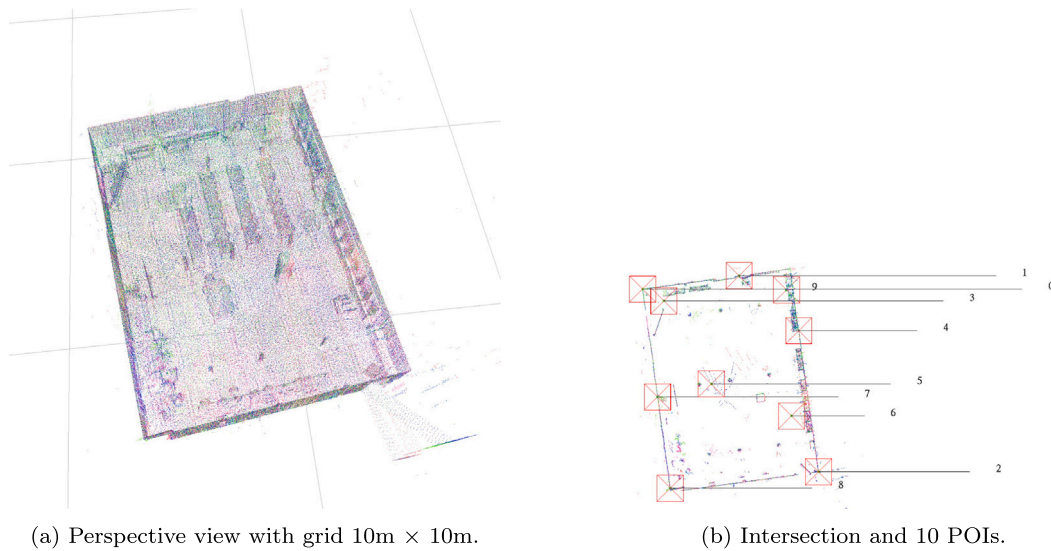
The condition that point $P^g = [x^g, y^g, z^g]^\top$ lies on the plane $[a^g, b^g, c^g, d^g]$ is using also point $P^{pl,g} = [x^{pl,g}, y^{pl,g}, z^{pl,g}]^\top \neq P^g$ belonging to this plane and it is expressed as (10). It means that vector $[a^g, b^g, c^g]^\top$ and $[P^g - P^{pl,g}]$ are orthogonal.

$$\begin{bmatrix} a^g & b^g & c^g \end{bmatrix} \begin{bmatrix} P^g - P^{pl,g} \end{bmatrix} = 0 \quad (10)$$

Thus, assuming $P^g = [R,t]^{3 \times 4}_{W \leftarrow TLS} P^l$ model function is expressed as (11).

$$\Psi_{[R,t]_{W \leftarrow TLS}}(R_{W \leftarrow TLS}, t_{W \leftarrow TLS}, P^l, a^g, b^g, c^g, P^{pl,g})$$
$$= \begin{bmatrix} a^g & b^g & c^g \end{bmatrix} \begin{bmatrix} [R,t]^{3 \times 4}_{W \leftarrow TLS} P^l - P^{pl,g} \end{bmatrix} \quad (11)$$

Point to plane observation equation is given as (12) with the target value $[0^{1 \times 1}]$ corresponding to the geometrical orthogonality condition.

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 6. ETH data set [11], scene: office. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

$$\underbrace{\left[\boldsymbol{\delta}^{1\times1}\right]}_{residual} = \underbrace{\left[\boldsymbol{0}^{1\times1}\right]}_{target\ value} - \underbrace{\left[a^g \quad b^g \quad c^g\right]\left[[\boldsymbol{R},\boldsymbol{t}]^{3\times4}_{W\leftarrow TLS}\boldsymbol{P}^l - \boldsymbol{P}^{pl,g}\right]}_{model\ function} \quad (12)$$

2.2.4. Distance point to plane

Distance of point $\boldsymbol{P}^l = [x^l, y^l, z^l, 1]^{\mathsf{T}}$ to plane $[a^g, b^g, c^g, d^g]$ is straight forward (13) and it is considered as model function. Thus, the distance point to plane observation equation is given in (14).

$$\boldsymbol{\Psi}_{[\boldsymbol{R},\boldsymbol{t}]_{W\leftarrow TLS}}(\boldsymbol{R}_{W\leftarrow TLS}, \boldsymbol{t}_{W\leftarrow TLS}, \boldsymbol{P}^l, a^g, b^g, c^g, d^g)$$

$$= \begin{bmatrix} a & b & c & d \end{bmatrix} \begin{bmatrix} x^g \\ y^g \\ z^g \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} a & b & c & d \end{bmatrix}([\boldsymbol{R},\boldsymbol{t}]^{4\times4}_{W\leftarrow TLS} \begin{bmatrix} x^l \\ y^l \\ z^l \\ 1 \end{bmatrix}) \quad (13)$$

$$\underbrace{\left[\boldsymbol{\delta}^{1\times1}\right]}_{residual} = \underbrace{\left[\boldsymbol{0}^{1\times1}\right]}_{target\ value} - \underbrace{\left[a \quad b \quad c \quad d\right]([\boldsymbol{R},\boldsymbol{t}]^{4\times4}_{W\leftarrow TLS} \begin{bmatrix} x^l \\ y^l \\ z^l \\ 1 \end{bmatrix})}_{model\ function} \quad (14)$$

2.2.5. Plane to plane

Plane to plane observation equation produces four residuals. It adopts Eq. (3) and corresponding nearest plane $[a^{nn,g}, b^{nn,g}, c^{nn,g}, d^{nn,g}]^{\mathsf{T}}$
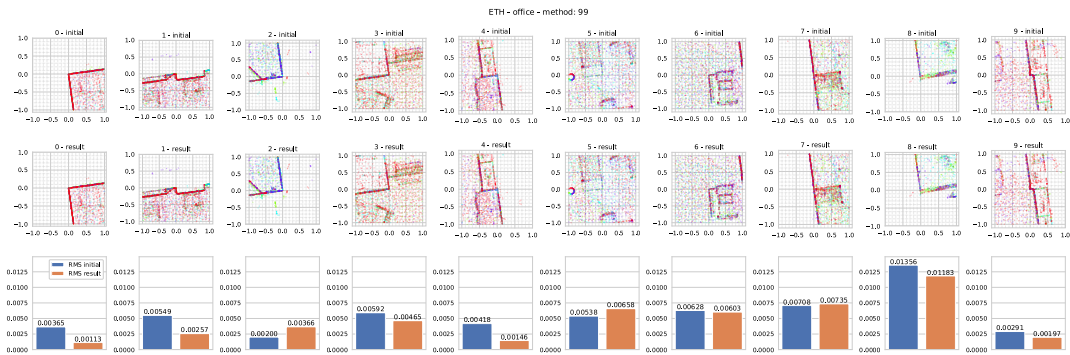
(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 7. ETH data set [11], scene: trees. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is slightly improved.

expressed in global frame to form model function as (15). Thus, plane to plane observation equation is given in (16).

$$\boldsymbol{\Psi}_{[\boldsymbol{R},\boldsymbol{t}]_{W \leftarrow TLS}}(\boldsymbol{R}_{W \leftarrow TLS}, \boldsymbol{t}_{W \leftarrow TLS}, a^g, b^g, c^g, d^g, a^{nn,g}, b^{nn,g}, c^{nn,g}, d^{nn,g})$$

$$= \begin{bmatrix} a^l & b^l & c^l & d^l \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} & -\boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} \boldsymbol{t}_{W \leftarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix} - \begin{bmatrix} a^{nn,g} \\ b^{nn,g} \\ c^{nn,g} \\ d^{nn,g} \end{bmatrix} \quad (15)$$

$$\underbrace{\begin{bmatrix} \boldsymbol{\delta}^{4 \times 1} \end{bmatrix}}_{residuals}$$

$$= \underbrace{\begin{bmatrix} \boldsymbol{0}^{4 \times 1} \end{bmatrix}}_{target\ values} - (\underbrace{\begin{bmatrix} a^l & b^l & c^l & d^l \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} & -\boldsymbol{R}_{W \leftarrow TLS}^{\mathsf{T}} \boldsymbol{t}_{W \leftarrow TLS} \\ \boldsymbol{0}^{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} a^{nn,g} \\ b^{nn,g} \\ c^{nn,g} \\ d^{nn,g} \end{bmatrix}}_{model\ function})$$

$$(16)$$

2.3. Constraints

This section elaborates on crucial constraints looking at building an optimization system. To keep the orthonormality of the rotation matrix expressed as quaternion the norm of the quaternion observation equation is added for each optimized pose. The fixed optimized parameter enables maintaining values of the chosen state vectors during the optimization process. Finally, the relative pose observation equation enables building Pose Graph SLAM [32,33] an alternative approach for multi-view point cloud data registration.

2.3.1. Norm of the quaternion

This constraint maintains the norm of unit quaternion $q(q_0, q_1, q_2, q_3)$. The condition $\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1$ results an observation Eq. (17)

$$\underbrace{\delta}_{residual} = \underbrace{1}_{target\ value} - \underbrace{\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}}_{model\ function} \quad (17)$$

where δ is *residual*, 1 is *target value* and $\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$ is *model function*.

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 8. RESSO data set [10], scene: figure7a. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

2.3.2. Fixed optimized parameter

This constraint maintains the value of the optimized parameter. Thus, the desired behavior is that it will not change during the optimization process. Typically, the first pose in Pose Graph SLAM is set to be fixed. In this case, the result of the *model function* equals *target value*, therefore *residual* suppose to be 0. Eq. (18) shows the fixed optimized parameter observation equation.

$$\underbrace{\delta}_{residual} = \underbrace{\beta^{target}}_{target\ value} - \underbrace{\boldsymbol{\Psi}_{[\beta]}}_{model\ function} = \beta^{target} - \beta = 0 \tag{18}$$

It can be seen that *model function* is expressed as $\boldsymbol{\Psi}_{[\beta]} = \beta$, the *target value* $\beta^{target} = \beta$.

2.3.3. Relative pose

This constraint is a fundamental concept of Pose Graph SLAM. It allows modeling the relations between measurements and observations in form of a graph [34], where nodes are related with poses end edges that relate the difference between measures and observations. Relative pose $[\boldsymbol{R}, t]_{W \leftarrow TLS,12}$ from pose $[\boldsymbol{R}, t]_{W \leftarrow TLS,1}$ to pose $[\boldsymbol{R}, t]_{W \leftarrow TLS,2}$ is

expressed in Eq. (19).

$$[\boldsymbol{R}, t]_{W \leftarrow TLS,12} = [\boldsymbol{R}, t]_{W \leftarrow TLS,1}^{-1} [\boldsymbol{R}, t]_{W \leftarrow TLS,2} = [\boldsymbol{R}, t]_{W \rightarrow TLS,1} [\boldsymbol{R}, t]_{W \leftarrow TLS,2} \tag{19}$$

To construct a relative pose observation equation it is necessary to introduce function $\beta = m2v([\boldsymbol{R}, t]_{W \leftarrow TLS,12})$ that retrieves parametric form of rotation matrix (Tait–Bryan angles, quaternion, Rodrigues etc.). The relative pose observation equation is given in (20).

$$\underbrace{[\delta]^{N \times 1}}_{residuals} = \underbrace{[\beta^{target}]^{N \times 1}}_{target\ values} - \underbrace{m2v_{[\beta]}([\boldsymbol{R}, t]^{12})}_{model\ function} \tag{20}$$

2.4. Optimization problem formulation

Multi-view TLS data registration can defined in general as (21).

$$\beta^* = \min_{\beta} \sum_{k=1}^{K} \sum_{i=1}^{C^K} \left(y_i^k - \boldsymbol{\Psi}_{[\beta]}^k \left(x_i^k \right) \right)^2 \tag{21}$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
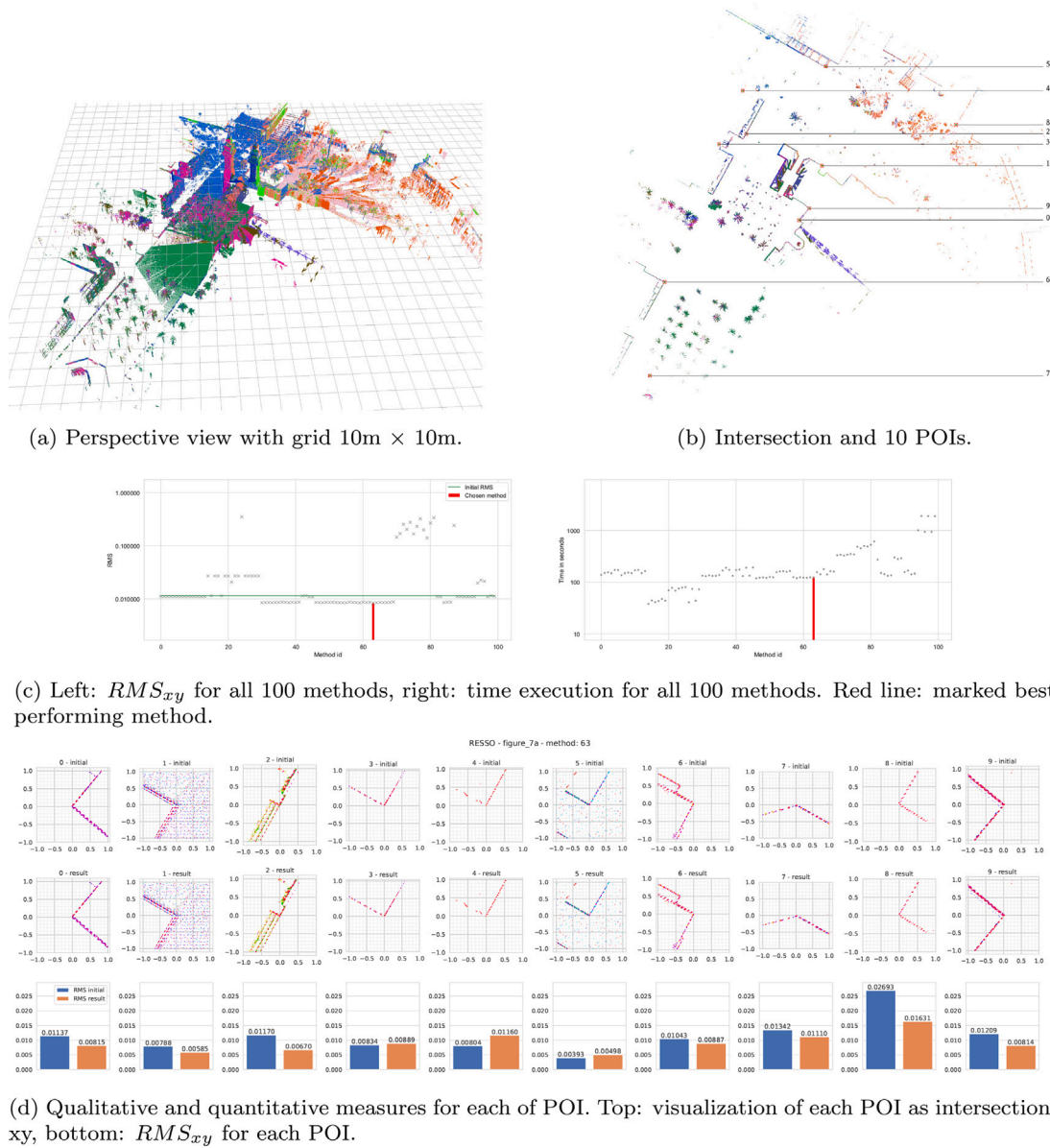
Fig. 9. RESSO data set [10], scene: figure7b. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

It contains K observation equation blocks. Each block contains C^K instances of observation equations that are considered the same type. It is the core concept for building modern SLAM and BA systems. Obviously, it can be extended to weighted non-linear least squares by adding weights for each observation equation. Thus, it will be possible to construct robust optimization and probabilistic approach to the given data registration problem. It will be discussed in the following sections.

2.4.1. Rotation matrix parameterization

Rigid transformation in SO(3) can be expressed in parametric form as Tait–Bryan angles, Rodrigues and quaternions. Other parametric forms exist [35], but they are not considered in this research since Tait–Bryan angles, Rodrigues and quaternions sufficiently cover the desired scope of the investigation. Starting from the basic concept, in three-dimensional space rotations via each axis are given as rotation ω via x axis (22), φ via (23) and κ via z axis (24).

$$\mathbf{R}_x(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\omega) & -\sin(\omega) \\ 0 & \sin(\omega) & \cos(\omega) \end{bmatrix} \tag{22}$$

$$\mathbf{R}_y(\varphi) = \begin{bmatrix} \cos(\varphi) & 0 & \sin(\varphi) \\ 0 & 1 & 0 \\ -\sin(\varphi) & 0 & \cos(\varphi) \end{bmatrix} \tag{23}$$

$$\mathbf{R}_z(\kappa) = \begin{bmatrix} \cos(\kappa) & -\sin(\kappa) & 0 \\ \sin(\kappa) & \cos(\kappa) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{24}$$

There are two different conventions for the definition of the rotation axes — proper Euler angles ($R_z - R_x - R_z$, $R_x - R_y - R_x$, $R_y - R_z - R_y$, $R_z - R_y - R_z$, $R_x - R_z - R_x$, $R_y - R_x - R_y$) and Tait–Bryan angles ($R_x - R_y - R_z$, $R_y - R_z - R_x$, $R_z - R_x - R_y$, $R_x - R_z - R_y$, $R_z - R_y - R_x$, $R_y - R_x - R_z$). For the convenience, in this research the rotation in SE(3) is calculated in arbitrary chosen Tait–Bryan convention ($R_x - R_y - R_z$) - Eq. (25),

$$\begin{aligned} \boldsymbol{R}(\omega, \varphi, \kappa) &= \mathbf{R}_x(\omega)\mathbf{R}_y(\varphi)\mathbf{R}_z(\kappa) \\ &= \begin{bmatrix} c(\varphi)c(\kappa) & -c(\varphi)s(\kappa) & s(\varphi) \\ c(\omega)s(\kappa) + s(\omega)s(\varphi)c(\kappa) & c(\omega)c(\kappa) - s(\omega)s(\varphi)s(\kappa) & -s(\omega)c(\varphi) \\ s(\omega)s(\kappa) - c(\omega)s(\varphi)c(\kappa) & s(\omega)c(\kappa) + c(\omega)s(\varphi)s(\kappa) & c(\omega)c(\varphi) \end{bmatrix} \end{aligned} \tag{25}$$

Table 4

Results for RESSO dataset [10] (metadata same as Table 3).

id	method	GN	LM	W ← TLS	W → TLS	TB	Q	R	leftJ	rightJ	figure_7a RMS_{xy}	figure_7b RMS_{xy}	figure_7c RMS_{xy}	figure_7d RMS_{xy}	figure_7e RMS_{xy}
-	initial[10]										0.011429	0.090880	0.170620	0.133107	0.036096
0	p2p	✓		✓		✓					0.011109	0.079024	0.140534	0.066331	0.018775
1	p2p	✓		✓			✓				0.011109	0.079355	0.140512	0.066397	0.018775
2	p2p	✓		✓				✓			0.011109	0.079012	0.140536	0.066315	0.018775
3	p2p	✓			✓	✓					0.011109	0.078995	0.140532	0.066332	0.018774
4	p2p	✓			✓		✓				0.011109	0.081286	0.140529	0.066360	0.018774
5	p2p	✓			✓			✓			0.011109	0.079013	0.140537	0.066315	0.018774
6	p2p		✓	✓		✓					0.011109	0.079010	0.140550	0.066331	0.018774
7	p2p		✓	✓			✓				0.011109	0.079371	0.140532	0.066396	0.018775
8	p2p		✓	✓				✓			0.011109	0.079030	0.140555	0.066315	0.018775
9	p2p		✓		✓	✓					0.011109	0.079016	0.140549	0.066332	0.018775
10	p2p		✓		✓		✓				0.011109	0.081297	0.140552	0.066359	0.018774
11	p2p		✓		✓			✓			0.011109	0.079034	0.140552	0.066315	0.018775
12	p2p	✓		✓				✓	✓		0.011053	0.092132	0.141210	0.071493	0.021004
13	p2p	✓		✓				✓		✓	0.011053	0.078608	0.148733	0.071114	0.021004
14	ndt	✓		✓		✓					0.027014	0.078699	0.154395	0.025414	0.013981
15	ndt	✓		✓			✓				0.011429	0.079491	0.152893	0.025617	0.036096
16	ndt	✓		✓				✓			0.027057	0.078701	0.154407	0.025408	0.013980
17	ndt	✓			✓	✓					0.027019	0.078700	0.154387	0.025410	0.013982
18	ndt	✓			✓		✓				0.011429	0.081143	0.156877	0.025665	0.036096
19	ndt	✓			✓			✓			0.027060	0.078704	0.154404	0.025410	0.013982
20	ndt		✓	✓		✓					0.027014	0.078699	0.154395	0.025414	0.013981
21	ndt		✓	✓			✓				0.020709	0.079491	0.152893	0.025617	0.291522
22	ndt		✓	✓				✓			0.027057	0.078701	0.154407	0.025408	0.013980
23	ndt		✓		✓	✓					0.027019	0.078700	0.154387	0.025410	0.013982
24	ndt		✓		✓		✓				0.349635	0.081143	0.156877	0.025665	1.092360
25	ndt		✓		✓			✓			0.027060	0.078704	0.154404	0.025410	0.013982
26	ndt	✓		✓				✓	✓		0.027061	0.089817	0.146466	0.025298	0.013980
27	ndt		✓	✓				✓	✓		0.027061	0.089817	0.146466	0.025298	0.013980
28	ndt	✓		✓				✓		✓	0.027057	0.078701	0.154408	0.025410	0.013980
29	ndt		✓	✓				✓		✓	0.027057	0.078701	0.154408	0.025410	0.013980
30	p2proj	✓		✓		✓					0.008490	0.084682	0.024816	0.029461	0.012430
31	p2proj	✓		✓			✓				0.008541	0.082795	0.032877	0.028536	0.012587
32	p2proj	✓		✓				✓			0.008591	0.078343	0.023955	0.029307	0.012538
33	p2proj	✓			✓	✓					0.008517	0.082824	0.021893	0.029610	0.012949
34	p2proj	✓			✓		✓				0.008574	0.081435	0.032784	0.029534	0.012638
35	p2proj	✓			✓			✓			0.008593	0.082198	0.045799	0.029714	0.012552
36	p2proj		✓	✓		✓					0.008713	0.082619	0.022990	0.029472	0.013396
37	p2proj		✓	✓			✓				0.008616	0.084302	0.037296	0.029500	0.012965
38	p2proj		✓	✓				✓			0.008526	0.075386	0.036042	0.030082	0.012719
39	p2proj		✓		✓	✓					0.008636	0.086762	0.022204	0.030906	0.013243
40	p2proj		✓		✓		✓				0.008567	0.084623	0.025707	0.028836	0.012650
41	p2proj		✓		✓			✓			0.008697	0.083844	0.049979	0.029514	0.012988
42	p2proj	✓		✓				✓	✓		0.011334	0.089465	0.127239	0.065282	0.019326
43	p2proj		✓	✓				✓	✓		0.011534	0.088581	0.113642	0.069273	0.020503
44	p2proj	✓		✓				✓		✓	0.011024	0.077941	0.138302	0.067019	0.018381
45	p2proj		✓	✓				✓		✓	0.010944	0.076473	0.131910	0.062212	0.018984
46	p2pl	✓		✓		✓					0.008610	0.077315	0.026148	0.028365	0.012698
47	p2pl	✓		✓			✓				0.008561	0.085208	0.020604	0.029270	0.012372
48	p2pl	✓		✓				✓			0.008526	0.073994	0.035310	0.029248	0.012741
49	p2pl	✓			✓	✓					0.008631	0.082937	0.027449	0.029004	0.013072
50	p2pl	✓			✓		✓				0.008601	0.086256	0.023143	0.029360	0.012786
51	p2pl	✓			✓			✓			0.008629	0.078331	0.033725	0.030311	0.013782
52	p2pl		✓	✓		✓					0.008556	0.083137	0.038652	0.029500	0.013524
53	p2pl		✓	✓			✓				0.008505	0.081469	0.028390	0.029805	0.012805
54	p2pl		✓	✓				✓			0.008485	0.083601	0.031952	0.028889	0.013479
55	p2pl		✓		✓	✓					0.008471	0.077153	0.025139	0.029435	0.012733
56	p2pl		✓		✓			✓			0.008685	0.085326	0.019757	0.028962	0.013822
57	p2pl		✓		✓			✓			0.008605	0.077734	0.048600	0.028810	0.013868
58	dp2pl	✓		✓		✓					0.008566	0.081414	0.020634	0.038029	0.013293
59	dp2pl	✓		✓			✓				0.008437	0.079694	0.028336	0.028433	0.012014
60	dp2pl	✓		✓				✓			0.008705	0.088979	0.023282	0.029705	0.013101
61	dp2pl	✓			✓	✓					0.008579	0.080925	0.022040	0.028854	0.012894
62	dp2pl	✓			✓		✓				0.008661	0.087688	0.024448	0.030398	0.012824
63	dp2pl	✓			✓			✓			0.008395	0.088059	0.041486	0.029087	0.012835
64	dp2pl		✓	✓		✓					0.008478	0.082682	0.025512	0.029252	0.013225
65	dp2pl		✓	✓			✓				0.008577	0.087818	0.030653	0.029495	0.012986
66	dp2pl		✓	✓				✓			0.008541	0.083242	0.025511	0.029093	0.013077
67	dp2pl		✓		✓	✓					0.008539	0.082135	0.021247	0.030504	0.012795
68	dp2pl		✓		✓		✓				0.008601	0.080424	0.022611	0.029841	0.012206
69	dp2pl		✓		✓			✓			0.008770	0.091940	0.036806	0.028990	0.014256
70	pl2pl	✓		✓		✓					0.144612	0.191285	0.177589	0.054713	0.147304
71	pl2pl	✓		✓			✓				0.169572	0.496605	0.196923	0.051981	0.170194
72	pl2pl	✓		✓				✓			0.252915	0.089252	0.195637	0.043675	0.145499
73	pl2pl	✓			✓	✓					0.203364	0.133687	0.517107	0.033052	0.170671
74	pl2pl	✓			✓		✓				0.275695	0.122292	0.223111	0.054307	0.168820
75	pl2pl	✓			✓			✓			0.166145	0.136023	0.554722	0.041359	0.169672
76	pl2pl		✓	✓		✓					0.230890	0.140300	0.276533	0.030141	0.185919
77	pl2pl		✓	✓			✓				0.320611	0.162178	0.296625	0.034018	0.171135
78	pl2pl		✓	✓				✓			0.198834	0.566884	0.172114	0.041508	0.160257
79	pl2pl		✓		✓	✓					0.140805	0.072470	0.219767	0.034635	0.149668
80	pl2pl		✓		✓		✓				0.266258	0.159803	0.429262	0.049411	0.176938
81	pl2pl		✓		✓			✓			0.337681	0.115091	0.200154	0.045993	0.130892
82	pgslam(ndt)	✓		✓		✓					0.011140	0.146640	0.158999	0.042810	0.068246
83	pgslam(p2p)	✓		✓		✓					0.011055	0.090930	0.161610	0.075313	0.014806
84	pgslam(p2proj)	✓		✓		✓					0.008555	0.090930	0.196165	0.027112	0.010442
85	pgslam(p2pl)	✓		✓		✓					0.008699	0.090930	0.077243	0.133107	0.011044
86	pgslam(dp2pl)	✓		✓		✓					0.008711	0.117926	0.077243	0.027112	0.012822
87	pgslam(pl2pl)	✓		✓		✓					0.240892	6.095710	0.193490	0.052295	0.328777
88	pgslam(ndt, left-J)	✓		✓		✓			✓		0.011115	0.090930	0.163082	0.058518	0.042098
89	pgslam(ndt, right-J)	✓		✓		✓				✓	0.011113	0.089485	0.159017	0.133107	0.038955
90	pgslam(p2p, left-J)	✓		✓		✓			✓		0.011022	0.090930	0.189442	0.063202	0.016216
91	pgslam(p2p, right-J)	✓		✓		✓				✓	0.011022	0.090930	0.159586	0.082208	0.016213
92	pgslam(p2proj, left-J)	✓		✓		✓			✓		0.011049	0.079522	0.194578	0.057464	0.015828
93	pgslam(p2proj, right-J)	✓		✓		✓				✓	0.011065	0.100603	0.160289	0.074350	0.015915
94	pgslam(ndt, pcl)	✓		✓				✓			0.019839	0.255118	0.021699	0.026901	0.027467
95	pgslam(icp, pcl)	✓		✓				✓			0.022456	0.090926	0.162882	0.040063	0.016054
96	pgslam(ndt, pcl, GTSAM)	✓		✓				✓		✓	0.021451	0.145534	0.027262	0.039054	0.023876
97	pgslam(icp, pcl, GTSAM)	✓		✓				✓		✓	0.011041	0.102743	0.162815	0.040074	0.010739
98	pgslam(ndt, pcl, manif)	✓		✓				✓		✓	0.011429	0.084757	0.023669	0.047259	0.024350
99	pgslam(icp, pcl, manif)	✓		✓				✓		✓	0.011198	0.101128	0.223186	0.032286	0.010513

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

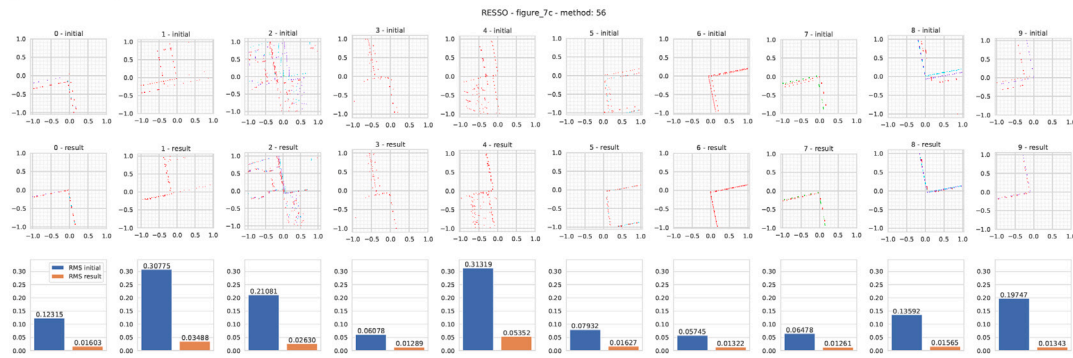(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 10. RESSO data set [10], scene: figure7c. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved.

where s is sin and c is cos functions.

Euler–Rodrigues formula [36,37] is given in Eq. (26). It presents a way for constructing a rotation matrix using a rotation angle θ around the axis $\mathbf{u} = (u_x, u_y, u_z)$[38,39].

$$R = \cos\theta\mathbf{I} + \sin\theta\,[\boldsymbol{u}]_\times + (1 - \cos\theta)\,\boldsymbol{u}\boldsymbol{u}^\mathsf{T} \qquad (26)$$

where:

$$[\boldsymbol{u}]_\times = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \qquad (27)$$

is the cross product (skew-symmetric) matrix. For the Rodrigues parameters (u_x, u_y, u_z, θ) the rotation matrix $R \in$ SO(3) is given in Eq. (28)

$$R = \begin{bmatrix} u_x^2(1-c\theta)+c\theta & u_xu_y(1-c\theta)-u_zs\theta & u_xu_z(1-c\theta)+u_ys\theta \\ u_xu_y(1-c\theta)+u_zs\theta & u_y^2(1-c\theta)+c\theta & -u_xs\theta+u_yu_z(1-c\theta) \\ u_xu_z(1-c\theta)-u_ys\theta & u_xs\theta+u_yu_z(1-c\theta) & u_z^2(1-c\theta)+c\theta \end{bmatrix}$$

$$(28)$$

where $s\theta$ is $\sin(\theta)$ and $c\theta$ is $\cos(\theta)$ function. Furthermore, the rotation vector $\boldsymbol{s} = \theta\boldsymbol{u}$ is compact representation. Eq. (29) shows how to retrieve θ from $\boldsymbol{s}$ or R.

$$\theta = \|\boldsymbol{s}\| = \arccos\left(\frac{\mathrm{Tr}(R)-1}{2}\right) \qquad (29)$$

Consequently, it can be done based on Eq. (30).

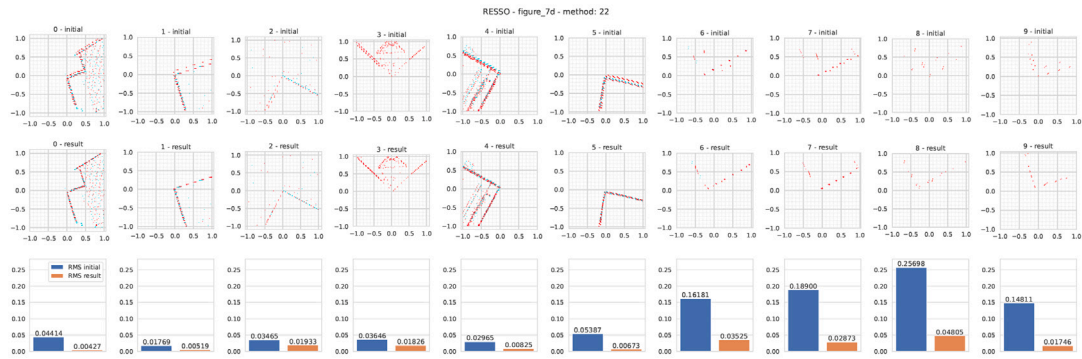$$\mathbf{u} = \frac{\boldsymbol{s}}{\theta} = \frac{\boldsymbol{s}}{\|\boldsymbol{s}\|} \qquad (30)$$

Finally, compact representation of the $R \in$ SO(3) for the Rodrigues parameters (s_x, s_y, s_z) is given in (31) (see Box I).

The third rotation matrix parameterization considered in this paper is based on unit quaternions. The quaternion is given in (32) that satisfies (33).

$$q = q_0 + iq_1 + jq_2 + kq_3 \qquad (32)$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
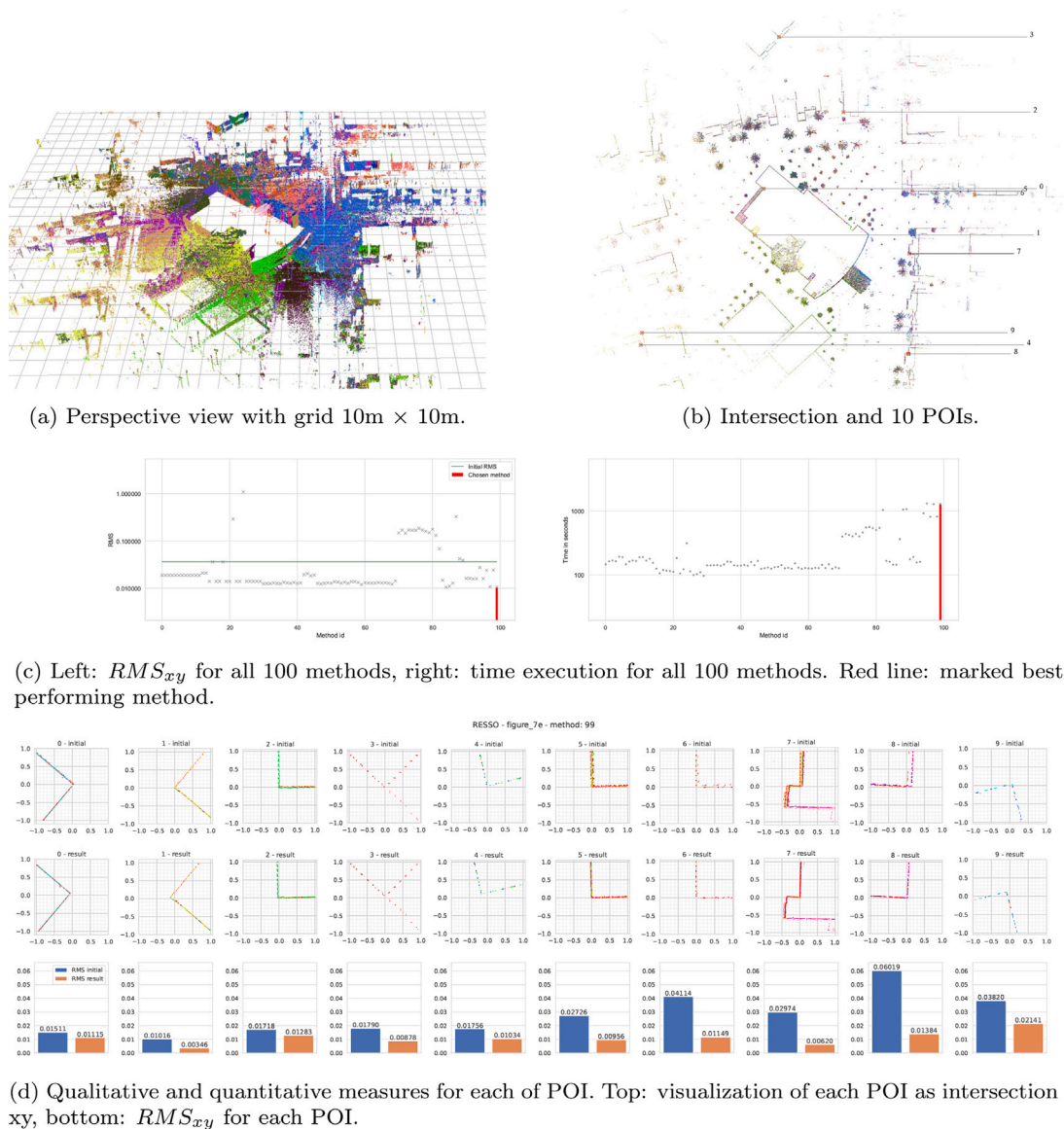
Fig. 11. RESSO data set [10], scene: figure7d. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved.

$$
\begin{bmatrix}
\frac{s_x^2\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} + \cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right) & \frac{s_x s_y\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} - \frac{s_z\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} & \frac{s_x s_z\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} + \frac{s_y\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} \\[3mm]
\frac{s_x s_y\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} + \frac{s_z\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} & \frac{s_y^2\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} + \cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right) & -\frac{s_x\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} + \frac{s_y s_z\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} \\[3mm]
\frac{s_x s_z\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} - \frac{s_y\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} & \frac{s_x\sin\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)}{\sqrt{s_x^2+s_y^2+s_z^2}} + \frac{s_y s_z\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} & \frac{s_z^2\left(1.0-\cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)\right)}{s_x^2+s_y^2+s_z^2} + \cos\left(\sqrt{s_x^2+s_y^2+s_z^2}\right)
\end{bmatrix}
\tag{31}
$$

Box I.

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 12. RESSO data set [10], scene: figure7e. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved.

$$i^2 = j^2 = k^2 = -1$$
$$ij = k, ji = -k$$
$$jk = i, kj = -i \tag{33}$$
$$ki = j, ik = -j$$

A geometrical interpretation of an unit quaternion $q(q_0, q_1, q_2, q_3), \|q\| = 1$ which represents a general rotation is given in Eq. (34).

$$q_0 = \cos(\omega/2)$$
$$q_1 = X \sin(\omega/2)$$
$$q_2 = Y \sin(\omega/2) \tag{34}$$
$$q_3 = Z \sin(\omega/2)$$

Where (X, Y, Z) is the unit length axis of rotation in 3D space and ω is the angle of the unit rotation about the axis in radians. For the unit quaternion $q(q_0, q_1, q_2, q_3), \|q\| = 1$ the rotation matrix $R \in$ SO(3) is

given:

$$R = \begin{bmatrix} 1 - 2q_2^2 - 2q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & 1 - 2q_1^2 - 2q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & 1 - 2q_1^2 - 2q_2^2 \end{bmatrix} \tag{35}$$

An example of the application of quaternions is the SLERP method [40] that interpolates a point movement along an arc given by Eq. (36). For $t = 0$, the interpolation is the starting point of the arc as q_0. For $t = 1$ it is the ending point of the arc as q_1.

$$\text{SLERP}(q_0, q_1, t) = (q_1 q_0^{-1})^t q_0 \tag{36}$$

2.4.2. Weighted non-linear least squares optimization

We start by rearranging Eq. (21) to form (37) by concatenating all residual blocks into sum square errors for the clarity of further notation.

$$SSR = \sum_{i=1}^{C} r_i^2 = \underbrace{\sum_{i=1}^{C} \left(y_i - \Psi_{[\beta]}(x_i) \right)^2}_{objective\ function} \tag{37}$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
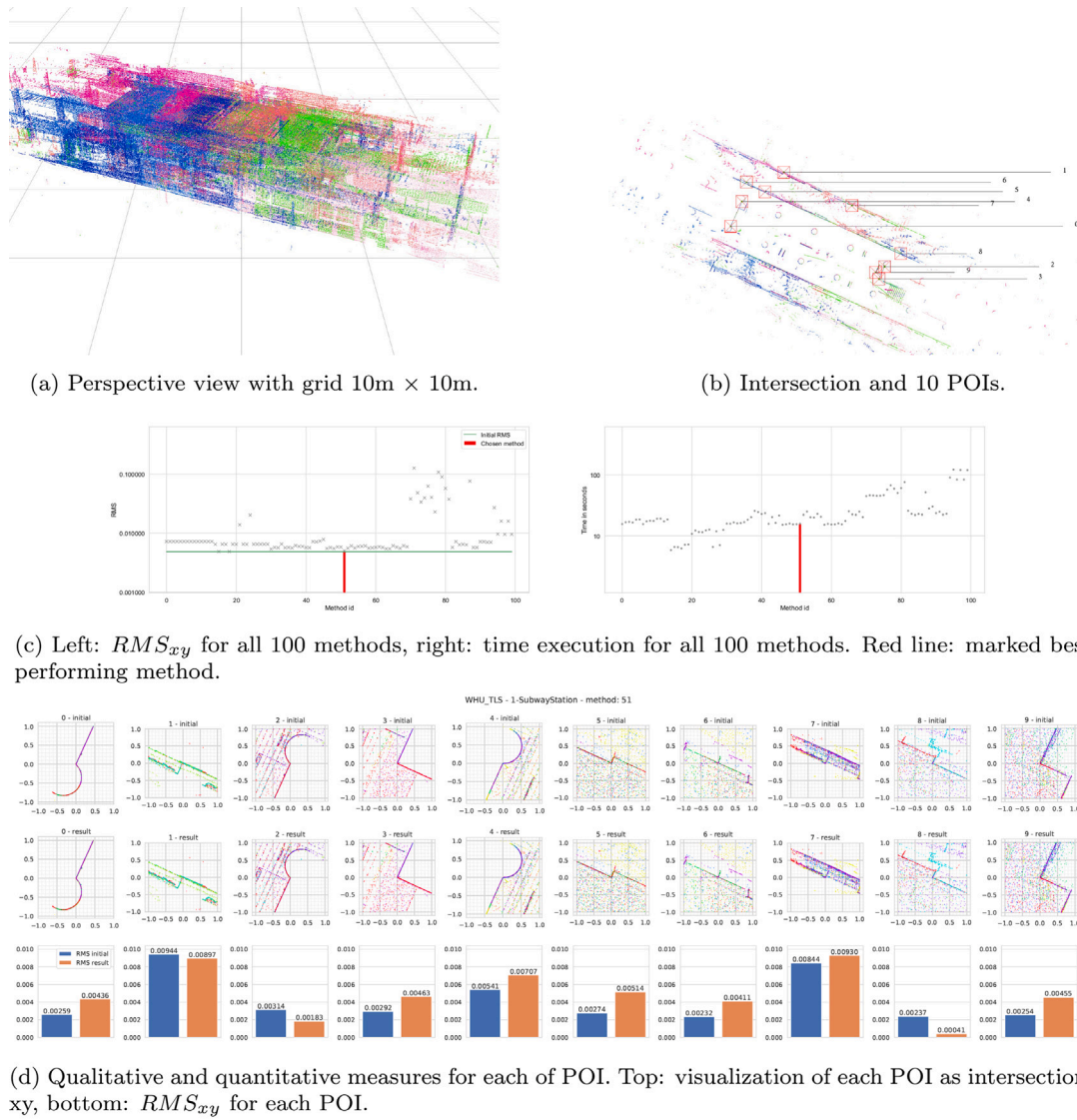
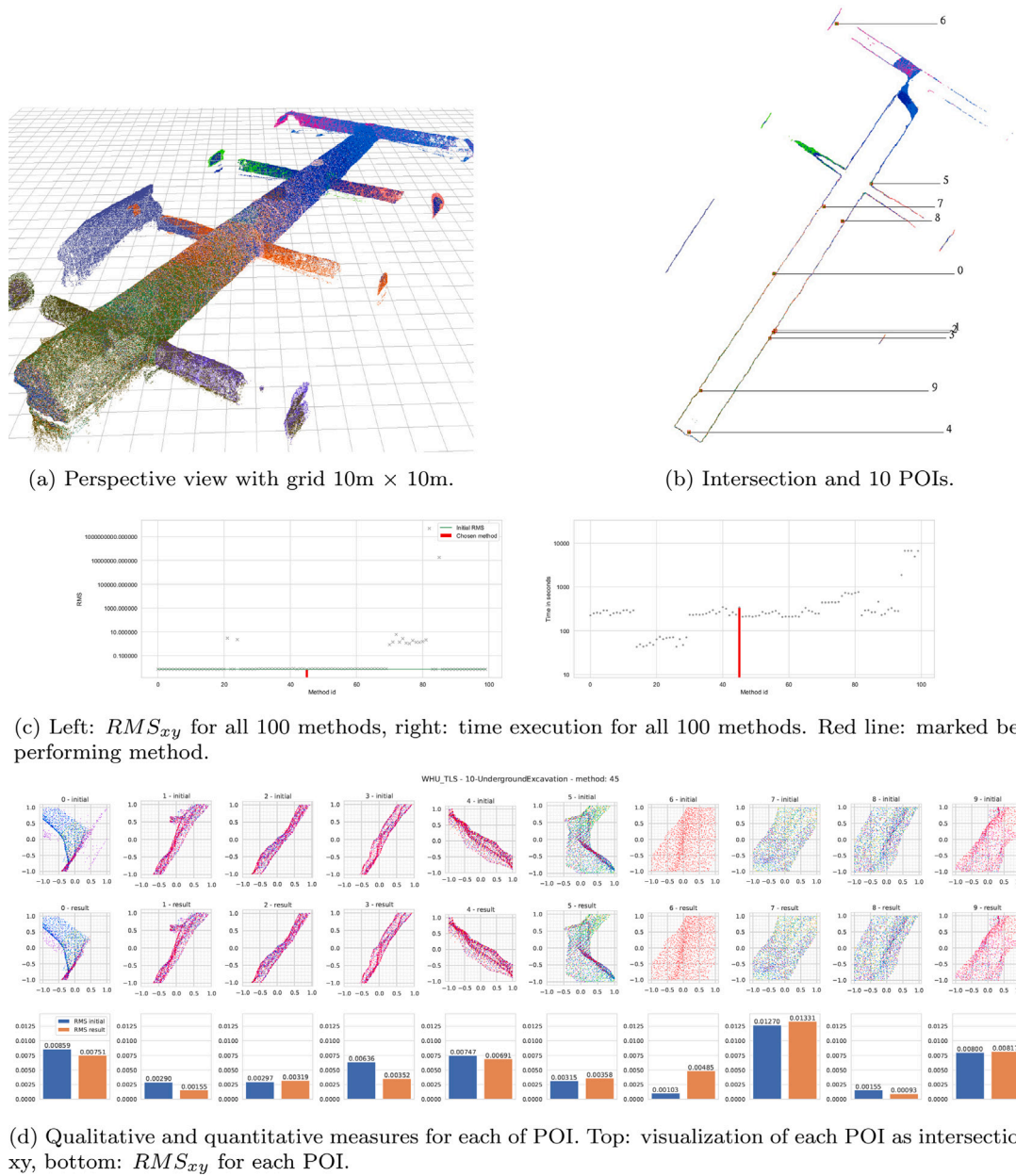**Fig. 13.** WHU_TLS data set [12], scene: 1-SubwayStation. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is not improved.

The model function $\boldsymbol{\Psi}_{[\beta]}(\boldsymbol{x}_i)$ contains n parameters of $\boldsymbol{\beta}$, thus there are n gradient Eqs. (38) derived from (37).

$$\frac{\partial SSR}{\partial \beta_j} = 2\sum_{i=1}^{C} \frac{\partial r_i}{\partial \beta_j} r_i = -2\sum_{i=1}^{C} \frac{\partial \boldsymbol{\Psi}_{[\beta]}(\boldsymbol{x}_i)}{\partial \beta_j} r_i = 0 \quad (j = 1, \dots, n) \tag{38}$$

Such a system can be solved in some cases by so-called closed form solution [41,42], thus the solution is obtained in a single step, otherwise it has to be solved using an iterative approach by solving Eq. (39),

$$\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\Delta\beta = \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\Delta\boldsymbol{y} \tag{39}$$

where $\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}$ is the Jacobian of the model function $\boldsymbol{\Psi}$. Extending Eq. (37) by weights w provides a mechanism to control the impact of each observation equation into the optimization process and it is formulated as (40).

$$SSR = \sum_{i=1}^{C} w_i r_i^2 \tag{40}$$

The optimal solution that minimizing SSR can be found with solving formula (41),

$$\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{W}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\Delta\beta = \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{W}\Delta\boldsymbol{y} \tag{41}$$

where $\boldsymbol{W}$ is a diagonal weight matrix, thus it is easy to provide the independent impact for each observation equation. Once we introduce rotation matrix $\boldsymbol{R}$ that provides a correlation mechanism to $\boldsymbol{W}$ the final Weighted non-linear Least Squares Optimization is formulated as (42).

$$\begin{aligned}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{R}\boldsymbol{W}\boldsymbol{R}^{\mathsf{T}}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\Delta\beta &= \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{R}\boldsymbol{W}\boldsymbol{R}^{\mathsf{T}}\Delta\boldsymbol{y} \overset{\boldsymbol{R}\boldsymbol{W}\boldsymbol{R}^{\mathsf{T}}=\boldsymbol{\Omega}}{\Rightarrow}\\ \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{\Omega}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\Delta\beta &= \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{\Omega}\Delta\boldsymbol{y} \overset{\boldsymbol{\Omega}=\boldsymbol{\Sigma}^{-1}}{\Rightarrow}\\ \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\Delta\beta &= \overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\Delta\boldsymbol{y}\end{aligned} \tag{42}$$

where $\boldsymbol{\Omega} = \boldsymbol{R}\boldsymbol{W}\boldsymbol{R}^{\mathsf{T}}$ is the information matrix and $\boldsymbol{\Sigma} = \boldsymbol{\Omega}^{-1}$ is the covariance matrix.

2.4.3. Solvers

In this research, we consider Gauss–Newton and Levenberg–Marquardt optimization methods. For observation Eq. (5) the Gauss–Newton method is expressed as (43),

$$\beta^{k+1} = \beta^k + \left(\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}\right)^{-1}\overset{\boldsymbol{\Psi}}{\boldsymbol{J}}{}^{\mathsf{T}}\boldsymbol{r}(\beta^k) \tag{43}$$

16

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 14. WHU_TLS data set [12], scene: 10-UndergroundExcavation. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that it is difficult to justify if it is improved.

where β^{k+1} is an updated vector state of the optimization step. Levenberg–Marquardt optimization method introduces a damping factor λ to Eq. (43) resulting (44).

$$\beta^{k+1} = \beta^k + \left(\overset{\Psi}{J}{}^{\top} \overset{\Psi}{J} + \lambda I \right)^{-1} \overset{\Psi}{J}{}^{\top} r(\beta^k) \tag{44}$$

It can be deduced that a large value of damping factor λ results fix optimization parameter observation equation discussed in 2.3.2. Generally, the smaller value of λ results in optimization behavior more similar to the Gauss–Newton method. Larger values of λ prevent large optimization steps, thus the accuracy of the final data registration can be significantly better.

2.4.4. Robust least squares

Robust Least Square are capable reducing negative impact of the outliers using so-called *M-estimator* [43]. It can be done by adding small weight to such observation equation as in Eq. (40). Unfortunately,

the choice of the best *M-estimator* for a given problem is not straightforward. Prominent candidates are Huber, Cauchy, Geman-McClure or Welsch functions [44]. An alternative approach called Barron's adaptive robust loss function was introduced in [7]. An investigation in relation to the LiDAR data registration and BA is shown in [45]. Barron's adaptive robust loss function is given in Eq. (45)

$$\rho(r, \alpha, c) = \frac{|\alpha - 2|}{\alpha} \left(\left(\frac{(\frac{r}{c})^2}{|\alpha - 2|} + 1 \right)^{\frac{\alpha}{2}} - 1 \right) \tag{45}$$

where $\alpha \in \mathbb{R}$ is a shape parameter that controls the robustness of the loss and $c > 0$ is a scale parameter that controls the size of the loss quadratic bowl near $r = 0$. Setting $\alpha = -2$ the $\rho(r, \alpha, c)$ becomes the *Geman − McClure* loss function (46).

$$\rho(r, -2, c) = \frac{2 \left(\frac{r}{c} \right)^2}{\left(\frac{r}{c} \right)^2 + 4} \tag{46}$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
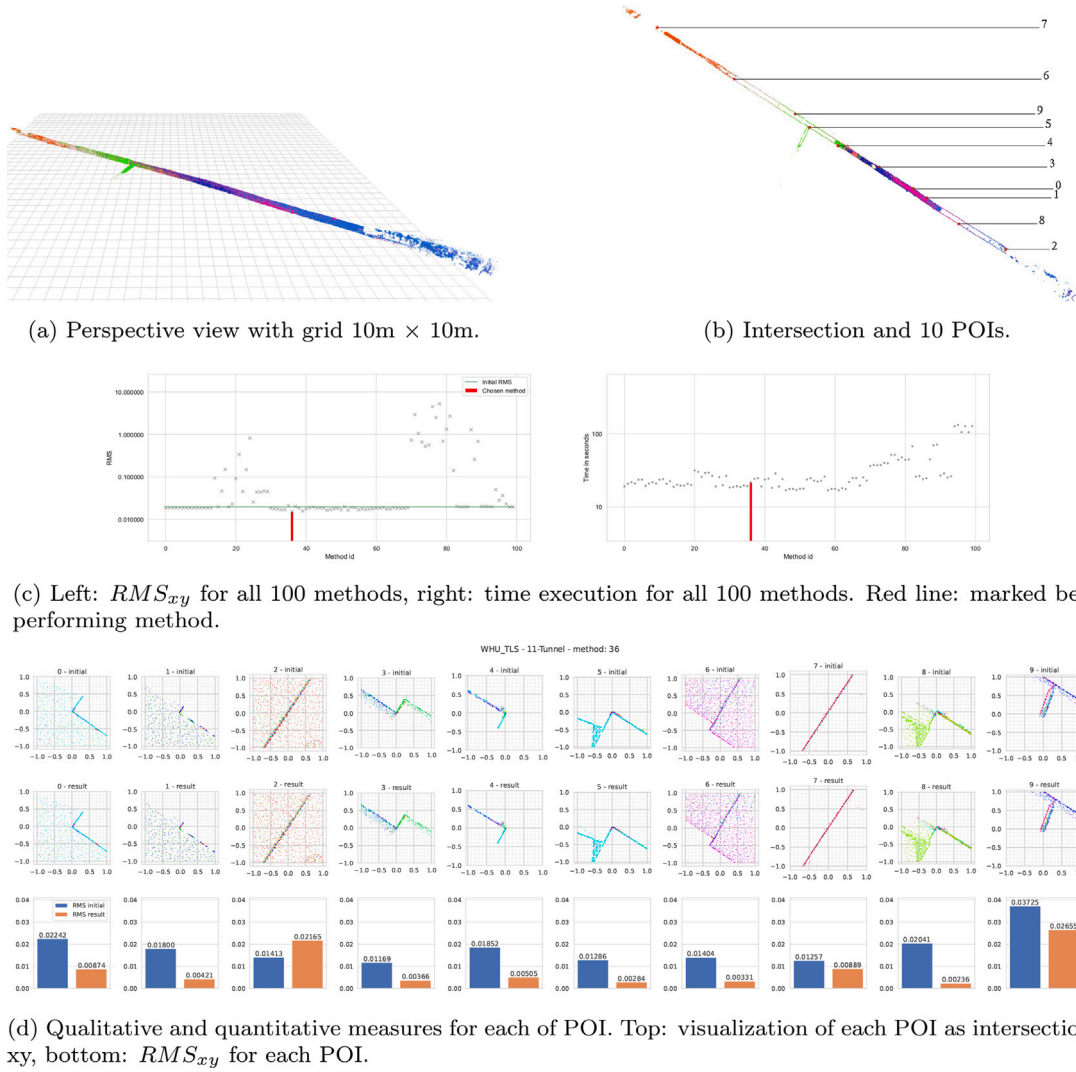
Fig. 15. WHU_TLS data set [12], scene: 11-Tunnel. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is improved.

For $\alpha = 0$ the $\rho(r, \alpha, c)$ is undefined and it approaches $Cauchy$ loss in the limit as shown in Eq. (47).

$$\lim_{\alpha \to 0} \rho(r, \alpha, c) = \log\left(\frac{1}{2}\left(\frac{r}{c}\right)^2 + 1\right) \tag{47}$$

Similarly for $\alpha = 2$ the $\rho(r, \alpha, c)$ is undefined and it approaches $L2$ loss in the limit as shown in Eq. (48).

$$\lim_{\alpha \to 2} \rho(r, \alpha, c) = \frac{1}{2}\left(\frac{r}{c}\right)^2 \tag{48}$$

For $\alpha = 1$ the loss function $\rho(r, \alpha, c)$ becomes the $L1 - L2$ loss function (49).

$$\rho(r, 1, c) = \sqrt{1 + \left(\frac{r}{c}\right)^2} - 1 \tag{49}$$

For α approaching the negative infinity the $\rho(r, \alpha, c)$ becomes the $Welsch$ loss function (50).

$$\lim_{\alpha \to -\infty} \rho(r, \alpha, c) = 1 - e^{-\frac{1}{2}\left(\frac{r}{c}\right)^2} \tag{50}$$

Finally, the $\rho(r, \alpha, c)$ Barron's loss function with its removable singularities at $\alpha = 0$, $\alpha = 2$ and its limit at $\alpha = -\infty$, the $Y(r)$ - influence function and the corresponding $w(r)$ - weight is given in table 1.

Barron's loss function in practical application adapts to data set by optimizing α for arbitrary chosen c [45]. In such a scenario this robust approach can efficiently reduce the impact of outliers without the need for the data set manual inspection.

2.4.5. Probabilistic approach

Poses of TLS are described by the random variables $p_{1:P} = \{p_1, \ldots, p_P\}$. During TLS displacements a sequence of the relative poses measurements $u_{1:T} = \{u_1, \ldots, u_T\}$ and the perception measurements $o_{1:O} = \{o_1, \ldots, o_O\}$ are acquired. The perception measurements can be done e.g. with an ICP algorithm that calculates the relative pose between two overlapping scans. The optimization problem is formulated as the estimation of the posterior probability of all nodes of the set of poses $p_{1:P}$ and the map m of the environment for given all the measurements and the arbitrarily chosen initial pose p_0 as in Eq. (51).

$$p(\underbrace{p_{1:P}, m}_{x=\{x_1, \ldots, x_N\}} \mid \underbrace{o_{1:O}, u_{1:T}, p_0}_{z=\{z_1, \ldots, z_K\}}) \tag{51}$$

The map m is represented as a dense point cloud. A probabilistic problem definition assumes that $p_{1:P}, m$ are state variables of the stationary system denoted as $x = \{x_1, \ldots, x_N\}$. The indirect observation of the system is done by a set of e.g. ICP measurements denoted as $z = \{z_1, \ldots, z_K\}$. The measurements are affected by noise, therefore z is the multi-dimensional vector of random variables. Normal distribution $\mathcal{N}(x; \mu, \Sigma)$ is characterized by density function of the form (52).

$$p(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x - \mu)^\top \Sigma^{-1}(x - \mu)\right\} \tag{52}$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
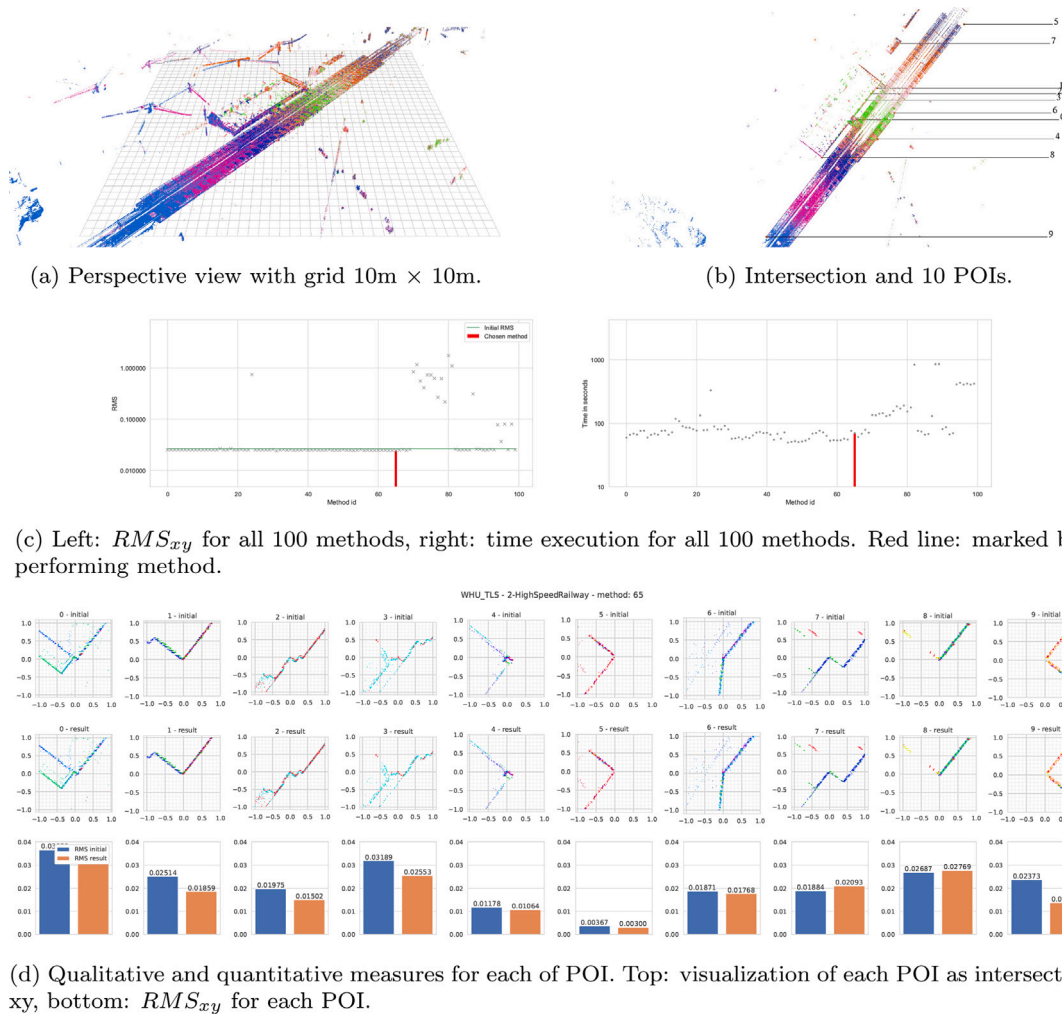
Fig. 16. WHU_TLS data set [12], scene: 2-HighSpeedRailway. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is slightly improved.

where, $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\Sigma}$ is a covariance matrix and $(\boldsymbol{x} - \boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})$ denotes the squared Mahalanobis distance. It is impossible to estimate the state of the system for the given measurements affected by the noise. To do so, we compute a distribution over the potential states of the system given these measurements. Thus, the probability distribution of the state $\boldsymbol{x}$, given the measurement $\boldsymbol{z}$ is estimated with (53).

$$p(\boldsymbol{x}|\boldsymbol{z}) = p\left(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N | \boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\right) = p\left(\boldsymbol{x}_{1:N} | \boldsymbol{z}_{1:K}\right) \tag{53}$$

Furthermore, from Bayes' theorem we obtain (54).

$$p\left(\boldsymbol{x}_{1:N} | \boldsymbol{z}_{1:K}\right) = \frac{\overbrace{p\left(\boldsymbol{z}_{1:K} | \boldsymbol{x}_{1:N}\right)}^{\substack{\text{likelihood of} \\ \text{measurements} \\ \text{given the states}}} p\left(\boldsymbol{x}_{1:N}\right)}{p\left(\boldsymbol{z}_{1:K}\right)} \tag{54}$$

Where $p\left(\boldsymbol{z}_{1:K} | \boldsymbol{x}_{1:N}\right)$ is *likelihood* of the measurements given the states. The prior $p\left(\boldsymbol{x}_{1:N}\right)$ models the states before the measurements. The prior is a uniform distribution whose value is a constant p_x. The $p\left(\boldsymbol{z}_{1:K}\right)$ does not dependent on the states and it is a constant number p_z. Therefore, Eq. (54) become:

$$p\left(\boldsymbol{x}_{1:N} | \boldsymbol{z}_{1:K}\right) = \frac{p_x}{p_z} p\left(\boldsymbol{z}_{1:K} | \boldsymbol{x}_{1:N}\right) \propto \prod_{k=1}^{K} p\left(\boldsymbol{z}_k | \boldsymbol{x}_{1:N}\right) \tag{55}$$

Thus, the distribution over the possible states given the measurements is proportional to the likelihood of the measurement given the states. Assuming that the measurements are affected by a zero mean additive Gaussian noise characterized by an information matrix $\boldsymbol{\Omega}_k = \boldsymbol{\Sigma}_k^{-1}$ the distribution of the measurement $\boldsymbol{z}_k$ given a state $\boldsymbol{x}$ is proportional to $\boldsymbol{N}(\boldsymbol{z}_k; \hat{\boldsymbol{z}}_k, \boldsymbol{\Omega}_k^{-1})$, thus

$$p\left(\boldsymbol{z}_k | \boldsymbol{x}\right) \propto \exp\left(-\left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)^{\mathsf{T}} \boldsymbol{\Omega}_k \left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)\right) \tag{56}$$

where $\hat{\boldsymbol{z}}_k = \boldsymbol{h}_k(\boldsymbol{x})$ is the prediction of the measurement for given state $\boldsymbol{x}$. $p\left(\boldsymbol{z}_k | \boldsymbol{x}\right)$ is commonly known as *sensor model* or *observation model*. It models the probability of performing the observation $\boldsymbol{z}_k$ given that the TLS is at a known location in the map (the state $\boldsymbol{x}$ is known). From (55) and (56) the probabilistic approach can be formulated as finding $\boldsymbol{x}^*$ that:

$$\boldsymbol{x}^* = \max_{\boldsymbol{x}} \prod_{k=1}^{K} p\left(\boldsymbol{z}_k | \boldsymbol{x}_{1:N}\right) = \max_{\boldsymbol{x}} \prod_{k=1}^{K} \exp\left(-\left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)^{\mathsf{T}} \boldsymbol{\Omega}_k \left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)\right) \tag{57}$$

After taking the logarithm and removing the minus Eq. (57) becomes:

$$\boldsymbol{x}^* = \max_{\boldsymbol{x}} \sum_{k=1}^{K} \left(-\left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)^{\mathsf{T}} \boldsymbol{\Omega}_k \left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)\right) = \min_{\boldsymbol{x}} \sum_{k=1}^{K} \left(\left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)^{\mathsf{T}} \boldsymbol{\Omega}_k \left(\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k\right)\right) \tag{58}$$

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
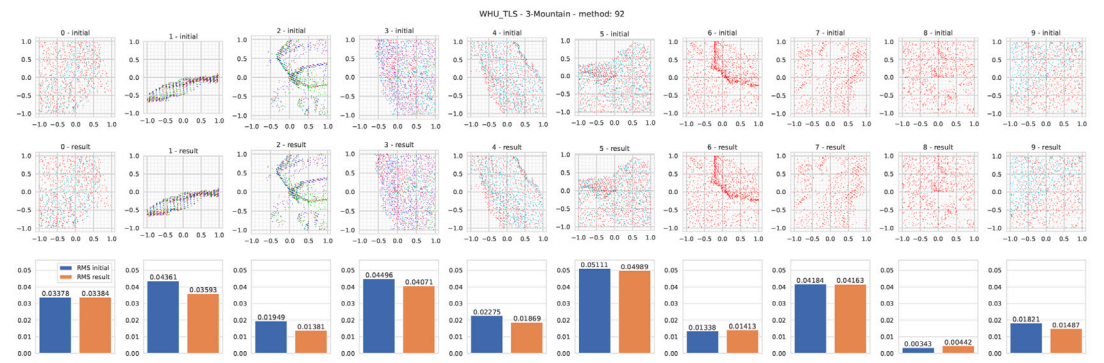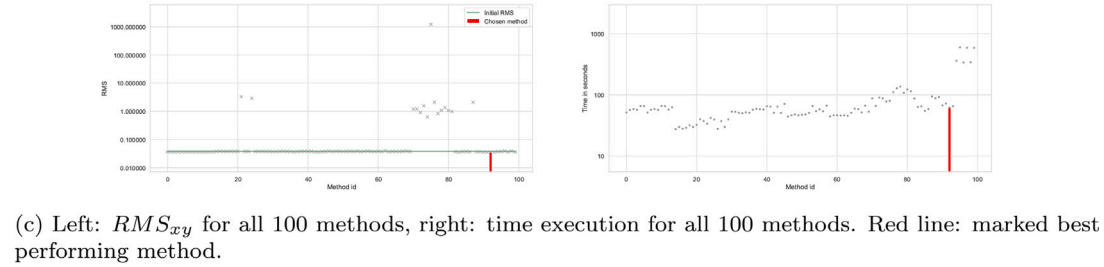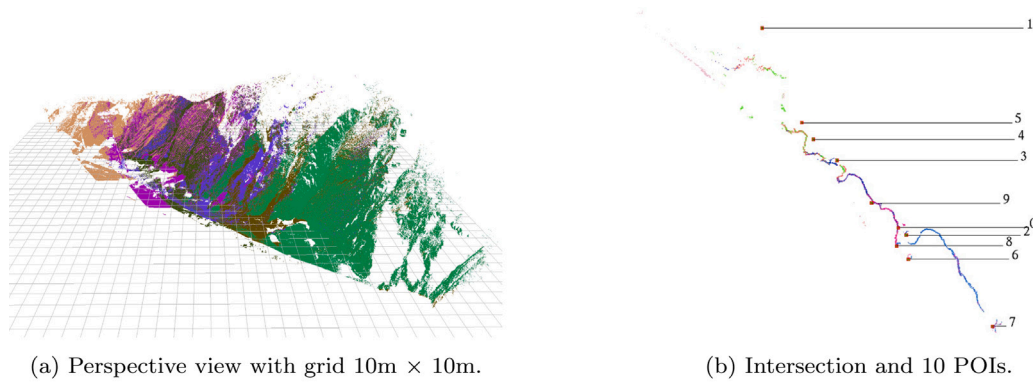
Fig. 17. WHU_TLS data set [12], scene: 3-Mountain. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is slightly improved.

After introducing the error function $e_k(x)$ as a difference between the observation z_k and the prediction $\hat{z}_k$.

$$e_k(x) = z_k - \hat{z}_k \tag{59}$$

final optimization problem is defined as (60).

$$x^* = \min_x F(x) = \min_x \sum_{k=1}^{K} e_k(x)^{\mathsf{T}} \Omega_k e_k(x) \tag{60}$$

It can be solved as typical Weighted non-linear Least Squares Optimization as in Eq. (42) with an assumption of known information matrix Ω modeling the difference between the observation z_k and the prediction $\hat{z}_k$ in a probabilistic fashion.

2.4.6. Point to point left and right Jacobians in Lie algebra

Jacobian from Eq. (42) for point-to-point observation equation discussed in Section 2.2.1 can be drastically reduced using Lie algebra theory [46] to the form of left (62) and right Jacobians (62),

$$\begin{bmatrix} I & -[Rp]_\times \end{bmatrix}^{3\times 6} \tag{61}$$

$$\begin{bmatrix} R & -R[p]_\times \end{bmatrix}^{3\times 6} \tag{62}$$

where p denotes coordinates of the source point expressed in local coordinates and $[\cdot]_\times$ denotes the skew-symmetric matrix. The choice between left and right Jacobians has to be done arbitrarily, thus if we optimize locally (as most in the mobile robotic community) we use right Jacobian otherwise we use left Jacobian. It is straightforward to observe that left Jacobian is identical to analytically retrieved Jacobian for the translation part t of the transformation matrix $[R, t]_{W \leftarrow TLS}$. This approach drastically reduces the number of required calculations. Unfortunately, it introduces lower accuracy for point cloud registration which will be shown in the experimental part. This issue was not elaborated on in literature so far.

2.5. Normal distributions transform

Normal Distributions Transform [47] is an alternative technique to ICP for point cloud data registration and it is available in a well-known Point Cloud Library [48] open source project. It is limited to the pairwise matching of two point clouds, thus a contribution of the proposed research is a novel approach to NDT enabling large-scale datasets registration using a multi-view method implemented both by (i) data registration system optimizing all poses simultaneously and (ii)

Table 5

Results for WHU dataset [12] (metadata same as Table 3).

id	method	GN	LM	W←TLS	W→TLS	TB	Q	R	leftJ	rightJ	1-SubwayStation	10-UndergroundExcavation	11-Tunnel	2-HighSpeedRailway	3-Mountain	5-Park	6-Campus	7-Residence	8-RiverBank	9-HeritageBuilding
–	initial[12]										0.004872	0.006922	0.019648	0.026564	0.037906	0.060645	0.065898	0.066544	0.010751	0.011927
0	p2p	✓		✓		✓					0.007254	0.007019	0.018713	0.025222	0.036563	0.040017	0.052178	0.049762	0.009904	0.007235
1	p2p	✓		✓			✓				0.007254	0.007022	0.018712	0.025222	0.036563	0.040018	0.052178	0.049762	0.009904	0.007235
2	p2p	✓		✓				✓			0.007254	0.007023	0.018712	0.025222	0.036563	0.040017	0.052178	0.049762	0.009904	0.007235
3	p2p	✓				✓	✓				0.007254	0.007023	0.018711	0.025222	0.036563	0.040017	0.052178	0.049761	0.009904	0.007235
4	p2p	✓		✓		✓	✓				0.007254	0.007023	0.018711	0.025222	0.036563	0.040017	0.052178	0.049761	0.009904	0.007235
5	p2p	✓		✓				✓			0.007254	0.007023	0.018711	0.025222	0.036563	0.040017	0.052178	0.049761	0.009904	0.007235
6	p2p	✓		✓	✓			✓			0.007253	0.007022	0.018713	0.025222	0.036563	0.040019	0.052178	0.049853	0.009904	0.007235
7	p2p		✓	✓			✓				0.007253	0.007023	0.018712	0.025222	0.036563	0.040019	0.052178	0.049854	0.009904	0.007235
8	p2p		✓	✓				✓			0.007253	0.007023	0.018712	0.025222	0.036563	0.040019	0.052178	0.049854	0.009904	0.007235
9	p2p		✓	✓	✓			✓			0.007253	0.007023	0.018711	0.025222	0.036563	0.040019	0.052178	0.049981	0.009904	0.007235
10	p2p		✓			✓	✓				0.007253	0.007023	0.018712	0.025222	0.036563	0.040019	0.052178	0.049851	0.009904	0.007235
11	p2p		✓	✓	✓						0.007253	0.007023	0.018712	0.025222	0.036563	0.040018	0.052178	0.049851	0.009904	0.007235
12	p2p	✓		✓				✓	✓		0.007242	0.006945	0.018709	0.025261	0.036708	0.041133	0.053493	0.053077	0.010012	0.007542
13	p2p	✓		✓				✓	✓	✓	0.007242	0.006944	0.018710	0.025261	0.036708	0.041134	0.053493	0.053073	0.010012	0.007542
14	ndt	✓		✓		✓					0.006489	0.007149	0.093025	0.025179	0.037645	0.019075	0.054346	0.062965	0.009448	0.007982
15	ndt	✓		✓			✓				0.004873	0.009648	0.026565	0.037906	0.060646	0.065898	0.066544	0.010751	0.011927
16	ndt	✓		✓				✓			0.006515	0.007152	0.046199	0.025183	0.037647	0.019086	0.054347	0.062963	0.009451	0.007978
17	ndt	✓				✓	✓				0.006491	0.007163	0.149475	0.025178	0.037645	0.019074	0.054346	0.062966	0.009448	0.007983
18	ndt	✓		✓		✓	✓				0.004873	0.006923	0.019648	0.026565	0.037906	0.060646	0.065898	0.066544	0.010751	0.011927
19	ndt	✓		✓				✓			0.006515	0.007167	0.022869	0.025183	0.037647	0.019083	0.054347	0.062966	0.009451	0.007978
20	ndt	✓		✓	✓			✓			0.006489	0.007149	0.091831	0.025179	0.037645	0.019075	0.054346	0.062967	0.009448	0.007982
21	ndt		✓	✓			✓				0.013866	2.936300	0.336437	0.029693	3.289770	3.270860	0.063106	0.064474	0.137244	2.106000
22	ndt		✓	✓				✓			0.006515	0.007152	0.045254	0.025183	0.037647	0.019085	0.054347	0.062967	0.009451	0.007978
23	ndt		✓	✓	✓			✓			0.006491	0.007163	0.147329	0.025178	0.037645	0.019074	0.054346	0.062966	0.009448	0.007983
24	ndt		✓			✓	✓				0.020200	2.239080	1.819134	0.742909	2.892310	3.862080	1.428450	0.064469	0.301800	2.106170
25	ndt		✓	✓	✓						0.006515	0.007167	0.025458	0.025183	0.037647	0.019083	0.054347	0.062966	0.009451	0.007978
26	ndt		✓	✓				✓	✓		0.006516	0.007188	0.043858	0.025183	0.037647	0.019083	0.054347	0.062965	0.009451	0.007978
27	ndt		✓	✓				✓	✓		0.006516	0.007152	0.043743	0.025183	0.037647	0.019082	0.054347	0.062966	0.009451	0.007978
28	ndt		✓	✓				✓	✓	✓	0.006516	0.007152	0.045998	0.025183	0.037647	0.019086	0.054347	0.062967	0.009451	0.007978
29	ndt		✓	✓				✓			0.006515	0.007152	0.045265	0.025183	0.037647	0.019086	0.054347	0.062967	0.009451	0.007978
30	p2proj	✓		✓		✓					0.005364	0.007393	0.018298	0.025686	0.037154	0.022436	0.034215	0.049878	0.009386	0.006825
31	p2proj	✓		✓			✓				0.005732	0.007786	0.017600	0.023900	0.037242	0.021461	0.031372	0.010143	0.009370	0.006254
32	p2proj	✓		✓				✓			0.005664	0.007525	0.017338	0.025096	0.037545	0.020891	0.028685	0.011139	0.009542	0.006172
33	p2proj	✓				✓	✓				0.006557	0.007671	0.016778	0.025542	0.037813	0.029923	0.023388	0.009206	0.009578	0.006033
34	p2proj	✓		✓		✓	✓				0.005670	0.007735	0.016806	0.024883	0.037691	0.021809	0.033493	0.010452	0.009687	0.005816
35	p2proj	✓		✓				✓			0.005732	0.007609	0.020891	0.025366	0.037428	0.024327	0.032420	0.010400	0.009683	0.006196
36	p2proj		✓	✓			✓				0.005620	0.007721	0.015324	0.024790	0.036725	0.022203	0.030350	0.009856	0.009400	0.005971
37	p2proj		✓	✓				✓			0.006168	0.007640	0.020163	0.025053	0.037225	0.020665	0.032774	0.010834	0.009319	0.006469
38	p2proj		✓	✓	✓			✓			0.005971	0.007531	0.018464	0.024944	0.037681	0.022166	0.033450	0.009624	0.009662	0.005888
39	p2proj		✓			✓	✓				0.005965	0.007642	0.015726	0.025643	0.038000	0.024410	0.032059	0.009519	0.009585	0.006176
40	p2proj		✓	✓	✓						0.005705	0.007679	0.017498	0.024799	0.037748	0.024841	0.029538	0.010887	0.009578	0.006497
41	p2proj		✓	✓				✓	✓		0.006081	0.008085	0.018620	0.025111	0.037738	0.020250	0.029986	0.009825	0.009579	0.006271
42	p2proj	✓		✓				✓	✓		0.007218	0.007091	0.019452	0.025289	0.036619	0.040745	0.052588	0.050370	0.009915	0.007164
43	p2proj	✓		✓				✓	✓		0.007145	0.007823	0.018521	0.025038	0.036240	0.040285	0.052471	0.052020	0.009766	0.007302
44	p2proj		✓	✓				✓		✓	0.007368	0.007788	0.018674	0.024878	0.036979	0.041614	0.052254	0.051453	0.009997	0.007050
45	p2proj		✓	✓				✓		✓	0.007842	0.007009	0.018407	0.024981	0.037447	0.041263	0.052491	0.036192	0.009916	0.007210
46	p2pl	✓		✓				✓			0.005495	0.007643	0.016368	0.025499	0.038179	0.020762	0.031929	0.009185	0.009554	0.006284
47	p2pl	✓		✓		✓					0.005705	0.007814	0.016858	0.024880	0.037586	0.022377	0.029249	0.009491	0.010068	0.006244
48	p2pl	✓		✓			✓				0.005494	0.007853	0.017729	0.024827	0.037995	0.023467	0.029367	0.010117	0.009584	0.005784
49	p2pl	✓		✓				✓			0.005854	0.007474	0.018209	0.024856	0.037830	0.021966	0.032552	0.010149	0.009542	0.005842
50	p2pl	✓				✓	✓				0.005816	0.007813	0.016793	0.025448	0.037412	0.020251	0.033120	0.009930	0.010036	0.005677
51	p2pl	✓		✓		✓	✓				0.007557	0.007557	0.019850	0.024957	0.037508	0.021031	0.035257	0.010363	0.010447	0.006082
52	p2pl	✓		✓				✓			0.005575	0.007761	0.016517	0.024896	0.037861	0.020773	0.031623	0.010447	0.009651	0.005763
53	p2pl		✓	✓			✓				0.006092	0.007770	0.019224	0.024504	0.037554	0.019887	0.032762	0.010876	0.009688	0.006284
54	p2pl		✓	✓				✓			0.005613	0.007661	0.018794	0.024870	0.038273	0.023268	0.031161	0.010041	0.009401	0.006035
55	p2pl		✓	✓	✓			✓			0.005949	0.007518	0.016693	0.025583	0.037151	0.022035	0.032013	0.009840	0.009541	0.006394
56	p2pl		✓			✓	✓				0.005991	0.007413	0.017601	0.024525	0.037926	0.020119	0.030501	0.010235	0.009770	0.005961
57	p2pl		✓	✓	✓						0.006085	0.007566	0.016989	0.024930	0.038187	0.021075	0.029169	0.009618	0.009488	0.005776
58	dp2pl	✓		✓				✓			0.005768	0.007420	0.024619	0.037542	0.021842	0.034713	0.009321	0.009477	0.006425	
59	dp2pl	✓		✓				✓			0.005647	0.007536	0.017773	0.024569	0.037429	0.020720	0.031515	0.009255	0.009599	0.006221
60	dp2pl	✓		✓				✓			0.005736	0.007816	0.018336	0.024701	0.037934	0.022583	0.029394	0.009707	0.009566	0.006109
61	dp2pl	✓		✓				✓			0.005898	0.007452	0.018323	0.024905	0.037121	0.022071	0.032087	0.009369	0.009553	0.005969
62	dp2pl	✓				✓	✓				0.005727	0.007697	0.018969	0.024835	0.039337	0.020916	0.033408	0.009516	0.009421	0.005853
63	dp2pl	✓		✓				✓			0.005435	0.007499	0.017489	0.024536	0.037218	0.023904	0.030789	0.010810	0.009539	0.005610
64	dp2pl	✓		✓				✓			0.005430	0.007522	0.017337	0.024596	0.037669	0.021570	0.032514	0.009204	0.009761	0.005888
65	dp2pl		✓	✓				✓			0.006126	0.007785	0.018930	0.024473	0.037376	0.020387	0.031642	0.009717	0.009629	0.006222
66	dp2pl		✓	✓			✓				0.005793	0.007778	0.018371	0.025085	0.037267	0.022013	0.033756	0.010536	0.009433	0.006339
67	dp2pl		✓	✓				✓			0.005541	0.007559	0.017941	0.025760	0.037941	0.021834	0.033200	0.009573	0.009766	0.006228
68	dp2pl		✓	✓	✓			✓			0.006124	0.007677	0.018178	0.024597	0.038503	0.021439	0.032693	0.009787	0.009886	0.006264
69	dp2pl		✓			✓	✓				0.005880	0.007591	0.018912	0.025686	0.037679	0.021164	0.032451	0.010016	0.009556	0.005944
70	pl2pl	✓		✓		✓					0.038084	0.821131	0.731518	0.841434	1.188160	0.259967	0.233275	0.352397	0.577785	0.095774
71	pl2pl	✓		✓			✓				0.126949	1.382720	2.920050	1.159370	1.205060	0.281119	0.326554	30.414900	0.470178	0.108995
72	pl2pl	✓		✓				✓			0.048356	3.898750	1.046420	0.556596	0.905365	0.139597	0.332312	0.483662	0.504652	0.088373
73	pl2pl	✓				✓	✓				0.033716	1.355810	0.660329	0.411051	1.569620	0.228070	0.157052	0.545376	0.590531	0.083949
74	pl2pl	✓		✓		✓	✓				0.039876	2.682710	0.520756	0.732706	0.631501	0.221493	0.184822	1.353890	0.930525	0.086568
75	pl2pl	✓		✓				✓			0.062516	1.133540	0.563656	0.728894	failure	0.201277	0.159906	0.522184	0.582346	0.114169
76	pl2pl		✓	✓			✓				0.040424	1.017110	4.496860	0.620658	2.091090	0.284072	0.141320	0.779983	0.623403	0.115250
77	pl2pl		✓	✓				✓			0.022890	1.859610	2.978900	0.266861	0.828823	0.227546	0.216731	0.494020	0.584247	0.082855
78	pl2pl		✓	✓	✓			✓			0.108108	1.317460	5.250300	0.620177	1.076750	0.209272	0.182869	0.339580	0.529837	0.091802
79	pl2pl		✓			✓	✓				0.089963	1.287350	0.694254	0.218081	1.358920	0.232178	0.175796	0.344242	0.584814	0.069141
80	pl2pl		✓	✓	✓						0.057583	1.508330	1.324110	1.747990	1.060200	0.190222	0.219271	0.365754	0.797830	0.117689
81	pl2pl		✓	✓				✓			0.037530	2.075200	2.683140	1.099220	0.981518	0.190410	0.205613	0.387148	0.663376	0.093042
82	pgslam(ndt)	✓		✓			✓				0.005734	0.141487	0.025997	0.036760	0.039573	0.065898	0.066544	0.009504	0.007320	
83	pgslam(p2p)	✓		✓			✓				0.007203	0.006923	0.021017	0.025040	0.035665	0.037463	0.065898	0.066544	0.009622	0.006588
84	pgslam(p2proj)	✓		✓			✓				0.006383	0.006923	0.019568	0.025289	0.037007	0.016561	0.065898	0.066544	0.009503	0.005152
85	pgslam(p2pl)	✓		✓			✓				0.006482	failure	0.019693	0.025301	0.037474	0.019672	0.065898	0.066544	0.010362	0.005221
86	pgslam(dp2pl)	✓		✓			✓				0.006368	0.006923	0.019693	0.025248	0.036982	0.015661	0.065898	0.066544	0.010226	0.005143
87	pgslam(pl2pl)	✓		✓			✓				0.076314	0.006923	1.284440	0.312282	2.093590	0.178237	0.130777	0.066544	0.486067	0.095490
88	pgslam(ndt, left-J)	✓		✓			✓			✓	0.005714	0.006923	0.256255	0.026000	0.036766	0.065972	0.065898	0.066544	0.009503	0.007322
89	pgslam(ndt, right-J)	✓		✓			✓			✓	0.005714	0.006923	0.662496	0.025999	0.036766	0.039572	0.065898	0.066544	0.009503	0.007322
90	pgslam(p2p, left-J)	✓		✓			✓			✓	0.007197	0.006923	0.020071	0.025063	0.035947	0.038302	0.065898	0.066544	0.009720	0.006987
91	pgslam(p2p, right-J)	✓		✓			✓				0.007196	0.006923	0.020070	0.025063	0.035947	0.038298	0.065898	0.066544	0.009720	0.006987
92	pgslam(p2proj, left-J)	✓		✓			✓			✓	0.007024	0.006923	0.019677	0.025862	0.035607	0.040195	0.065898	0.066544	0.009735	0.006713
93	pgslam(p2proj, right-J)	✓		✓			✓				0.007034	0.006923	0.019704	0.025322	0.035756	0.039319	0.065898	0.066544	0.009759	0.006960
94	pgslam(icp, left-J)	✓		✓			✓				0.027079	0.006927	0.050230	0.077403	0.036825	0.060648	0.065891	0.066551	0.051459	0.008153
95	pgslam(icp, pcl)	✓		✓			✓				0.009673	0.006930	0.028057	0.036526	0.037275	0.027275	0.065898	0.066542	0.019635	0.007461
96	pgslam(ndt, pcl, GTSAM)	✓		✓				✓	✓		0.035978	0.006923	0.036396	0.080154	0.138657	0.042461	0.027059	0.066544	0.034244	0.008372
97	pgslam(icp, pcl, GTSAM)	✓		✓				✓	✓		0.009553	0.006923	0.023153	0.025363	0.036594	0.025298	0.047268	0.066544	0.010367	0.004268
98	pgslam(ndt, pcl, manif)	✓		✓				✓			0.015985	0.006923	0.019648	0.080152	0.038795	0.030899	0.065898	0.066544	0.034248	0.008405
99	pgslam(icp, pcl, manif)	✓		✓				✓			0.009565	0.006923	0.019648	0.025392	0.036161	0.023749	0.065898	0.066544	0.010751	0.004329

pose graph SLAM. The key element of the NDT is the representation of the data as a set of normal distributions organized in the regular grid over 3D space. These distributions describe the probability of finding a 3D point at a certain position. The advantage of the method is that it gives a smooth representation of the point cloud, with continuous first and second-order derivatives. Thus, standard optimization techniques described in this paper can be applied. Another advantage of NDT over ICP is its much less computational complexity since the consumptive nearest neighborhood search procedure is not needed. The 3D space decomposition into the regular grid introduces some minor artefacts, but in a presented experiment it is a negligibly small disadvantage. For each bucket from a regular grid containing a sufficient number of measured points NDT calculates the mean given by Eq. (63) and the covariance given by Eq. (64).
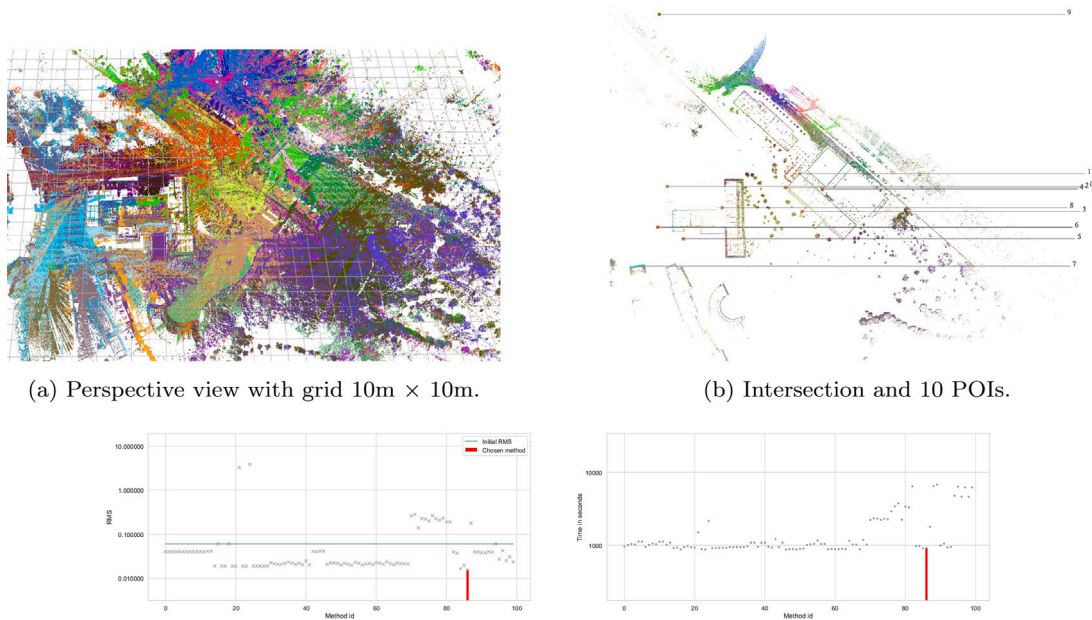
$$\boldsymbol{\mu} = \frac{1}{m} \sum_{k=1}^{m} \boldsymbol{P}_k^g \tag{63}$$

$$\boldsymbol{\Sigma} = \frac{1}{m-1} \sum_{k=1}^{m} (\boldsymbol{P}_k^g - \boldsymbol{\mu})(\boldsymbol{P}_k^g - \boldsymbol{\mu})^{\mathsf{T}} \tag{64}$$

The likelihood of having measured point $\boldsymbol{P}_m^g$ is given by Eq. (65).

$$p(\boldsymbol{P}_m^g) = \frac{1}{(2\Pi)^{\frac{1}{2}} \sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{(\boldsymbol{P}_m^g - \boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{P}_m^g - \boldsymbol{\mu})}{2}\right) \tag{65}$$
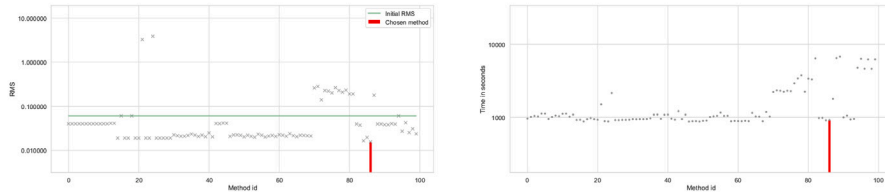
Each $p(\boldsymbol{P}_m^g)$ can be seen as an approximation of the local surface within the range of the bucket. It describes the position $\boldsymbol{\mu}$ of the surface as well as its orientation and smoothness given by $\boldsymbol{\Sigma}$. Let $\boldsymbol{\Psi}([\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}^{3 \times 4}, \boldsymbol{P}_m^l)$ will be a transformation function of the local measurement point $[\boldsymbol{P}_m^l, 1]^{\mathsf{T}}$ via pose $[\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}^{3 \times 4}$ expressed as (66).

$$\boldsymbol{\Psi}([\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}^{3 \times 4}, \boldsymbol{P}_m^l) = \boldsymbol{P}_m^g = [\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}^{3 \times 4} \begin{bmatrix} \boldsymbol{P}_m^l \\ 1 \end{bmatrix} \tag{66}$$
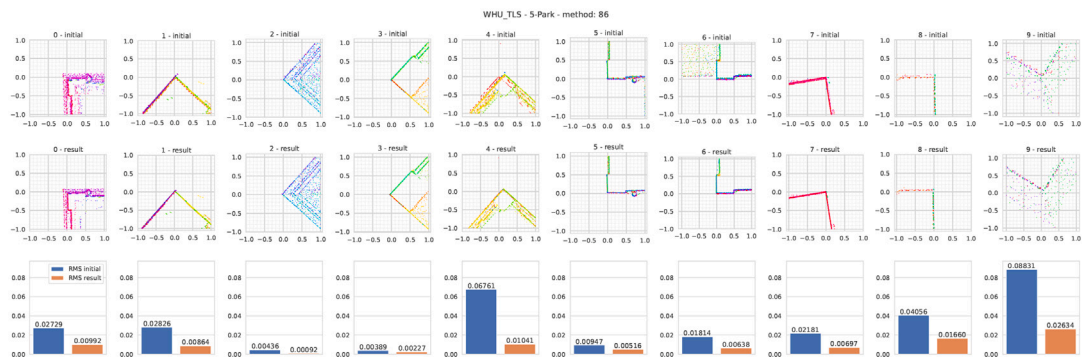
(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

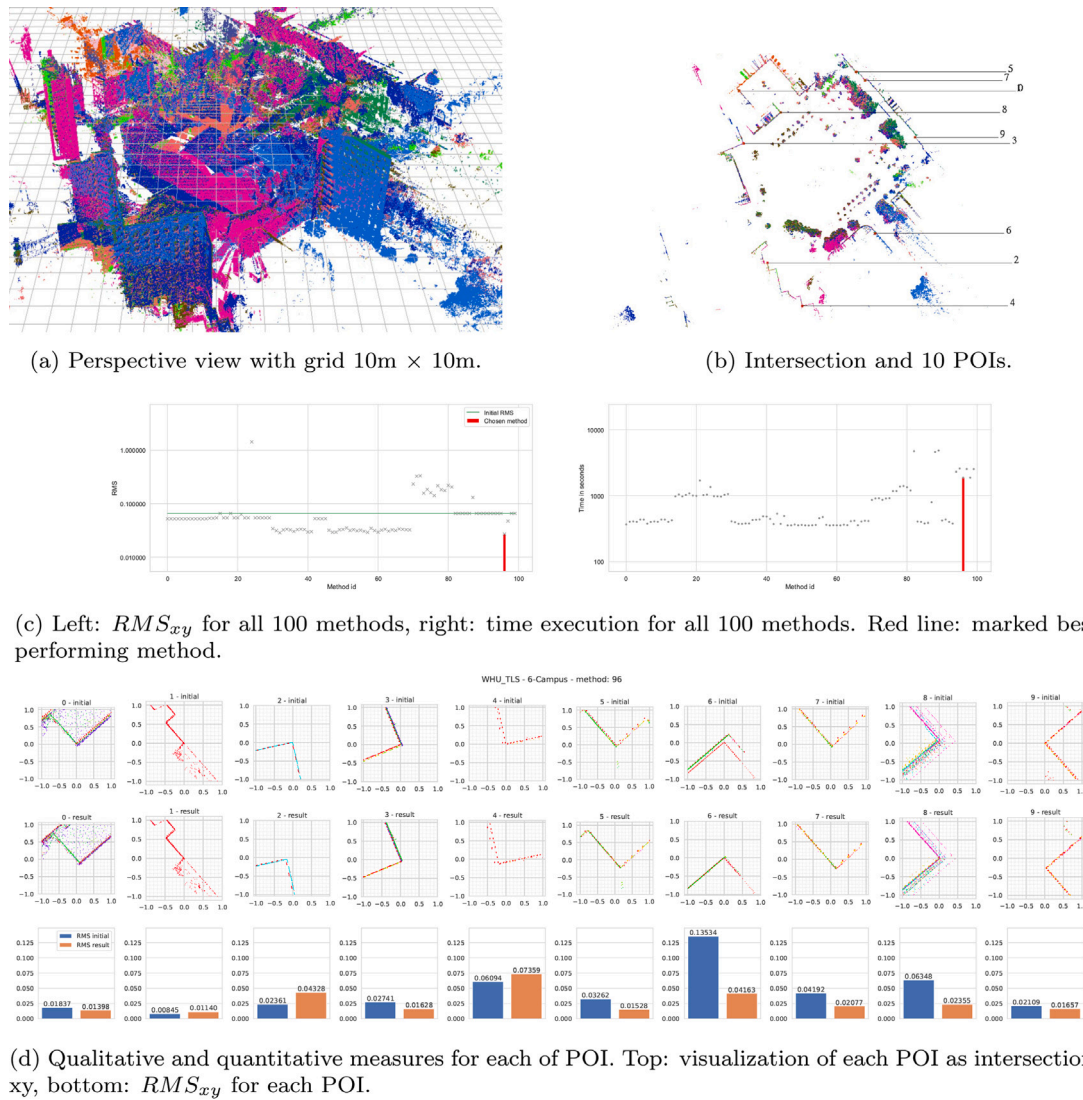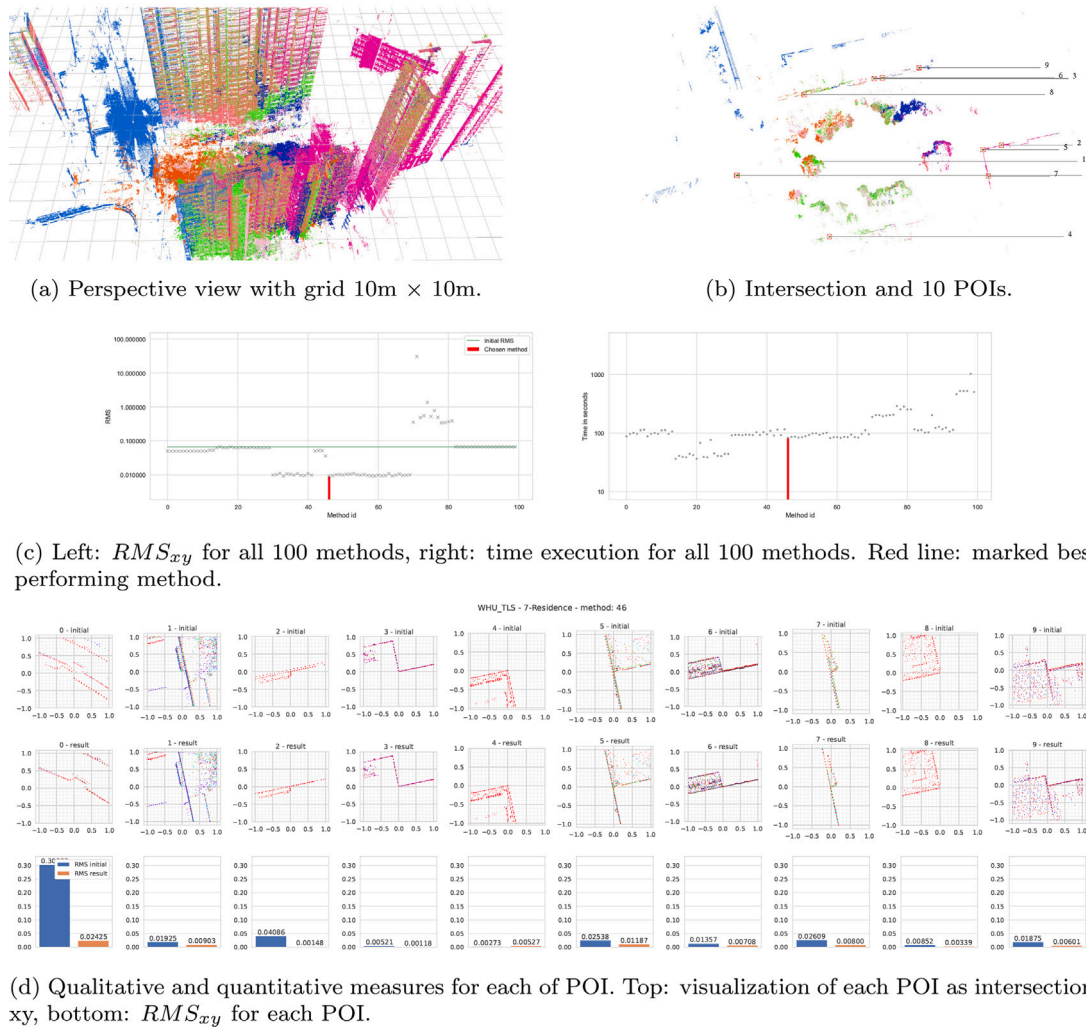(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.
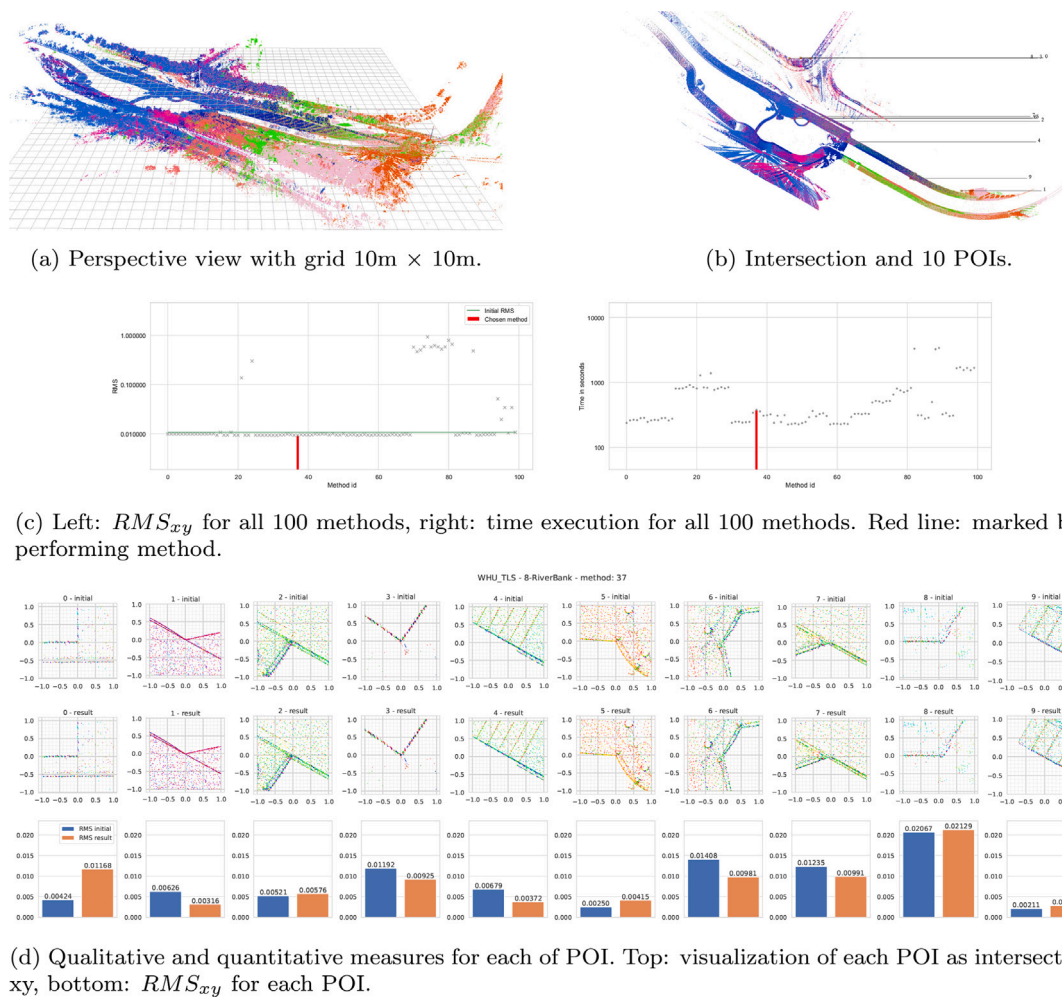
Fig. 18. WHU_TLS data set [12], scene: 5-Park. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved. Moreover, it is great example of the large scale survey challenge.

Thus, the NDT optimization problem is defined as the maximization of the likelihood function given in Eq. (67).

$$[\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4, *}_{W \leftarrow TLS} = \max_{[\boldsymbol{R}, \boldsymbol{t}]_{W \leftarrow TLS}} \prod_{k=1}^{N} p(\boldsymbol{\Psi}([\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4}_{W \leftarrow TLS}, \boldsymbol{P}^{l}_{m})) \qquad (67)$$

Furthermore, the optimization problem is equivalent to the minimization of the negative log-likelihood given in Eq. (68).

$$[\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4, *}_{W \leftarrow TLS} = \min_{[\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4}_{W \leftarrow TLS}} - \sum_{k=1}^{N} \log \left(p(\boldsymbol{\Psi}([\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4}_{W \leftarrow TLS}, \boldsymbol{P}^{l}_{m})) \right) \qquad (68)$$

The proposed NDT implementation is using already discussed point-to-point observation equation. The target value is $[0, 0, 0]^{\intercal}$ and the model function is $\boldsymbol{\Psi}([\boldsymbol{R}, \boldsymbol{t}]^{3 \times 4}_{W \leftarrow TLS}, \boldsymbol{P}^{l}_{m}) - \boldsymbol{\mu}$. The information matrix $\boldsymbol{\Omega}$ is calculated as an inverse of the covariance matrix from Eq. (64). An extension to a multi pose optimization problem is straightforward and it is advised to study an open-source project supporting this paper.

3. Experiments

Experiments include a benchmark of 100 variations of multi-view TLS data registrations. It is available and maintained as an open-source project [9]. All quantitative measures obtained with a representative subset of the possible multi-view TLS data registration approaches are collected in Tables 3–5 as RMS_{xy} (Eq. (69) [11]).

$$RMS_{xy} = \sqrt{\frac{1}{C} \sum_{i \in C_s, j \in C_t} (p_i - q_i)^2}, p \in S, q \in T \qquad (69)$$

Eq. (69) conducts all point pairs for all POIs (Point of Interest) and it is denoted as (source S: p, target T: q), a sum of this pairs is C. It uses only x and y coordinates. Three publicly available data sets are incorporated ETH [11], RESSO [10] and WHU_TLS [12] (see Fig. 2 and Table 2). ETH and RESSO data sets provide registered data obtained with ICP. Data registration for WHU_TLS is calculated with method from [49]. The goal of the presented benchmark is to conduct data registration for all of those data sets and provide quantitative and qualitative measures based on the following steps:

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 19. WHU_TLS data set [12], scene: 6-Campus. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved. Moreover, it is another example of the large scale survey challenge.

- for each scan the down-sampling is performed with a grid of 10 cm × 10 cm × 10 cm cells resulting remained single point per cell,
- assign 10 POIs, being virtual corners or easily identified points in the point cloud intersection (manual step),
- performs automatic multi-view TLS data registration with chosen 100 variants,
- choose the best performing method (minimal RMS_{xy}),
- generate intersections for all POIs for qualitative comparison,
- compute RMS_{xy} for each POI for quantitative comparison,
- performs a subjective investigation and adds to final remarks.

Quantitative and qualitative results are organized into three sub-sections related to different reference data sources. The calculations for methods 1–93 were conducted by a laptop with AMD Ryzen 7 5800H CPU 3.20 GHz, 64 GB RAM. The calculations for methods 94–99 were conducted by a laptop with Intel Core i7-7700HQ CPU 2.80 GHz, 32 GB RAM. The parameters for all methods are set as equal as it was possible e.g. the number of iterations, search radius and other initial parameters are the same. The first step (down-sampling) is mandatory for incorporating state-of-the-art computers for large-scale experiments. Future work will be related to the full resolution

of data. It will require more technical work being out of the paper's scope. The tool for the manual selection of control points is designed to minimize ambiguity. Thus, manually generated intersections for all POIs are as precise as it was possible. Moreover, a similar approach is evident in literature [11] and it does not affect the general outcome of this research. The most important insight from this is that in most cases proposed framework improves state-of-the-art registered data. It is considered a major contribution of this paper.

3.1. ETH data set

This data set is composed of five scenes: arch, courtyard, facade, office and trees [11]. Table 3 shows that only arch scene (Fig. 3) is difficult to refine where only 3% of conducted methods perform better than provided reference registration. Columns in Tables 3, 4, 5 describe following metadata

- 'id' is identification number of the registration method, the initial values of RMS_{xy} for reference method are given in first raw,
- 'GN' is Gauss–Newton,
- 'LM' is Levenberg–Marquardt,
- '$W \leftarrow TLS$' is frame of Terrestrial Laser Scanner to frame World,
- '$W \rightarrow TLS$' is frame World to frame of Terrestrial Laser Scanner,

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

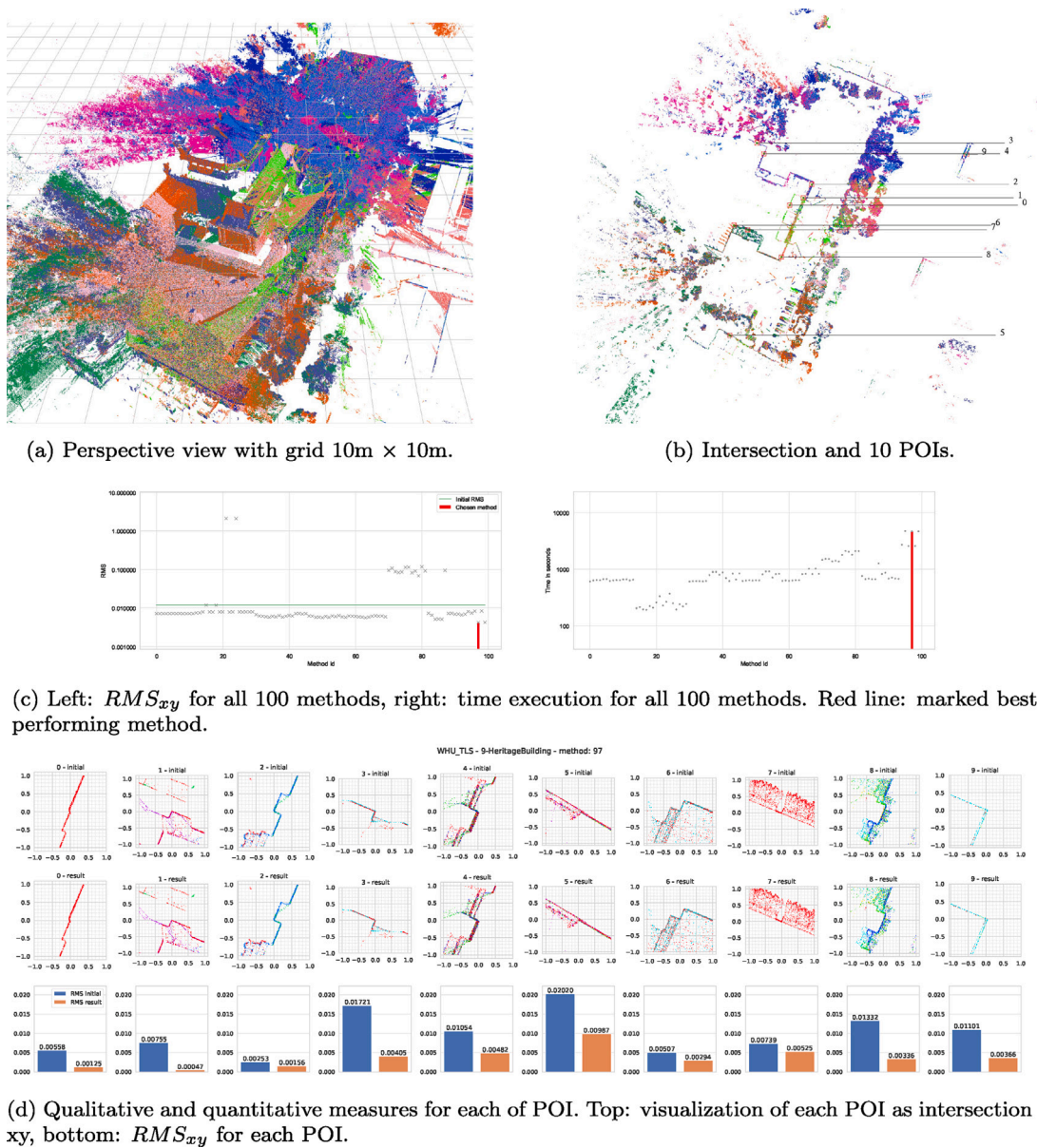**Fig. 20.** WHU_TLS data set [12], scene: 7-Residence. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved. Moreover, it is another example of the large scale survey challenge.

- 'TB' is Tait–Bryan angles,
- 'Q' is quaternion,
- 'R' is Rodrigues,
- '$left-J$' is Lie algebra left Jacobian,
- '$right-J$' is Lie algebra right Jacobian.
- ...RMS_{xy} denote errors.

Column 'method' contains following variants: 'p2p' is point to point, 'ndt' is Normal Distributions Transform, 'p2proj' is point to projection onto plane, 'p2pl' is point to plane using dot product, 'dp2pl' is distance point to plane, 'pl2pl' is plane to plane, 'pgslam' is pose graph SLAM, 'pcl' is Point Cloud Library [48], 'icp' is Iterative Closest Point [48], 'GTSAM' is pose graph SLAM [50], 'manif' is Lie algebra implementation [51,52]. Blue cells are winning methods, bold: RMS_{xy} smaller than reference means improved state of the art. For scene courtyard (Fig. 4) 83% of conducted methods perform better. For scenes facade (Fig. 5) and trees (Fig. 7) 84% of conducted methods perform better. For scene office (Fig. 6) 69% of conducted methods perform better. It can be observed that plane-to-plane metric provides poor results.

Fig. 3 shows the benchmark conducted on manually registered *arch* data set [11]. It can be seen that only three variants can slightly im-

prove reference data registration. This experiment shows only the fact that proposed methods are fair enough compared with the reference.

Fig. 4 shows an improvement of data set *courtyard* from initial $RMS_{xy} = 0.031$ down to $RMS_{xy} = 0.016$. Twice better registration is evident for distance point to plane metric. It can be seen that the metric is dominant factor from all of the registration variants. Important observation is that ndt, point to projection onto plane and point to plane using dot product variants provide comparable improvements. It can be justified that all of these registration variants are based on the same principle. It means that each source point is exactly measurement TLS point, target information is constructed from the neighboring points. In such scenario, where there is not so much planar shapes, all observation equations relying on this principle perform similar. It can be seen that performs not as efficient as other metrics since the data down sampling occurs.

Fig. 5 shows great improvement of data set *facade*. All registration variants improve such scenario with plenty planar shapes.

Fig. 6 shows small improvement of data set *office*. Most of the indoor open source datasets are well registered since they are rather small and easy to register with state of the art techniques.

Fig. 7 shows small improvement of data set *trees*. This scenario is challenging since there are plenty of noisy data. This data set is difficult to analyze quantitatively without additional source of ground

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 21. WHU_TLS data set [12], scene: 8-RiverBank. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that it is difficult to justify if we observe an improvement. It means that the reference data is well registered.

truth data. The benchmark shows that proposed variants of the data registration slightly improve reference one.

3.2. RESSO data set

This data set is composed of five scenes: figure_7a, figure_7b, figure_7c, figure_7d and figure_7e [10], where the reference data registration was obtained with combination of automatic approaches and manual registration. For scene figure_7a (Fig. 8) 58% of conducted methods perform better. It can be observed that Normal Distributions Transform and plane to plane methods provide poor results. For scene figure_7b (Fig. 9) 76% of conducted methods perform better. In figure_7c (Fig. 10) 84%, figure_7d (Fig. 11) 98% and in figure_7e (Fig. 12) 81% of conducted methods perform better. It can be observed that for almost all scenes (excluding figure_7d) plane to plane methods provide poor results.

Fig. 8 shows small improvement of the reference registration. An important observation is the fact that the result of ndt is rather poor. One we look closer to data set, plenty of vegetation is evident. Thus, the combination of planar shapes and noisy data is challenging for this metric.

Fig. 9 shows a very small improvement in the reference registration. On one hand, this particular case demonstrates only that the proposed set of variants is sufficient. On the other hand, this case demonstrates

the stability of the results. It can be considered empirical evidence that the proposed framework is reliable.

Figs. 10–12 show great improvement in the reference registration by all data registration variants. It can be considered empirical evidence that the proposed framework is reliable. For this reason, this benchmark can be used for data registration validation.

3.3. WHU_TLS data set

This data set is composed of ten scenes: 1-SubwayStation, 10-UndergroundExcavation, 11-Tunnel, 2-HighSpeedRailway, 3-Mountain, 5-Park, 6-Campus, 7-Residence, 8-RiverBank, 9-HeritageBuilding [12]. It is a very interesting data set since for two scenes 1-SubwayStation and 10-UndergroundExcavation there is no better result than reference registration. Other results are as follows: 11-Tunnel: 52%, 2-HighSpeedRailway: 80%, 3-Mountain: 74%, 5-Park: 83%, 6-Campus: 70%, 7-Residence:69%, 8-RiverBank: 78%, 9-HeritageBuilding: 83% of methods refine reference data. It can be observed as in previous trials that the plane-to-plane methods perform worse. Surprisingly, Normal Distributions Transform methods fail for 11-Tunnel scene.

Figs. 13, 14, 16, 17 and 21 show grate example of accurate data registration provided by [12]. No variant in the proposed benchmark can provide evident improvement of this data set according to chosen quantitative measure. The difference in RMS_{xy} are rather small,

(a) Perspective view with grid 10m × 10m.

(b) Intersection and 10 POIs.

(c) Left: RMS_{xy} for all 100 methods, right: time execution for all 100 methods. Red line: marked best performing method.

(d) Qualitative and quantitative measures for each of POI. Top: visualization of each POI as intersection xy, bottom: RMS_{xy} for each POI.

Fig. 22. WHU_TLS data set [12], scene: 9-HeritageBuilding. Comparison between reference data 'initial' and best performing method 'result'. It can be seen that this data set is much improved. Moreover, it is another example of the large scale survey challenge.

therefore at this stage of research, it is difficult to justify more than the statement that the proposed benchmark does not affect the consistency of these data.

Figs. 15, 18, 19, 20 and 22 show great improvements. Once we take a closer look at these particular data sets, we can see that long tunnels and large-scale complex data registration scenarios are still challenging for researchers. Fortunately, this fully automatic framework can refine this particular data sets by almost all registration variants.

The major outcome of this benchmark is that the recommended registration variant is ndt since its performance is most promising. Some issues should be addressed in future, especially removing noisy data (vegetation). Only ndt can reach scalability, therefore it can address large-scale scenarios.

4. Conclusion

Multi-view TLS (Terrestrial Laser Scanner) data registration is essential for many applications and domains such as geoscience, mobile robotics and computer vision. Many researchers from the mobile robotics domain claim high accuracy of delivered maps, thus it was a prerequisite for conduced research closing the gap between those domains. This paper verified the optimization criterion that relays on the metric e.g. point to point, point to projection onto a plane, plane to plane etc..., thus the sum of the distances between basic geometric primitives (or their parameterized form) is minimized. It was shown that it is possible to express SO(3) using Euler, Tait–Bryan, quaternion or Rodrigues' rotation formula and its impact on the data registration accuracy. The optimization can be performed locally or globally using the weighted nonlinear least squares method, Gauss–Newton, Levenberg–Marquardt algorithms including robust variants and based on probabilistic approaches e.g. Normal Distributions Transform, Generalized Iterative Closest Point and Pose Graph Simultaneous Localization and Mapping, but the impact into accuracy is rather negligible in considered scenarios. The combination of metrics, rotation matrix parameterization and optimization algorithms creates hundreds of possible approaches. It was shown that chosen metric is a dominant factor in data registration. An open-source project is created and it is

used for improving open-source data. The main claim is that most of the proposed approaches improve registered reference data provided by other researchers.

First and the same time most important insight from conducted research is the fact that only for 2 from 20 scenes the proposed set of 100 multi-view TLS data registration variants is not capable of providing better results than reported registered reference data. The second fact is that the largest improvements are evident for large-scale scenes, which shows future research direction. The third fact is that 7 scenes can be improved by already existing open-source software, thus this benchmark is pushing the state of the art forward and it is an added value for the research community.

The final statement after this benchmark is that the recommended data registration variant is multi-view Normal Distributions Transform. It is scalable and reliable. The rotation parameterization and other degrees of freedom of proposed variants are rather negligible compared with chosen metric. This information is the main contribution of this paper. This framework and benchmark are available and maintained at https://github.com/MapsHD/HDMapping.

CRediT authorship contribution statement

Janusz Będkowski: Conceptualization, Methodology, Data curation, Writing – original draft, Visualization, Investigation, Supervision, Software, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

data links are available

Acknowledgment

References

[1] Andreas Nuchter, Robotic 3D scan repository, 2021.

[2] UNAVCO, Terrestrial laser scanning (TLS) data, 2021.

[3] Washington State Department of Transportation, 3D terrestrial laser scanning, 2021.

[4] Jose Gonzalez de Tanago, Alvaro Lau, Harm Bartholomeus, Martin Herold, Valerio Avitabile, Pasi Raumonen, Christopher Martius, Rosa C. Goodman, Mathias Disney, Solichin Manuri, Andrew Burt, Kim Calders, Estimation of above-ground biomass of large tropical trees with terrestrial lidar, Methods Ecol. Evol. 9 (2) (2018) 223–234.

[5] Hao Wu, Li Yan, Hong Xie, Pengcheng Wei, Jicheng Dai, A hierarchical multiview registration framework of tls point clouds based on loop constraint, ISPRS J. Photogramm. Remote Sens. 195 (2023) 65–76.

[6] Xufei Wang, Zexin Yang, Xiaojun Cheng, Jantien Stoter, Wenbing Xu, Zhenlun Wu, Liangliang Nan, Globalmatch: Registration of forest terrestrial point clouds by global matching of relative stem positions, ISPRS J. Photogramm. Remote Sens. 197 (2023) 71–86.

[7] Jonathan T. Barron, A general and adaptive robust loss function, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 4326–4334.

[8] Jihua Zhu, Di Wang, Jiaxi Mu, Huimin Lu, Zhiqiang Tian, Zhongyu Li, 3dmndt:3d multi-view registration method based on the normal distributions transform, 2021.

[9] Janusz Bedkowski, 2023. https://github.com/mapshd/hdmapping.

[10] Songlin Chen, Liangliang Nan, Renbo Xia, Jibin Zhao, Peter Wonka, Plade: A plane-based descriptor for point cloud registration with small overlap, 58, (4) 2020, pp. 2530–2540.

[11] Pascal Willy Theiler, Jan Dirk Wegner, Konrad Schindler, Keypoint-based 4-points congruent sets – automated marker-less registration of laser scans, ISPRS J. Photogramm. Remote Sens. 96 (2014) 149–163.

[12] Zhen Dong, Fuxun Liang, Bisheng Yang, Yusheng Xu, Yufu Zang, Jianping Li, Yuan Wang, Wenxia Dai, Hongchao Fan, Juha Hyyppä, Uwe Stilla, Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark, ISPRS J. Photogramm. Remote Sens. 163 (2020) 327–342.

[13] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Konrad Schindler Andreas Wieser, Predator: Registration of 3d point clouds with low overlap, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2021.

[14] Pierre Grussenmeyer, Emmanuel Alby, Pierre Assali, Valentin Poitevin, Jean-François Hullo, Eddie Smigiel, Accurate documentation in cultural heritage by merging TLS and high-resolution photogrammetric data, in: Fabio Remondino, Mark R. Shortis (Eds.), Videometrics, Range Imaging, and Applications XI, in: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 8085, 2011, 808508.

[15] Seung Woo Son, Dong Woo Kim, Woong Gi Sung, Jae Jin Yu, Integrating UAV and TLS approaches for environmental management: A case study of a waste stockpile area, Remote Sens. 12 (10) (2020).

[16] Balasubramanian Muralikrishnan, Performance evaluation of terrestrial laser scanners; a review, 2021, 2021.

[17] Simon J. Buckley, J.A. Howell, H.D. Enge, T.H. Kurz, Terrestrial laser scanning in geology: data acquisition, processing and accuracy considerations, J. Geol. Soc. 165 (3) (2008) 625–638.

[18] Markku Åkerblom, Pekka Kaitaniemi, Terrestrial laser scanning: a new standard of forest measuring and modelling? Ann. Botany 128 (6) (2021) 653–662.

[19] Rafael Rodrigues Cunha, Claude Thiago Arrabal, Marcelo Mourão Dantas, Hélio Rodrigues Bassanelli, Laser scanner and drone photogrammetry: A statistical comparison between 3-dimensional models and its impacts on outdoor crime scene registration, Forensic Sci. Int. 330 (2022) 111100.

[20] Chan-Seong Park, Hong-Pil Jeon, Kwang-Soo Choi, Jin-Pyo Kim, Nam-Kyu Park, Application of 3d laser scanner to forensic engineering, J. Forensic Sci. 63 (3) (2018) 930–934.

[21] Ján Šašak, Michal Gallay, Ján Kaňuk, Jaroslav Hofierka, Jozef Minár, Combined use of terrestrial laser scanning and uav photogrammetry in mapping alpine terrain, Remote Sens. 11 (2019) 2154.

[22] Jacek Katzer, Czesław Suchocki, Wioleta Błaszczak-Bąk, Marzena Damięcka-Suchocka, Intensity dataset acquired through laser scanning of lunar and martian soil simulants, Data Brief 39 (2021) 107616.

[23] M. Saponaro, L. Pratola, A. Capolupo, A. Saponieri, L. Damiani, U. Fratino, E. Tarantino, Data fusion of terrestrial laser scanner and remotely piloted aircraft systems points clouds for monitoring the coastal protection systems, Aquat. Ecosyst. Health Manag. 23 (4) (2020) 389–397.

[24] Zhihua Xu, Ershuai Xu, Lixin Wu, Shanjun Liu, Yachun Mao, Registration of terrestrial laser scanning surveys using terrain-invariant regions for measuring exploitative volumes over open-pit mines, Remote Sens. 11 (6) (2019).

[25] Zhen Dong, Fuxun Liang, Bisheng Yang, Yusheng Xu, Yufu Zang, Jianping Li, Wang Yuan, Wenxia Dai, Hongchao Fan, Juha Hyyppäb, Uwe Stilla, Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark, ISPRS J. Photogramm. Remote Sens. 163 (2020) 327–342.

[26] Miro Demol, Kim Calders, Hans Verbeeck, Bert Gielen, Forest above-ground volume assessments with terrestrial laser scanning: a ground-truth validation experiment in temperate, managed forests, Ann. Botany 128 (6) (2021) 805–819.

[27] Oliver Wulf, Andreas Nuchter, Joachim Hertzberg, Bernardo Wagner, Ground truth evaluation of large urban 6d slam, 2007, pp. 650–657.

[28] Michael Helmberger, Kristian Morin, Beda Berner, Nitish Kumar, Giovanni Cioffi, Davide Scaramuzza, The hilti SLAM challenge dataset, IEEE Robotics Autom. Lett. 7 (3) (2022) 7518–7525.

[29] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, Andrew W. Fitzgibbon, Bundle adjustment - a modern synthesis, in: Proceedings of the International Workshop on Vision Algorithms: Theory and Practice, ICCV '99, Springer-Verlag, Berlin, Heidelberg, 1999, pp. 298–372.

[30] Zhen Dong, Bisheng Yang, Fuxun Liang, Ronggang Huang, Sebastian Scherer, Hierarchical registration of unordered tls point clouds based on binary shape context descriptor, ISPRS J. Photogramm. Remote Sens. 144 (2018) 61–79.

[31] P.J. Besl, Neil D. McKay, A method for registration of 3-d shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (2) (1992) 239–256.

[32] G. Grisetti, R. Kuemmerle, C. Stachniss, W. Burgard, A tutorial on graph-based SLAM, Intell. Transp. Syst. Mag. IEEE 2 (4) (2010) 31–43.

[33] N. Sunderhauf, P. Protzel, Switchable constraints for robust pose graph slam, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 1879–1884.

[34] R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, W. Burgard, g2o: A general framework for graph optimization, in: Proc. of the IEEE Int. Conf. on Robotics and Automation, ICRA, Shanghai, China, 2011.

[35] Sıddıka Karakuş, Halit Gündoğan. C., 2010. pages 367–377.

[36] L. Euler, Nova methodus motum corporum rigidorum determinandi, Novi Comment. Acad. Imp. Petrop. 20 (1775) (1775) 208–238.

[37] O. Rodrigues, Des lois géométriques qui régissent les déplacements d'un systéme solide dans l'espace, et de la variation des coordonnees provenant de ces déplacements consideres indépendamment des causes qui peuvent les produire, J. Math. 5 (1840) (1840) 380–440.

[38] J.S. Dai, Geometrical Foundations and Screw Algebra for Mechanisms and Robotics (Translated from Screw Algebra and Kinematic Approaches for Mechanisms and Robotics, 2016), Higher Education Press, Beijing, ISBN: 9787040334838, 2014.

[39] K.K. Hunt, J. Davidson, Robots and Screw Theory, Applications of Kinematics and Statics to Robotics, Oxford University Press, New York, 2004.

[40] Ken Shoemake, Animating rotation with quaternion curves, SIGGRAPH Comput. Graph 19 (3) (1985) 245–254.

[41] Berthold K.P. Horn, Closed-form solution of absolute orientation using unit quaternions, J. Opt. Soc. Amer. A 4 (4) (1987) 629–642.

[42] Berthold K.P. Horn, Hugh M. Hilden, Shahriar Negahdaripour, Closed-form solution of absolute orientation using orthonormal matrices, J. Opt. Soc. Amer. A 5 (7) (1988) 1127–1135.

[43] D.Q.F. de Menezes, D.M. Prata, A.R. Secchi, J.C. Pinto, A review on robust m-estimators for regression analysis, Comput. Chem. Eng. 147 (2021) 107254.

[44] Zhengyou Zhang, Parameter estimation techniques: a tutorial with application to conic fitting, Image Vis. Comput. 15 (1) (1997) 59–76.

[45] Nived Chebrolu, Thomas Läbe, Olga Vysotska, Jens Behley, Cyrill Stachniss, Adaptive robust kernels for non-linear least squares problems, 2020, CoRR, abs/2004.14938.

[46] Joan Sola, Jeremie Deray, Dinesh Atchuthan, A micro lie theory for state estimation in robotics, 2020.

[47] Martin Magnusson, Achim J. Lilienthal, Tom Duckett, Scan registration for autonomous mining vehicles using 3d-ndt, J. Field Robotics 24 (10) (2007) 803–827.

[48] R.B. Rusu, S. Cousins, 3d is here: Point cloud library (pcl), in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, 2011, pp. 1–4.

[49] John Williams, Mohammed Bennamoun, Simultaneous registration of multiple corresponding point sets, Comput. Vis. Image Underst. 81 (1) (2001) 117–142.

[50] Michael Kaess, Gtsam library, 2015.

[51] Jérémie Deray, Joan Solà, Manif: A micro Lie theory library for state estimation in robotics applications, J. Open Source Softw. 5 (46) (2020) 1371.

[52] Joan Solà, Jeremie Deray, Dinesh Atchuthan, A Micro Lie Theory for State Estimation in Robotics, Technical Report IRI-TR-18-01, Institut de Robòtica i Informàtica Industrial, Barcelona, 2018.