

**PLADE**

**A Plane-Based Descriptor for Point Cloud Registration with Small Overlap**

Chen, Songlin; Nan, Liangliang; Xia, Renbo; Zhao, Jibin; Wonka, Peter

**DOI**

[10.1109/TGRS.2019.2952086](https://doi.org/10.1109/TGRS.2019.2952086)

**Publication date**

2020

**Document Version**

Final published version

**Published in**

IEEE Transactions on Geoscience and Remote Sensing

**Citation (APA)**

Chen, S., Nan, L., Xia, R., Zhao, J., & Wonka, P. (2020). PLADE: A Plane-Based Descriptor for Point Cloud Registration with Small Overlap. *IEEE Transactions on Geoscience and Remote Sensing*, 58(4), 2530-2540. <https://doi.org/10.1109/TGRS.2019.2952086>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

***Green Open Access added to TU Delft Institutional Repository***

***'You share, we take care!' - Taverne project***

**<https://www.openaccess.nl/en/you-share-we-take-care>**

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

# PLADE: A Plane-Based Descriptor for Point Cloud Registration With Small Overlap

Songlin Chen, Liangliang Nan<sup>ID</sup>, Renbo Xia, Jibin Zhao, and Peter Wonka<sup>ID</sup>

**Abstract**—Traditional point cloud registration methods require large overlap between scans, which imposes strict constraints on data acquisition. To facilitate registration, users have to carefully position scanners to ensure sufficient overlap. In this article, we propose to use high-level structural information (i.e., plane/line features and their interrelationship) for registration, which is capable of registering point clouds with small overlap, allowing more freedom in data acquisition. We design a novel plane/line-based descriptor dedicated to establishing structure-level correspondences between point clouds. Based on this descriptor, we propose a simple but effective registration algorithm. We also provide a data set of real-world scenes containing a larger number of scans with a wide range of overlap. Experiments and comparisons with state-of-the-art methods on various data sets reveal that our method is superior to existing techniques. Though the proposed algorithm outperforms state-of-the-art methods on the most challenging data set, the point cloud registration problem is still far from being solved, leaving significant room for improvement and future work.

**Index Terms**—Data set, descriptor, point cloud, registration, scanning.

## I. INTRODUCTION

THE proliferation of acquisition devices (e.g., laser scanners and depth cameras) enables us to quickly obtain a massive volume of 3-D point clouds of indoor and outdoor environments. The obtained point clouds have many applications in computer vision and computer graphics, including navigation and virtual/augmented reality. The nature of the

scanning process typically results in a set of randomly oriented point clouds captured from different viewpoints, waiting to be registered. Although the registration problem has been extensively studied in the last decades, it still remains an open problem due to three main reasons.

First, existing methods assume sufficient overlap between point clouds, which imposes restrictions on the scanning process, i.e., the user has to strategically position or move the scanner to ensure proper overlap between scans, making data acquisition a challenging task [1], [2]. In realistic scanning conditions, it is quite common that scans with insufficient overlap are obtained. This issue becomes vital when a scene is simultaneously scanned by multiple scanners and users. Another important scenario is when one wants to obtain complete scans of a scene, the user may apply a static laser scanner to capture the major part of the scene and a mobile scanner to complete the occluded regions. Scanning in such a fashion typically leads to a global point cloud and a set of local point clouds capturing local regions of the scene. These scans often have a too small overlap for traditional registration methods to succeed.

Second, traditional registration methods focus on establishing correspondences between point clouds using local salient features. However, man-made scenes, such as building interiors and exteriors comprising mainly planar structures, are common in the real world [3], for which sufficient descriptive local features cannot be extracted for registration [4].

Third, developing reliable point cloud registration approaches brings up significant challenges in evaluation tasks that involve capturing massive data sets and providing ground-truth registrations. Unfortunately, very limited data sets are available and are typically created for specific environments (e.g., urban scenes) by using a single type of scanner (e.g., high-range laser scanners) [5], [6] and typically have only a few scan pairs. The lack of diverse data sets (e.g., different environments and acquired using different sensors) and accurate ground-truth has caused various point cloud registration techniques to be poorly and unfairly evaluated [7], [8]. In fact, existing techniques can only be evaluated against small carefully crafted data sets.

In this article, we address the problem of registering point clouds with small overlap captured from real-world scenes. Since sufficient overlap and descriptive features cannot be guaranteed, our approach relies on high-level structures of the scene for registration. Specifically, man-made environments typically consist of a planar structure; thus, we represent the main structures of the scene as a collection of planes.

Manuscript received June 5, 2019; revised August 12, 2019 and October 2, 2019; accepted October 21, 2019. Date of publication December 18, 2019; date of current version March 25, 2020. This work was supported in part by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research under Award OSR-CRG2018-3730 and in part by the Visual Computing Center (VCC), KAUST. The source code of PLADE is available at <https://github.com/chsl/PLADE>, and the dataset is available at <https://3d.bk.tudelft.nl/liangliang/publications/2019/plade/resso.html>. (Corresponding author: Liangliang Nan.)

S. Chen was with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia, and also with the Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China. He is now with Huawei Technologies Co., Ltd., Nanjing 210012, China.

L. Nan is with the Faculty of Architecture and the Built Environment, Delft University of Technology, 2628 BL Delft, The Netherlands (e-mail: [liangliang.nan@gmail.com](mailto:liangliang.nan@gmail.com)).

R. Xia and J. Zhao are with the State Key Laboratory of Robotics, Shenyang Institute of Automation and the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China.

P. Wonka is with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia.

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2019.2952086

0196-2892 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.  
See <https://www.ieee.org/publications/rights/index.html> for more information.

These planar structures along with their interrelations reveal high-level global characteristics of the scene and we believe that they provide sufficient information for registration. While there exists a fair amount of previous work using plane-/line-based features, the robustness of existing plane-based methods is still not satisfactory [9]–[14]. Our work proposes a plane-/line-based descriptor to establish structure-level correspondences between point clouds, with which robust registration can be effectively achieved.

In addition to the simple but effective registration algorithm, we provide a benchmark data set scanned from a set of indoor and outdoor scenes with varying overlapping ratios, complementing existing data sets. As for evaluation, the performance of a registration method can be simply measured by the percentage of the successfully registered scans. Though experiments demonstrate that our method significantly outperforms the state of the art, a large portion of point clouds still remains unregistered. This indicates that the registration problem is far from being solved, allowing significant room for improvement. In summary, our main contributions include the following.

- 1) A novel plane-/line-based descriptor dedicated to establishing structure-level correspondences between point clouds.
- 2) A robust and fast point cloud registration algorithm using the plane-/line-based descriptor, which significantly outperforms the state of the art.
- 3) A benchmark data set for evaluating point cloud registration algorithms. Our data set contains scans with varying overlap, posing interesting challenges for research in point cloud registration.

## II. RELATED WORK

Point cloud registration methods can be roughly classified into two categories: coarse registration and fine registration. Fine registration algorithms aim to improve a given initial coarse registration. Such algorithms include iterative closest point (ICP) [15] and its variants [16]–[19]. In contrast, the inputs to the coarse registration algorithms are point clouds with unknown orientations. Thus, coarse registration is considered more challenging and has been receiving increasing attention in the past years. Our method falls into the coarse registration category. Therefore, in this section, we mainly discuss recent work on coarse registration, in particular, algorithms on local descriptor-based registration, global feature-based registration, and registration without overlap. For a comprehensive review of general point cloud registration algorithms, please refer to the survey by Maiseli *et al.* [8].

### A. Local Descriptor-Based Registration

Algorithms in this category are most popular in point cloud registration. These algorithms focus on using/defining local salient point features (i.e., transformation invariant descriptors) to establish pointwise correspondences between subsets (i.e., sets of key points) of the two-point clouds [20]–[30]. The typical procedure is to first extract key points and compute their descriptors and then establish sparse correspondences between the key points based on the descriptors. After that, various

strategies have been developed to eliminate false correspondences. Commonly used techniques include geometric hashing [31] and random sample consensus (RANSAC) [32], [33]. Other schemes are also developed for obtaining good correspondences. For example, Gelfand *et al.* [21] exploit a branch-and-bound algorithm to find the optimal set of correspondences. Based on the fact that certain ratios defined on a planar congruent set remain invariant under rigid transformations, Mellado *et al.* [24] proposes to extract all sets of coplanar 4-points to register point clouds with certain levels of noises and outliers. With initial correspondences computed using the fast point feature histogram (FPFH) feature [26], Zhou *et al.* [34] propose an optimization framework that simultaneously suppresses spurious correspondences. These methods demonstrated satisfactory performance on point clouds of general surfaces. However, they require sufficient overlap and are usually slow in processing large point clouds (e.g., scans of buildings).

### B. Global Feature-Based Registration

Compared to local features, global features cover larger scales of the point clouds and thus are more descriptive. The most widely used global feature is the plane feature that can be reliably and efficiently extracted from point clouds, especially for man-made scenes. These methods first segment the point clouds into planar patches and then search for correspondences at the patch level using various strategies [10]–[13], [35].

Similar to local descriptors, various global shape descriptors have been developed for point cloud registration, such as the Hough transform descriptor, the spherical entropy image [36], and the viewpoint descriptor [14]. By considering the layout of indoor scenes, Lee *et al.* [37] propose to jointly estimate the layout and registration for indoor scene reconstruction.

Even higher-level features have also been studied in point cloud registration. Thapa *et al.* [38] propose a semantic feature-based method for registration of building scans. Their method starts with a semantic segmentation (achieved by using simple heuristics) of building scans. Then, correspondences are obtained by matching segments of the same semantic type and same pattern (topological relation with other features). Due to the difficulties in semantic segmentation, it remains unclear how to extend this method to register scans of general scenes/objects.

### C. Registration Without Overlap

When overlap between scans is low, registration algorithms seek help from additional information provided by the point clouds [4], [39], [40]. Yan *et al.* [4] propose to register building scans without overlap. The inputs to their system are scans capturing multiple rooms of a large building and/or scans capturing both the interior and exterior of a building. In their problem setting, the overlap between scans becomes extremely small or sometimes does not exist. The authors rely on portals (e.g., windows and doors) extracted from the point clouds to establish potential correspondences between scans. The global registration of the scans is then obtained by selecting a valid set of correspondences via a combinatorial optimization.



In recent years, researchers have also studied the problems of registering/assembling object pieces [41]–[44]. Huang *et al.* [43] assemble fractured object pieces based on roughness analysis and patch-based features defined on fractured surfaces. Then, object pieces are registered by pairwise matching validated via penetration and consistency checks. Based on the fact that certain objects demonstrate continuous sharp feature curves, Huang *et al.* [44] align distinct object parts by enforcing the continuity of the sharp feature curves. This method relies heavily on the rich geometric features of the objects. Thus, it may not be scaled to practical scans of general scenes.

Our approach falls in global feature-based registration. We aim at registering featureless scans of real-world scenes that demonstrate unpredictable levels of overlap. We introduce a novel global descriptor that captures high-level structure information (i.e., the interrelationship of the major planes) of the scenes for registration.

### III. OUR APPROACH

Our registration method is based on the traditional hypothesize-and-evaluate strategy. Specifically, the hypothesis is obtained by matching our novel plane-/line-based descriptor, followed by removing the redundant matchings in the transformation space. Finally, the optimal registration is identified by evaluating the matching scores of the candidate registrations. In the following, we first describe our plane-based descriptor. We then detail our registration algorithm.

#### A. Structure-Level Descriptor

A large portion of the traditional point cloud registration methods look into salient features and rely on a local geometric descriptor to establish correspondences between point clouds. Since sufficient overlaps and descriptive features may not be guaranteed, we rely on a high-level representation of the scene to achieve robust registration. We observe that man-made environments typically consist of planar structures; thus, we represent the main structure of the scene as a collection of planes. These planar structures along with their interrelations reveal high-level global characteristics of the scene, providing promising information for registration [9]. Specifically, we propose a structure-level descriptor defined on planes and lines extracted from the point clouds, from which a unique rigid registration transformation can be established between two descriptors.

1) *Plane/Line Extraction*: There exist a few approaches to extract basic geometric primitives (e.g., planes and lines) from point clouds [45]–[47]. As has been demonstrated that the RANSAC-based plane detection method is robust to noise and outliers and has been successfully applied to other tasks [48], we choose to utilize an efficient implementation of the RANSAC algorithm by Schnabel *et al.* [45] to extract planar segments from the point clouds. Fig. 1(b) shows an example of the extracted planar segments.

Given the planar segments, we then extract lines for each planar segment. Specifically, we first detect boundary points by looking into the distribution of the planar points within their

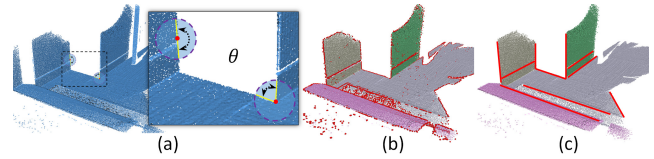


Fig. 1. Line extraction. (a) Input point cloud. (b) Extracted planar segments (in different colors) and boundary points (in red). (c) Extracted line segments (in red).

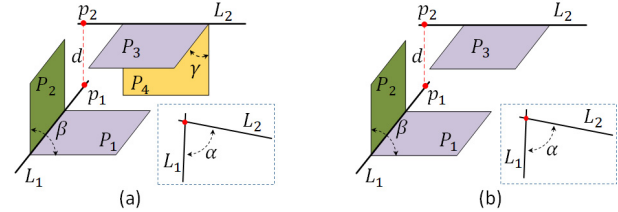


Fig. 2. Structure-level descriptor. (a) Descriptor defined on two pairs of planes. Line  $L_1$  is the intersection of planes  $P_1$  and  $P_2$ ;  $L_2$  is the intersection of planes  $P_3$  and  $P_4$ . (b) Descriptor defined on a line segment  $L_2$  and two planes  $P_1$  and  $P_2$ .

neighborhood. We use an angle criterion to determine if a point is lying on the boundary of a planar segment. Fig. 1(a) (inset) illustrates our angle criterion (we choose the angle threshold  $\theta$  to be  $\pi/2$ ). Similar to plane extraction, we use a RANSAC strategy to extract line segments from the boundary points [see Fig. 1(c)]. Alternative methods, such as [49], can also be applied to extract the lines segments directly from point clouds.

2) *Defining the Descriptor*: Given a certain amount of planes abstracting the main structure of the scene, at least three nonparallel planes are required to establish rigid transformations between two point clouds. To avoid ambiguities (i.e., a corner of three planes can be matched to multiple similar corners) and obtain a unique transformation, we look into quadruplets of nonparallel planes.

We first compute pairwise intersections of the supporting planes of the extracted planar segments, resulting in a set of lines. To cope with near coplanar planes, we discard a line  $L_i$  if  $\text{dist}(L_i, c) > r$ , where  $c$  and  $r$  denote the center and radius of the bounding sphere of the point cloud. Fig. 2(a) illustrates the primitives (i.e., four planes) on which our plane-based descriptor is defined. Specifically, the plane-based descriptor is an 8-D vector consisting of the following entries.

- 1)  $d$ : The distance between the two lines  $L_1$  and  $L_2$ .
- 2)  $\angle(L_1, L_2)$ : The angle between  $L_1$  and  $L_2$ .
- 3)  $\angle(P_1, P_2)$  and  $\angle(P_3, P_4)$ : Angles introduced by the two pairs of planes.
- 4)  $\angle(L_1, P_3)$ ,  $\angle(L_1, P_4)$ ,  $\angle(L_2, P_1)$ , and  $\angle(L_2, P_2)$ : The angles between the intersecting lines of two planes and the other planes.

Note that we choose the acute angle for each pair of primitives. To ensure descriptiveness, our plane-based registration descriptor is defined depending on the relative magnitudes of

the angles between primitives

$$\mathbf{d}^8 = \begin{bmatrix} \text{dist}(L_1, L_2) \\ \angle(L_1, L_2) \\ \angle(P_1, P_2) \\ \angle(P_3, P_4) \\ \min(\angle(L_1, P_3), \angle(L_1, P_4)) \\ \max(\angle(L_1, P_3), \angle(L_1, P_4)) \\ \min(\angle(L_2, P_1), \angle(L_2, P_2)) \\ \max(\angle(L_2, P_1), \angle(L_2, P_2)) \end{bmatrix} \quad (1)$$

if  $\angle(P_1, P_2) < \angle(P_3, P_4)$ . Otherwise, we change the order of the planes and then define the descriptor. Here,  $\min(*, *)$  and  $\max(*, *)$  indicate the smaller and greater value of two angles, respectively.

The above-mentioned plane-based registration descriptor is defined purely on two pairs of nonparallel planes, with which a unique rigid registration transformation can be established between two descriptors. In the very unlikely cases (in particular when the overlap between the point cloud pair is small), less than two pairs of nonparallel planes can be found. The point cloud shown in Fig. 1 is such an example, where only two parallel horizontal planes and two parallel vertical planes are extracted. Thus, no quadruplet of nonparallel planes exists to uniquely define a rigid transformation. In such a case, we seek help from additional line features of the scene. Thus, in addition to the 8-D plane-based descriptor, we also define another type of registration descriptor on a smaller number of geometric primitives, i.e., a pair of nonparallel planes and a line segment. Similarly, the plane-/line-based descriptor is a 6-D vector defined as

$$\mathbf{d}^6 = \begin{bmatrix} \text{dist}(L_1, L_2) \\ \angle(L_1, L_2) \\ \angle(P_1, P_2) \\ \angle(L_1, P_3) \\ \min(\angle(L_2, P_1), \angle(L_2, P_2)) \\ \max(\angle(L_2, P_1), \angle(L_2, P_2)) \end{bmatrix}. \quad (2)$$

### B. Registration

With the structure-level registration descriptor, we are now able to compute transformations between two point clouds. Since our descriptor characterizes the interrelation between nonparallel planes/lines, we can establish a unique rigid transformation using a descriptor  $\mathbf{d}_{\mathcal{L}}$  from a point cloud  $\mathcal{L}$  and its best-matched descriptor  $\mathbf{d}_{\mathcal{G}}$  from the other point cloud  $\mathcal{G}$ .

We enumerate all plane/line combinations to collect a set of descriptors in both  $\mathcal{L}$  and  $\mathcal{G}$ , namely,  $\mathbf{D}_{\mathcal{G}} = \mathbf{D}_{\mathcal{G}}^8 \cup \mathbf{D}_{\mathcal{G}}^6$  and  $\mathbf{D}_{\mathcal{L}} = \mathbf{D}_{\mathcal{L}}^8 \cup \mathbf{D}_{\mathcal{L}}^6$ , where  $\mathbf{D}_{\mathcal{L}}^8 = \{\mathbf{d}_{\mathcal{L}}^8\}$  and  $\mathbf{D}_{\mathcal{L}}^6 = \{\mathbf{d}_{\mathcal{L}}^6\}$  denote the 8-D and 6-D descriptors, respectively.

1) *Descriptor Matching*: To efficiently find the best matches of descriptor pairs, we build a KD-tree for the descriptors  $\mathbf{D}_{\mathcal{G}}$  and we query the most similar descriptor for each descriptor in  $\mathbf{D}_{\mathcal{L}}$ . The distance of a descriptor pair is computed as the Euclidean distance of the two descriptors. To compare a 6-D descriptor against an 8-D descriptor, we simply exclude the two extra dimensions from the 8-D descriptor vector. By doing this, the 8-D descriptor vector is degraded to 6-D. Thus,

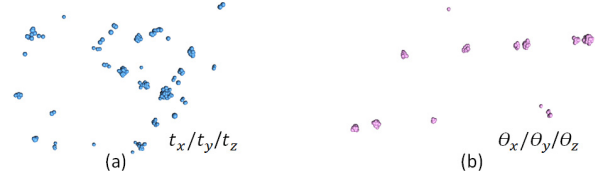


Fig. 3. Visualization of the computed translations and rotations from the best-matched descriptor pairs of two point clouds. Each point in (a) represents a translation and each point in (b) represents a rotation (denoted by the three angles w.r.t. the axes). Note that minor jittering has been added to reveal the duplicated transformations.

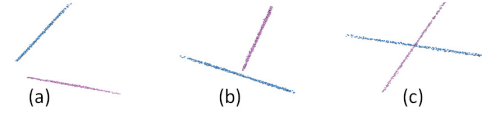


Fig. 4. Penetration tests for two planar segments. (a) and (b) do not have penetration. (c) Example of penetration.

the Euclidean distance between them can be computed using the corresponding entries.

2) *Transformation Redundancy*: Since our registration descriptor mainly encodes the geometric information of the planes, simply enumerating all combinations of the planes results in duplicated transformations. Fig. 3 visualizes the computed translations and rotations from the best-matched descriptor pairs of two point clouds. We can see that a large portion of the transformations is duplicated. This can be observed from a large number of points but fewer clusters in the visualization.

Given a large number of transformations computed from the best-matched descriptor pairs, our final goal is to choose the best transformation that can register the two point clouds. To achieve this goal, we have to evaluate the confidence of each transformation. Here, the confidence of the transformation is typically measured by the number of matched points. Precisely measuring the number of matched points requires querying the nearest neighbor for every point in one point cloud. Performing such queries on small numbers of transformations is affordable. However, the large portion of duplicated transformations hinders us from efficiently obtaining the optimal transformation. To this end, we first remove the redundancy in the transformations and we keep only the most representative ones. Using a KD-tree structure, we search for the neighbors  $\mathbf{T}_i$  of each transformation  $\mathbf{t}_i$  within a radius  $r$ . We simply replace  $\mathbf{t}_i \cup \mathbf{T}_i$  with their mass center. In our implementation, we chose  $r_t = 0.001 \cdot r_{\mathcal{L}}$  for translations and  $r_r = 2^\circ$  for rotations, where  $r_{\mathcal{L}}$  denotes the radius of the bounding sphere of the point cloud  $\mathcal{L}$ . After the redundancy being removed, the number of transformations is significantly reduced. Then, we perform penetration tests to further reduce infeasible transformations. To do so, we look into the point distribution of two planar segments (see Fig. 4). Penetration is considered occurring only if the points of each planar segment lie on both sides of the supporting plane of the other planar segment.

3) *Identifying Optimal Registration*: Intuitively, the optimal registration transforms the point cloud  $\mathcal{L}$  in a way such that

the most number of points can be matched to the points in the point cloud  $\mathcal{G}$ . This is true for most cases, especially for objects with curved surfaces. However, when dealing with man-made scenes that typically comprise planar regions, the transformation receiving most matched points does not always suggest the optimal registration. This is obvious because one planar segment (a set of points lying in a plane) can be matched with any other planar segments. In this work, we measure the confidence of a registration transformation (i.e., a translation denoted by  $\mathbf{t}$  and a rotation denoted by  $\mathbf{r}$ ) by combining two criteria

$$\text{conf}(\mathbf{t}, \mathbf{r}) = w_{\text{plane}} \cdot R_{\text{plane}} + w_{\text{points}} \cdot R_{\text{points}} \quad (3)$$

where  $R_{\text{plane}}$  and  $R_{\text{points}}$  denote the ratio of the matched planes and the ratio of the matched points, respectively. The two weights  $w_{\text{plane}}$  and  $w_{\text{points}}$  are empirically chosen to be 0.2 and 0.8, respectively. By computing the registration confidence for each transformation, the one with the highest registration confidence is considered as the optimal coarse registration.

#### IV. BENCHMARK DATA SET

To evaluate our method and, more importantly, to provide a more practical benchmark data set complementing existing data sets, we create a new data set RESSO<sup>1</sup> targeting both indoor and outdoor scenes.

##### A. Data Collection

Our data acquisition involves two different types of commercial scanners: a high-range static laser scanner (Leica ScanStation C10, with an effective operating range of 100 m) and a handheld scanner (FARO Freestyle X, operating range 3 m). These two scanners have significantly different operating ranges, accuracy, and resolutions, posing sufficient challenges to registration algorithms.

We scanned 187 point clouds in total for 15 different environments (ten indoor scenes and five outdoor scenes). Each indoor scene is captured by a few global point clouds using the static laser scanner and optionally multiple local point clouds using the handheld scanner. The global point clouds capture the majority of each indoor scene, and the local point clouds are intended to capture local regions of the scene, especially the regions that are occluded in the global point clouds. This further adds to the challenges for registration. Due to larger sizes, the outdoor scenes are mainly captured using the long-range static laser scanner.

We scanned the scenes without adding additional clutter for augmenting naturally occurring features, and we tried to create some overlap, but not an excessive amount. Also, the fact that the point clouds stem from different scanners is a possible challenge for some feature extractors.

##### B. Overlap Between Scans

Real-world scans typically have unpredictable varying overlap ratios, which is challenging to registration algorithms. We choose to quantify the overlap of two-point clouds by

measuring the percentage of points that have the closest corresponding point (in another scan of the pair) closer than a threshold  $\epsilon$ . Considering noises in the input point clouds and the unavoidable errors in the registration, we compute the  $\epsilon$ -overlap for each scan pair at a discrete set of  $\epsilon$  values. We depict these discrete  $\epsilon$ -overlap values in a curve, so as to intuitively reveal the overlap between scans. Fig. 5 demonstrates the  $\epsilon$ -overlap curves for a few point cloud pairs from RESSO and other data sets. From the  $\epsilon$  curves, we can see that RESSO has less but a wider range of overlap. Thus, our new data set is a more challenging and useful complement to existing data sets.

##### C. Ground-Truth Registration

Given the challenges in the registration problem itself and a large number of scans, we obtain ground-truth registrations using a combination of automatic approaches and manual registration. Specifically, we run our registration algorithm on the point clouds of each scene, and we record the transformation matrices of the successfully registered point clouds by visual inspection and fine tuning of the registration using ICP [15]. For those failed to be registered in the automatic phase, we manually registered them as initialization to ICP.

#### V. RESULTS AND DISCUSSION

We implemented our method in C++ using the Point Cloud Library [51]. In our current implementation, we mainly focus on local registration (i.e., pairs of the scans), leaving global registration (i.e., simultaneously registering all scans in a scene) as future work. Experiments on various data sets demonstrated that our method significantly outperforms state-of-the-art registration techniques.

##### A. Evaluation Method

Our work focuses on coarse registration, but, in practice, fine registration might be used as a postprocess. One possible evaluation method would be to evaluate the combination of coarse and fine registration algorithms. We opt for a more direct evaluation, where we separately evaluate the impact of coarse registration and fine registration results by comparing the transformed scans to their ground truth. While there are many fine registration methods, we use ICP [15] as a popular representative. Specifically, we consider a registration successful if the registered scan is close enough to the ground truth, that is

$$\text{dist}(\mathbf{s}_r, \mathbf{s}_g) > d_t \quad (4)$$

where  $\text{dist}(\mathbf{s}_r, \mathbf{s}_g)$  measures the average point distance between a registered scan  $\mathbf{s}_r$  and the ground-truth  $\mathbf{s}_g$ . To choose an appropriate value for the threshold  $d_t$ , we take into consideration that our coarse registration result is provided as initialization to a fine registration method. We conducted multiple experiments, and we present the one on all the point cloud pairs in Fig. 9. We introduced a sequence (i.e., ten) of random perturbation transformations (starting from the ground-truth transformation) such that the mean distance of all

<sup>1</sup>RESSO: Real-world Scans with Small Overlap.



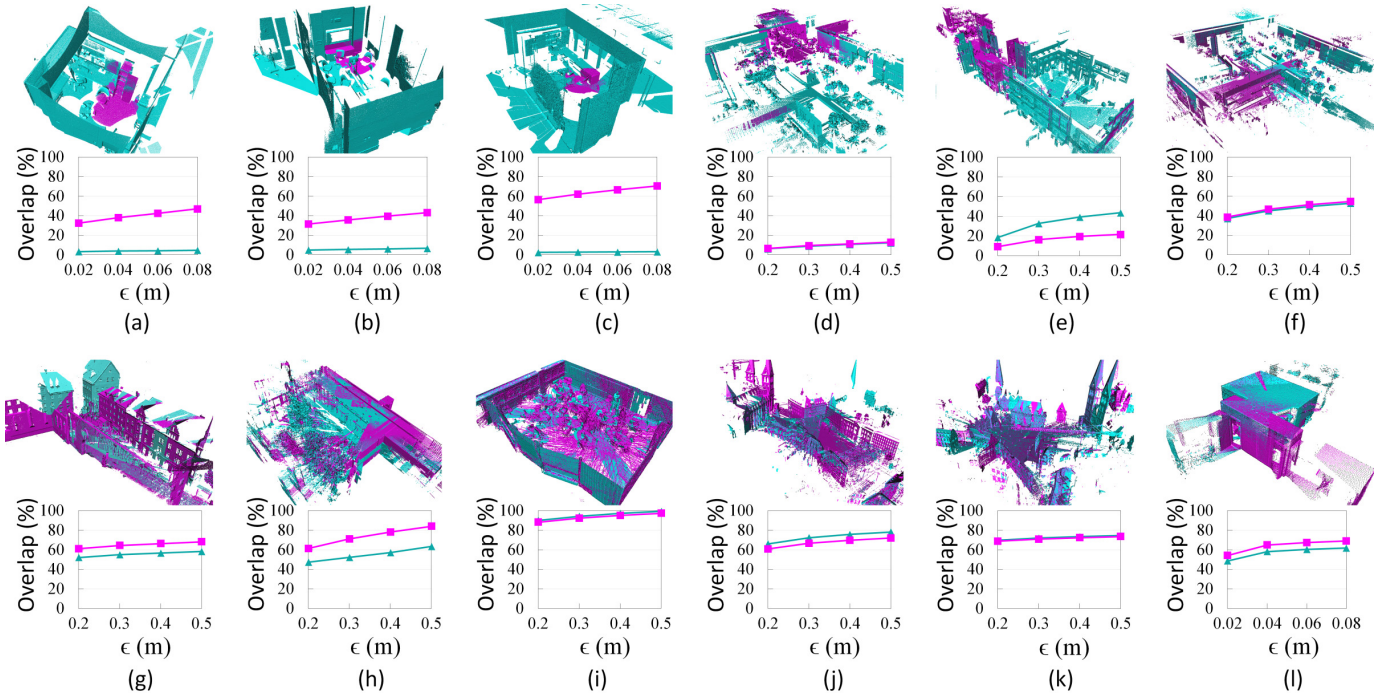


Fig. 5. Overlap of point cloud pairs. (Top Row) Scan pairs from RESSO. (Bottom Row) Scan pairs from existing data sets, i.e., (g) ETH [50], (h) DS2-L [13], (i) TLS-ZEB [14], (j) and (k) Robotic 3-D scan repository [5], and (l) DS1-H [6]. The corresponding  $\epsilon$ -overlap curve is shown below each scan pair.

TABLE I  
SUCCESS RATE (%) OF ICP [15] ON A SEQUENCE OF POINT CLOUD PAIRS WITH INCREASING PERTURBATION LEVELS

| <div>Figure</div> <div>Perturbation (cm)</div> | 5   | 10  | 15  | 20  | 25  | 30  | 35 | 40 | 45 |
|--|-----|-----|-----|-----|-----|-----|----|----|----|
| Figure 9 (a)                                   | 100 | 100 | 100 | 100 | 40  | 0   | 0  | 0  | 0  |
| Figure 9 (b)                                   | 100 | 100 | 100 | 100 | 70  | 0   | 0  | 0  | 0  |
| Figure 9 (c)                                   | 100 | 100 | 100 | 100 | 100 | 30  | 0  | 0  | 0  |
| Figure 9 (d)                                   | 100 | 100 | 100 | 100 | 100 | 100 | 90 | 30 | 0  |

the corresponding points was increased at a constant interval of 5 cm. Then, we ran the ICP algorithm of [15] on all the point cloud pairs in each sequence to test if ICP could converge. We recorded the success rate for each sequence, and the result is demonstrated in Table I. This experiment showed that ICP converged when the mean distance was smaller than 20 cm for indoor scenes and 25 cm for outdoor scenes. Based on these experiments, we conservatively set  $d_t$  to 10 cm for indoor scenes and 20 cm for outdoor scenes.

### B. Registration Results

Figs. 6 and 7 visualize the registration results of the proposed method on ten indoor scenes and five outdoor scenes from our data set RESSO, respectively. Due to the descriptive plane-based descriptor, our registration method managed to register all these scan pairs. Though the indoor scene in Fig. 6(j) and the outdoor scene in Fig. 7(d) partially consist of curved surfaces, planar structures still dominate and our method successfully registered these point clouds. The outdoor scene shown in Fig. 7(a) contains many trees. The planar regions still provide sufficient information for a reliable registration. Besides RESSO, we also tested our registration

method on point clouds from publicly available data sets and related works. The visual results are shown in Fig. 8.

Our method is capable of registering scans with a small overlap. Figs. 6–8 show the registration results of all point clouds for each scene; thus, it is difficult to observe the overlaps between scans. In Fig. 9, we demonstrate a few pairs of scans from our results shown in Figs. 6 and 7.

### C. Initialization to Fine Registration

To test if our coarse registration results can be further improved by a fine registration method, we ran the ICP algorithm of [15] on all the point cloud pairs shown in Fig. 9 and recorded the registration error before and after the ICP step. The result is reported in Table II. We can see that the ICP step significantly reduced the registration error compared to that of the coarse registration, indicating that our coarse registration results provided good initialization to the ICP algorithm.

### D. Robustness to Plane Detection

Our plane-based descriptor is designed to capture the global structure of a scene, allowing us to reliably establish

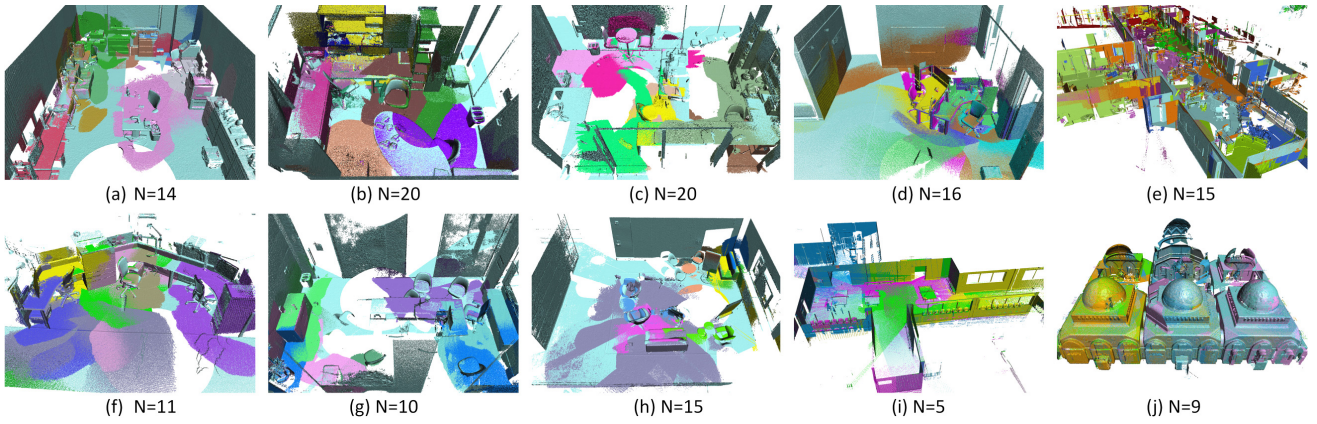


Fig. 6. Registration results of the indoor scenes from RESSO. The ceilings have been removed to better reveal the building interiors. The number below each subfigure indicates the total scans in each scene. (a)  $N = 14$ . (b)  $N = 20$ . (c)  $N = 20$ . (d)  $N = 16$ . (e)  $N = 15$ . (f)  $N = 11$ . (g)  $N = 10$ . (h)  $N = 15$ . (i)  $N = 5$ . (j)  $N = 9$ .

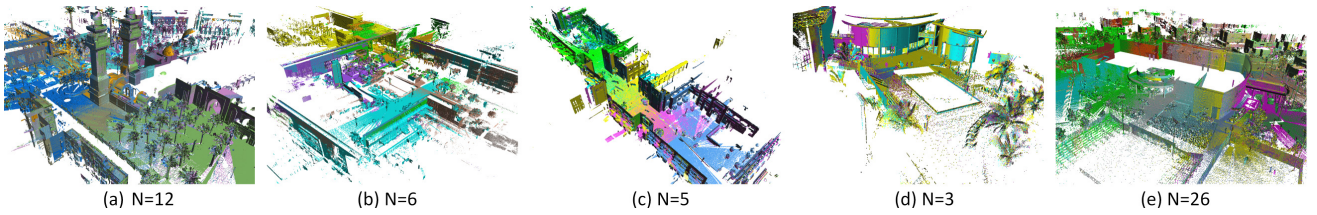


Fig. 7. Registration results of the outdoor scenes from RESSO. The number below each subfigure indicates the total scans in each scene. (a)  $N = 12$ . (b)  $N = 6$ . (c)  $N = 5$ . (d)  $N = 3$ . (e)  $N = 26$ .

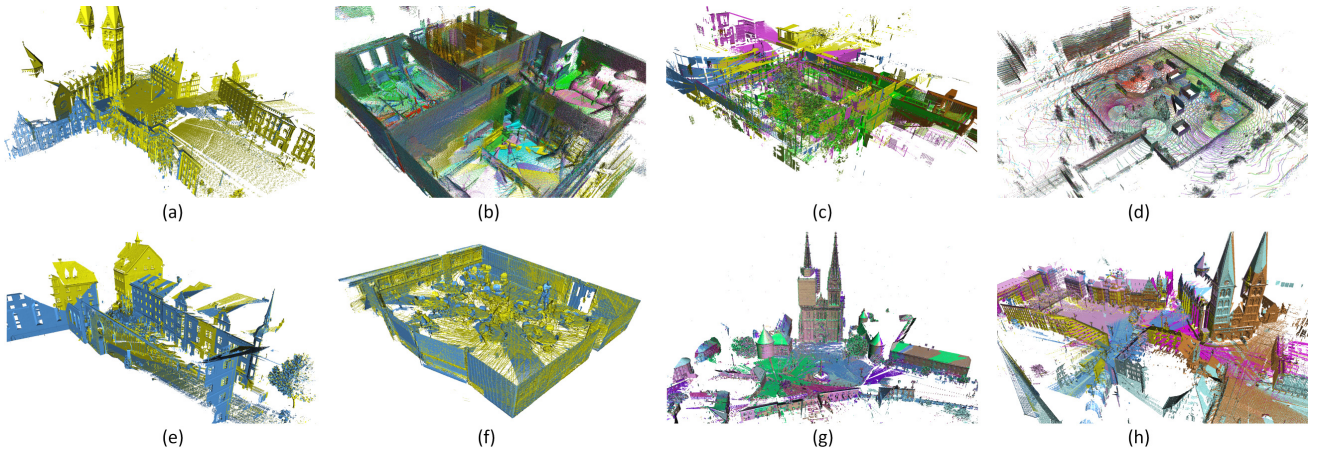


Fig. 8. Registration results of our method on various data sets. (a) Bremen [52]. (b) DS1-H [6]. (c) DS2-L [13]. (d) DS3-V [13]. (e) ETH [50]. (f) TLS-ZEB [14]. (g) and (h) Robotic 3-D scan repository [5]. The ceilings in (b), (c), and (f) have been removed to better reveal the building interiors.

TABLE II  
REGISTRATION ERRORS BEFORE AND AFTER APPLYING THE FINE  
REGISTRATION METHOD OF [15] ON THE CLOUD  
PAIRS SHOWN IN FIG. 9

| Figure       | Errors (m) | Coarse | Fine  |
|--------------|------------|--------|-------|
| Figure 9 (a) |            | 0.026  | 0.008 |
| Figure 9 (b) |            | 0.025  | 0.006 |
| Figure 9 (c) |            | 0.154  | 0.039 |
| Figure 9 (d) |            | 0.136  | 0.042 |

structure-level correspondences between two point clouds. Since a few descriptive planes are adequate in depicting the main structure of the scene, it is not necessary (nor possible) to obtain a complete set of planes accurately extracted from the

point clouds. To evaluate this, we repeatedly ran our method on the scene shown in Fig. 6(a) by incrementally removing planes. Specifically, we remove 10% of the extracted planes at each iteration until our algorithm breaks down. Fig. 10 reports how our method behaves by gradually dropping planes. Such a test confirms that a few dominant planes can provide adequate information for point cloud registration, allowing our method to achieve satisfactory registration results as long as certain descriptive planes (i.e., a small portion of planes) are present.

#### E. Robustness to Noise

In order to evaluate the impact of noisy surfaces, we added the Gaussian noise to a pair of point clouds from Fig. 6(a)



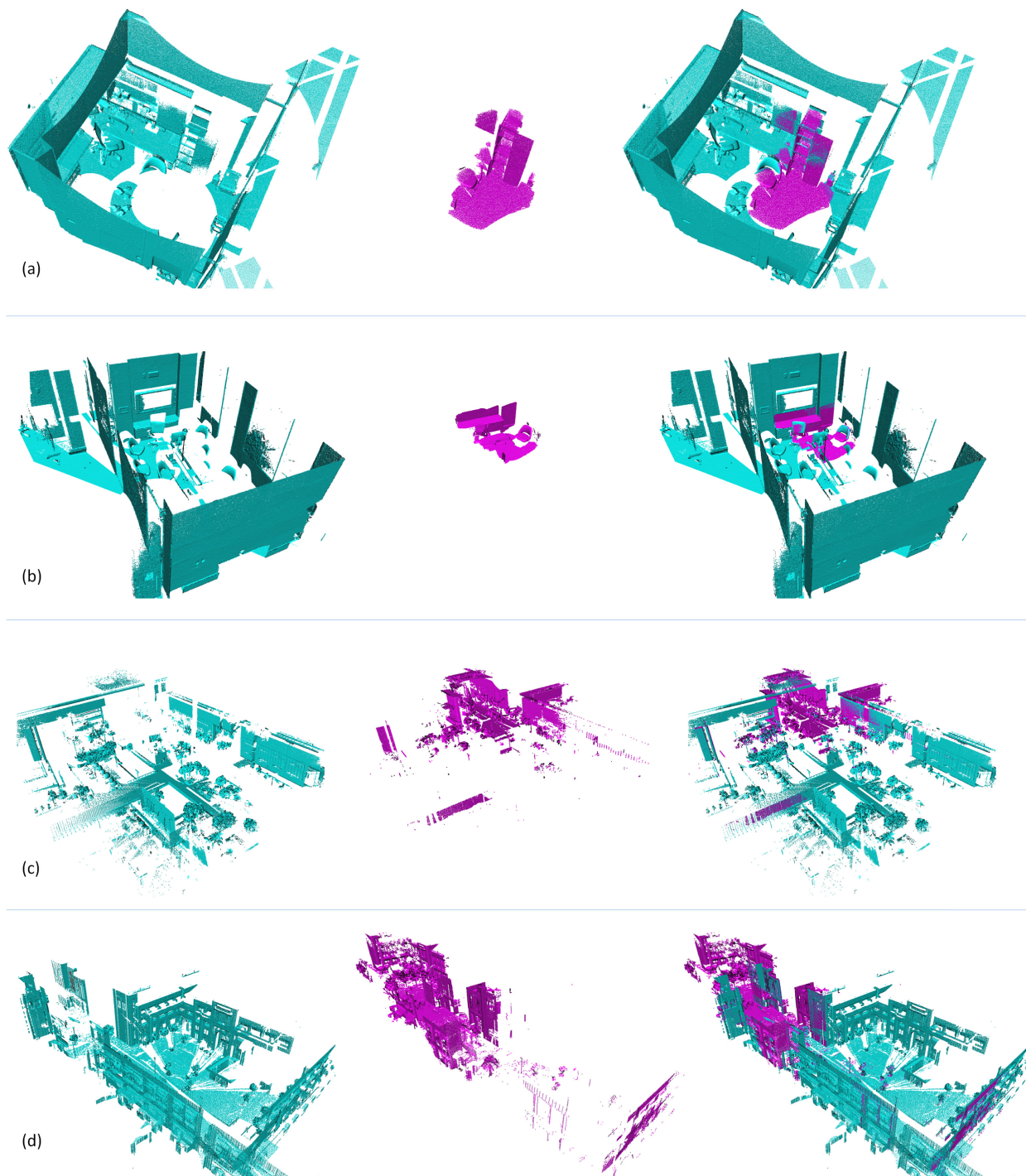


Fig. 9. Pairs of point clouds registered by our method, intended to reveal the overlaps between scans. (First Two Columns) Input scan pairs. (Right Column) Registration results.

with increasing noise levels, i.e., standard deviations ( $\sigma$ ) 15, 30, 45, and 60 cm, respectively. Though the noise levels are quite high, we were still able to extract planes with sufficient quality at three noise levels. Fig. 11 shows the registration result at noise level  $\sigma = 45$  cm. However, when the noise level reached 60 cm, where the smaller point cloud (in green) were completely contaminated by the noise (note how difficult to recognize the chairs in the scene), our RANSAC-based plane extraction algorithm failed to detect sufficient planes to

establish reliable correspondences for the registration. Such a test indicates that our method is robust to noise as long as the major representative planes can be extracted.

#### F. Comparison

We compared our method against various point cloud registration methods, including local descriptor-based approaches and plane-based approaches. Tables III and IV report the performance of our method and the competing methods on



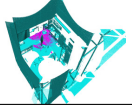
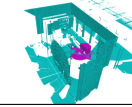
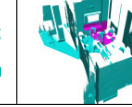
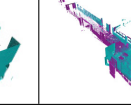
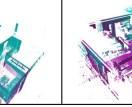
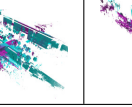
TABLE III

COMPARISON WITH A FEW LOCAL DESCRIPTOR-BASED METHODS AND SUPER4PCS [24]. THE PERFORMANCE HERE IS MEASURED BY THE PERCENTAGE (%) OF THE SCANS THAT WERE SUCCESSFULLY REGISTERED IN EACH SCENE

| Figure<br>Method | 6 (a) | 6 (b) | 6 (c) | 6 (d) | 6 (f) | 6 (g) | 6 (h) | 6 (j) | 7 (a) | 7 (b) | 7 (c) |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Super4PCS [24]   | 0     | 0     | 5     | 0     | 0     | 0     | 7     | 13    | 9     | 33    | 20    |
| FPPH [25]        | 15    | 11    | 5     | 7     | 0     | 11    | 14    | 33    | 42    | 17    | 40    |
| FastGlobal [34]  | 31    | 21    | 21    | 47    | 20    | 22    | 7     | 33    | 92    | 50    | 100   |
| CZK [53]         | 46    | 37    | 21    | 53    | 50    | 22    | 50    | 22    | 75    | 33    | 100   |
| RANSAC-based     | 15    | 16    | 16    | 13    | 50    | 67    | 43    | 0     | 43    | 33    | 100   |
| Ours             | 77    | 53    | 37    | 67    | 70    | 78    | 64    | 50    | 100   | 100   | 100   |

TABLE IV

COMPARISON WITH A FEW PLANE-BASED REGISTRATION METHODS ON RESSO. THE RATE IS MEASURED BY THE PERCENTAGE (%) OF THE TOTAL REGISTRATIONS EACH METHOD SUCCEEDED

| Method       | Data |  |  |  |  |  |  | Rate (%) |
|--------------|------|---|---|---|---|--|---|----------|
| VBPC [12]    |      |   |   |   | ✓   | ✓  | ✓   | 50       |
| SSFR [13]    |      | ✓   |   | ✓   | ✓   | ✓  | ✓   | 83       |
| RANSAC-based |      | ✓   |   |   | ✓   |  | ✓   | 50       |
| Ours         |      | ✓   | ✓   | ✓   | ✓   | ✓  | ✓   | 100      |

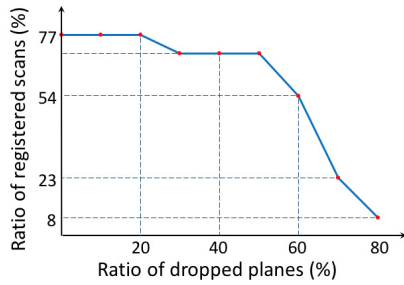


Fig. 10. Registration by gradually dropping planes on the scene shown in Fig. 6(a). Planes with smaller numbers of points are dropped first.

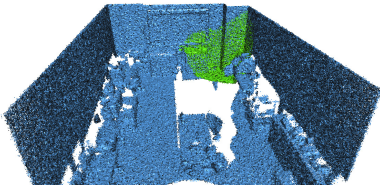


Fig. 11. Registration of two point clouds with Gaussian noise (standard deviation  $\sigma = 45$  cm).

some of the scan pairs from scenes shown in Figs. 6 and 7. The performance is measured in terms of the percentage of successfully registered point clouds. From Table III, we can see that Super4PCS [24] failed in registering most of the point cloud pairs from the indoor scenes. Other local descriptor-based registration methods managed to register only a small portion of the scans. Such poor performance is mainly due to the small overlap and the absence of local geometric features in the point clouds. As expected, the performance of these techniques improves when the scans have significantly larger overlap, e.g., the indoor scene shown in Fig. 6(j) and the outdoor scenes shown in Fig. 7. Besides, the scenes in Figs. 6(j) and 7(a) contain some curved structures, adding descriptive local geometric

features for registration. The large overlapping ratio and the geometric features bring the performance improvements.

We also compared our method against various state-of-the-art plane-based registration methods. Due to that source code of the competing methods is not available, we ask the authors to run their algorithms on a few scan pairs randomly chosen from RESSO. These scan pairs demonstrate a wide range of overlapping ratios. Table IV summarizes the comparison. Among these methods, the “RANSAC-based” approach is quite similar to our method, except that we replace our descriptor-based correspondence search with a RANSAC-based correspondence search. From all these comparisons, we can conclude that planes are effective features for registering scans of real-world scenes. Based on the novel plane based, our method significantly outperforms the competing methods in terms of the percentage of successfully registered scans.

### G. Running Times

Table V gives the running times of our method and the competing methods on the scenes shown in Figs. 6 and 7. The Super4PCS algorithm [24] requires to explore sufficient sets of coplanar 4-points, thus becomes more expensive for scans of large scenes. The method by Zhou *et al.* [34] demonstrates higher efficiency than the Super4PCS technique because its optimization process involves neither correspondence updates nor closest-point queries. Compared to these techniques, our method takes advantage of the plane-based descriptor so that structure-level correspondences between scans can be very efficiently established via nearest neighbor search. Thus, it has better efficiency than most of the competing methods, in particular for larger scenes. Note that the input point clouds were downsampled to enable the competing algorithms to generate their results within an acceptable time frame.

TABLE V

EXECUTION TIMES (IN SECONDS) OF OUR METHOD AND SOME COMPETING METHODS ON THE SCENES SHOWN IN FIGS. 6 AND 7. THE EXECUTION TIMES WERE MEASURED ON A LAPTOP WITH A DUAL-CORE 2.4 GHz INTEL CORE I3-4000M CPU

| Figure<br>Method | 6 (a) | 6 (b) | 6 (c) | 6 (d) | 6 (f) | 6 (g) | 6 (h) | 6 (j) | 7 (a) | 7 (b) | 7 (c) |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Super4PCS [24]   | 202   | 305   | 514   | 349   | 631   | 434   | 534   | 527   | 479   | 521   | 759   |
| FPFH [25]        | 198   | 20    | 61    | 130   | 107   | 51    | 79    | 84    | 477   | 260   | 270   |
| FastGlobal [34]  | 10    | 2     | 3     | 7     | 11    | 4     | 7     | 21    | 119   | 59    | 31    |
| CZK [53]         | 17    | 7     | 9     | 8     | 16    | 9     | 11    | 59    | 60    | 63    | 57    |
| RANSAC-based     | 504   | 223   | 587   | 542   | 168   | 450   | 98    | 3107  | 2478  | 5439  | 3196  |
| Ours             | 25    | 6     | 8     | 13    | 13    | 9     | 9     | 5     | 21    | 23    | 16    |

## H. Limitations

Our plane-based descriptor is dedicated to registering point clouds of scenes that at least partially consist of planar structures. Thus, the proposed descriptor is especially suitable for registering scans of man-made environments. The descriptor will probably not be successful for scans of vegetation and scans of individual objects that consist of curved surfaces.

Another limitation of our current implementation is that the confidence metric defined in (3) can only handle the majority of the tested point clouds in our benchmark data sets. It still remains a challenge to develop a reliable confidence metric that works for all scenarios.

## VI. CONCLUSION

In this article, we discussed several challenges of the point cloud registration problem. To address these challenges, we presented a simple but effective method for registering practical and feature-poor scans with a small overlap in arbitrary initial poses. Our method is based on a high-level descriptor that reveals structural characteristics of the scenes, leading to superior registration performance.

Despite the excellent performance of the proposed registration algorithm, we demonstrated that the point cloud registration problem is far from being solved, leaving significant room for improvement and future work. We also provide the community a new challenging benchmark data set that is large and challenging enough to ensure that registration algorithms are fairly evaluated and compared, in the hope that experts in related fields seize such research opportunities and push the state of the art in point cloud registration forward.

## ACKNOWLEDGMENT

The authors would like to thank Y. Tian for helping them with data acquisition. They would also like to thank the support of NVIDIA Corporation with the donation of the Titan V GPU used for rendering the point clouds in this research.

## REFERENCES

- [1] W. R. Scott, G. Roth, and J.-F. Rivest, "View planning for automated three-dimensional object reconstruction and inspection," *ACM Comput. Surv.*, vol. 35, no. 1, pp. 64–96, Mar. 2003.
- [2] S. Wu *et al.*, "Quality-driven poisson-guided autoscanning," *ACM Trans. Graph.*, vol. 33, no. 6, p. 203, 2014.
- [3] A. Monszpart, N. Mellado, G. J. Brostow, and N. J. Mitra, "Rapter: Rebuilding man-made scenes with regular arrangements of planes," *ACM Trans. Graph.*, vol. 34, no. 4, p. 103, Aug. 2015.
- [4] F. Yan, L. Nan, and P. Wonka, "Block assembly for global registration of building scans," *ACM Trans. Graph.*, vol. 35, no. 6, p. 237, Nov. 2016.
- [5] A. Nüchter and K. Lingemann, *Robotic 3D Scan Repository*. Accessed: Sep. 27, 2017. [Online]. Available: <https://kos.informatik.uos.de/3Dscans/>
- [6] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, "Challenging data sets for point cloud registration algorithms," *Int. J. Robot. Res.*, vol. 31, no. 14, pp. 1705–1711, Dec. 2012.
- [7] M. Magnusson, N. Vaskevicius, T. Stoyanov, K. Pathak, and A. Birk, "Beyond points: Evaluating recent 3D scan-matching algorithms," in *Proc. ICRA*, May 2015, pp. 3631–3637.
- [8] B. Maiseli, Y. Gu, and H. Gao, "Recent developments and trends in point set registration methods," *J. Vis. Commun. Image Represent.*, vol. 46, pp. 95–106, Jul. 2017.
- [9] C. Brenner, C. Dold, and N. Ripperda, "Coarse orientation of terrestrial laser scans in urban environments," *ISPRS J. Photogram. Remote Sens.*, vol. 63, pp. 4–18, Jan. 2008.
- [10] K. Khoshelham, "Automated localization of a laser scanner in indoor environments using planar objects," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Sep. 2010, pp. 1–7.
- [11] F. Bosché, "Plane-based registration of construction laser scans with 3D/4D building models," *Adv. Eng. Informat.*, vol. 26, no. 1, pp. 90–102, Jan. 2012.
- [12] Y. Xu, R. Boerner, W. Yao, L. Hoegner, and U. Stilla, "Automated coarse registration of point clouds in 3D urban scenes using VOXEL based plane constraint," *ISPRS Ann. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 4, pp. 1–7, Sep. 2017.
- [13] J. Sanchez, F. Denis, P. Checchin, F. Dupont, and L. Trassoudaine, "Global registration of 3D LiDAR point clouds based on scene features: Application to structured environments," *Remote Sens.*, vol. 9, no. 10, p. 1014, 2017.
- [14] W. Forstner and K. Khoshelham, "Efficient and accurate registration of point clouds with plane to plane correspondences," in *Proc. ICCV*, Oct. 2017, pp. 2165–2173.
- [15] P. J. Besl and D. N. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
- [16] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 8, pp. 712–721, Aug. 1999.
- [17] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3D Digit. Imag. Modeling*, Jun. 2001, pp. 145–152.
- [18] H. Li, R. W. Sumner, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," *Comput. Graph. Forum*, vol. 27, no. 5, pp. 1421–1430, Jul. 2008.
- [19] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," *Robot., Sci. Syst.*, vol. 2, no. 4, p. 435, Jun. 2009.
- [20] A. E. Johnson, "Spin-Images: A representation for 3D surface matching," Ph.D. dissertation, Robot. Inst. Carnegie Mellon Univ., Pittsburgh, PA, USA, 1997.
- [21] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust global registration," in *Proc. SGP*, Jul. 2005, p. 5.
- [22] X. Li and I. Guskov, "Multiscale features for approximate alignment of point-based surfaces," in *Proc. SGP*, Jul. 2005, p. 217.
- [23] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.

- [24] N. J. Mitra, N. Mellado, and D. Aiger, "Super 4PCS fast global pointcloud registration via smart indexing," *Comput. Graph. Forum*, vol. 33, no. 5, pp. 205–215, Aug. 2014.
- [25] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Proc. IROS* Sep. 2008, pp. 3384–3391.
- [26] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. ICRA*, May 2009, pp. 3212–3217.
- [27] J. Yang, Z. Cao, and Q. Zhang, "A fast and robust local descriptor for 3D point cloud registration," *Inf. Sci.*, vol. 346, pp. 163–179, Jun. 2016.
- [28] A. Zeng, S. Song, M. Niessner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *Proc. CVPR*, Jul. 2017, pp. 1802–1811.
- [29] X. Huang, "Learning a 3D descriptor for cross-source point cloud registration from synthetic data," 2017, *arXiv:1708.08997*. [Online]. Available: <https://arxiv.org/abs/1708.08997>
- [30] G. Elbaz, T. Avraham, and A. Fischer, "3d point cloud registration for localization using a deep neural network auto-encoder," in *Proc. CVPR*, Jul. 2017, pp. 4631–4640.
- [31] H. J. Wolfson and I. Rigoutsos, "Geometric hashing: An overview," *IEEE Comput. Sci. Eng.*, vol. 4, no. 4, pp. 10–21, Oct. 1997.
- [32] M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [33] C.-S. Chen, Y.-P. Hung, and J.-B. Cheng, "RANSAC-based DARCES: A new approach to fast automatic registration of partially overlapping range images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 11, pp. 1229–1234, Nov. 1999.
- [34] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Proc. ECCV*, 2016, pp. 766–782.
- [35] J. Xiao, B. Adler, J. Zhang, and H. Zhang, "Planar segment based three-dimensional point cloud registration in outdoor environments," *J. Field Robot.*, vol. 30, no. 4, pp. 552–582, Jul./Aug. 2013.
- [36] B. Sun, "3D global shape descriptors applied in scan registration," Ph.D. dissertation, Fac. Math., Inform. Natural Sci., Dept. Inform., Univ. Hamburg, Hamburg, Germany, 2015.
- [37] J.-K. Lee, J.-W. Yea, M.-G. Park, and K.-J. Yoon, "Joint layout estimation and global multi-view registration for indoor reconstruction," in *Proc. ICCV*, Oct. 2017, pp. 162–171.
- [38] A. Thapa, S. Pu, and M. Gerke, "Semantic feature based registration of terrestrial point clouds," in *Proc. Int. Soc. Photogramm. Remote Sens.*, 2009, pp. 1–6.
- [39] A. Cohen, J. L. Schönberger, P. Speciale, T. Sattler, J.-M. Frahm, and M. Pollefeys, "Indoor-outdoor 3D reconstruction alignment," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 285–300.
- [40] T. Koch, M. Körner, and F. Fraundorfer, "Automatic alignment of indoor and outdoor building models using 3D line segments," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 10–18.
- [41] W. Cutler, "The six-piece burr," *J. Recreational Math.*, vol. 10, no. 4, pp. 241–250, 1978.
- [42] D. F. Huber, "Automatic three-dimensional modeling from reality," Ph.D. dissertation, Robot. Inst. Carnegie Mellon Univ., Pittsburgh, PA, USA, 2002.
- [43] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann, "Reassembling fractured objects by geometric matching," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 569–578, Aug. 2006.
- [44] H. Huang, M. Gong, D. Cohen-Or, Y. Ouyang, F. Tan, and H. Zhang, "Field-guided registration for feature-conforming shape composition," *ACM Trans. Graph.*, vol. 31, no. 6, p. 179, Nov. 2012.
- [45] R. Schnabel, R. Wahl, and R. Klein, "Efficient ransac for point-cloud shape detection," *Comput. Graph. Forum*, vol. 26, no. 2, pp. 214–226, 2007.
- [46] R. Toldo and A. Fusiello, "Robust multiple structures estimation with J-Linkage," in *Proc. ECCV*, 2008, pp. 537–547.
- [47] W. S. Grant, R. C. Voorhies, and L. Itti, "Finding planes in LiDAR point clouds for real-time registration," in *Proc. IROS*, Nov. 2013, pp. 4347–4354.
- [48] L. Nan and P. Wonka, "Polyfit: Polygonal surface reconstruction from point clouds," in *Proc. ICCV*, Oct. 2017, pp. 2353–2361.
- [49] Y. Lin, C. Wang, B. Chen, D. Zai, and J. Li, "Facet segmentation-based line segment extraction for large-scale point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 4839–4854, Sep. 2017.
- [50] T. Hackel, J. D. Wegner, and K. Schindler, "Fast semantic segmentation of 3d point clouds with strongly varying density," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 3, no. 3, pp. 177–184, 2016.
- [51] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. ICRA*, May 2011, pp. 1–4.
- [52] D. Bormann, J. Elseberg, and A. Nüchter, "Thermal 3D mapping of building façades," in *Intelligent Autonomous Systems*. Cham, Switzerland: Springer, 2013, pp. 173–182.
- [53] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5556–5565.



**Songlin Chen** received the bachelor's degree from the Shenyang University of Technology, Shenyang, China, in 2013, and the Ph.D. degree from the Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, in 2019.

He is currently an Engineer with Huawei Technologies Company, Ltd, Nanjing, China. His research interests are in the fields of computer vision and deep learning.



**Liangliang Nan** received the bachelor's degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2003, and the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China, in 2009.

He was a Research Scientist with the King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, from 2013 to 2017, and an Assistant Professor and an Associate Professor with SIAT from 2009 to 2013. He is currently an Assistant Professor with the Delft University of Technology (TU Delft), Delft, The Netherlands. His research interests are in the fields of computer graphics, computer vision, and 3-D geoinformation.



**Renbo Xia** received the bachelor's and master's degrees from the Harbin Institute of Technology, Harbin, China, in 2000 and 2002, respectively, and the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China, in 2006.

He is currently a Research Fellow with the Shenyang Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests are in the fields of computer vision and image understanding.



**Jibin Zhao** received the bachelor's degree in mechanical engineering from the Hefei University of Technology, Hefei, China, in 1996, the master's degree from Shandong University, Jinan, China, in 2000, and the Ph.D. degree from the Graduate School, Chinese Academy of Sciences, Beijing, China, in 2004.

He is currently a Researcher with the Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include computer-aided design, rapid prototyping, and reverse engineering.



**Peter Wonka** received the M.Sc. degree in urban planning and the Ph.D. degree in computer science from the Technical University of Vienna, Vienna, Austria.

He was a Post-Doctoral Researcher with the Georgia Institute of Technology, Atlanta, GA, USA, and a Faculty Member with Arizona State University, Tempe, AZ, USA. He is currently a Full Professor of computer science with the King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, where he is also the Associate

Director of the Visual Computing Center (VCC). His research interests include various topics in computer graphics, computer vision, remote sensing, image processing, visualization, machine learning, and data mining.

Dr. Wonka also serves as an Associate Editor for the *ACM Transactions on Graphics*, the *IEEE COMPUTER GRAPHICS AND APPLICATIONS*, and the *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*.