

# Fast Descriptors and Correspondence Propagation for Robust Global Point Cloud Registration

Huan Lei, Guang Jiang, *Member, IEEE*, and Long Quan

**Abstract**—In this paper, we present a robust global approach for point cloud registration from uniformly sampled points. Based on eigenvalues and normals computed from multiple scales, we design fast descriptors to extract local structures of these points. The eigenvalue-based descriptor is effective at finding seed matches with low precision using nearest neighbor search. Generally, recovering the transformation from matches with low precision is rather challenging. Therefore, we introduce a mechanism named correspondence propagation to aggregate each seed match into a set of numerous matches. With these sets of matches, multiple transformations between point clouds are computed. A quality function formulated from distance errors is used to identify the best transformation and fulfill a coarse alignment of the point clouds. Finally, we refine the alignment result with the trimmed iterative closest point algorithm. The proposed approach can be applied to register point clouds with significant or limited overlaps and small or large transformations. More encouragingly, it is rather efficient and very robust to noise. A comparison to traditional descriptor-based methods and other global algorithms demonstrates the fine performance of the proposed approach. We also show its promising application in large-scale reconstruction with the scans of two real scenes. In addition, the proposed approach can be used to register low-resolution point clouds captured by Kinect as well.

**Index Terms**—Point cloud registration, correspondence propagation, match.

## I. INTRODUCTION

THE aim of registration is to find the best transformation that aligns a pair of point clouds in different coordinate system. It is critical to many applications in computer vision, such as 3D modeling [1], localization for robot navigation [2], [3], object recognition and segmentation [4]–[6], etc. However, when the data suffer from significant occlusions, large transformations and noise, the alignment between

Manuscript received September 13, 2016; revised February 3, 2017, March 9, 2017, and April 6, 2017; accepted April 28, 2017. Date of publication May 3, 2017; date of current version May 26, 2017. This work was supported in part by the Research Grants Council General Research Fund under Grant T22-603/15N and Grant 16208614, in part by Hong Kong under Grant ITC PSKL12EG02, in part by the 111 Project of China under Grant B08038, and in part by the Natural Science Foundation of China under Grant 61403292. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhengguo Li. (*Corresponding author: Guang Jiang*)

H. Lei and G. Jiang are with the State Key Laboratory of Integrated Services Networks, School of Telecommunication Engineering, Xidian University, Xi'an 710071, China (e-mail: hlei.ziyan@gmail.com; gjiang@mail.xidian.edu.cn).

L. Quan is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: quan@cse.ust.hk).

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

Digital Object Identifier 10.1109/TIP.2017.2700727

point clouds becomes challenging. In general, the problem of registration can be mathematically expressed as follows. Take two point clouds  $P$ ,  $Q$  as an example, and  $\mathcal{X} \subset P$ ,  $\mathcal{Y} \subset Q$  are the overlapped points between the two point clouds. The relationship between  $\mathcal{X}$  and  $\mathcal{Y}$  can be expressed as follows:

$$\mathbf{y} = \mathbf{Rx} + \mathbf{t}. \quad (1)$$

in which  $\mathbf{x} \in \mathcal{X}$ ,  $\mathbf{y} \in \mathcal{Y}$ .  $\mathbf{R}$  defines the  $3 \times 3$  rotation matrix and  $\mathbf{t}$  the  $3 \times 1$  translation vector. To calculate the parameters  $\mathbf{R}$ ,  $\mathbf{t}$  with high accuracy, many approaches have been developed.

The Iterative Closest Point (ICP) algorithm [7], [8], since introduced, has been popularly adopted to register point clouds due to its simplicity. Yet, it implicitly assumes that the point clouds to be aligned contain full overlap, which is violated mostly in practice. Chetverikov *et al.* introduced the trimmed ICP algorithm [9], which then extends the application of ICP to point clouds with partial overlap. On the other hand, the ICP algorithm is also well-known to be susceptible to the local minima, as a result making it demand good initializations for satisfactory registration results. Among the various approaches in registration, the local descriptor based ones are generally efficient to provide a coarse alignment of the point clouds. Usually, describing keypoints extracted by a keypoint detector (e.g., [10]–[12]) with a local descriptor (e.g., [13], [14]), matches with high-precision can be obtained using nearest neighbour search. With these matches, a coarse transformation matrix can be computed with RANSAC [15] between the point clouds. Yet, when the point clouds are with large transformation, the coarse alignment should be calculated with more efforts, such as using random sampling [13], Hough Transforms [16] or other techniques. Nonetheless, the local descriptors are altogether sensitive to noise and their performance degrades rapidly with the increase of noise [17]. And the detection of keypoints suffers from noise as well [18]. As a result, it is inevitable for the traditional descriptor based methods to return mediocre alignment for point clouds with significant noise.

In this paper, we present a novel approach for robust global point cloud registration, which is most similar to the local descriptor-based methods in essence. However, in contrast, the descriptors we design are much simpler and more efficient to be obtained. Specifically, they are formed with eigenvalues and normals computed from multiple scales. We perform the registration from uniformly sampled points since they effectively reflect the rough structure of a scene or an object. Despite its simplicity, the eigenvalue-based descriptor we design is effective at finding seed matches with low

precision using nearest neighbour search. The correspondence propagation mechanism we introduce is able to aggregate a set of matches with high precision from those seed matches, facilitating the computation of a coarse alignment between the point clouds with RANSAC [15]. Finally, the coarse registration is refined with the trimmed ICP algorithm. The proposed approach can be applied to register point clouds with significant or limited overlaps and small or large transformation. More encouragingly, it is rather efficient and quite robust to noise. With a comparison to traditional descriptor-based methods and other global algorithms, we demonstrate the fine performance of the proposed approach. We also show its promising application in large-scale reconstruction with scans of two real scenes. In addition, the proposed approach can be used to register low-resolution data captured by Kinect as well.

## II. RELATED WORK

We concisely review work on rigid 3D point cloud registration by grouping them into local methods and global methods. The proposed approach in this paper falls into the latter group.

### A. Local Methods

The Iterative Closest Point (ICP) algorithm [7] conducts correspondence search and transformation estimate alternately in the point clouds. Despite its popular use in point cloud registration, the ICP algorithm implicitly assumes that the two point clouds to be aligned contain full overlap, which is mostly inappropriate and limits its applications in practice. Chetverikov *et al.* extended the ICP algorithm to trimmed ICP [9], which makes it applicable to point clouds with partial overlap. However, the ICP algorithm and its variants [9], [19] are altogether vulnerable to local minima. There are approaches seeking for more robust registration based on the Gaussian Mixture Model (GMM), such as GMMReg [20], Kernel Correlation [21], Coherent Point Drift [22]. Jian and Vemuri [20] also showed that by minimizing the KL divergence of two GMMs, ICP should be a special case of GMMReg. The Normal Distributions Transform (NDT) algorithm [23], [24] defines Gaussian Models for each cell in a spatial grid and fulfills the registration with numeric optimization. Most recently, the Support Vector Registration (SVR) algorithm [25] uses a support vector machine to construct the GMM representation. These GMM-based methods, despite more robust than ICP, tend to get trapped in local minima as well.

### B. Global Methods

To address the problem of local minima, many stochastic algorithms are developed, such as particle filtering [26], simulated annealing [1], [27], and genetic algorithms [28], [29]. However, they require good initializations to narrow the parameter space for heuristic search. The feature-based approaches extract local descriptors such as FPFH [13], SHOT [30] from keypoints to obtain matches between the point clouds, and estimate the relative pose with

RANSAC [15], random sampling [13], or other techniques (e.g., [5], [16], [31], [32]). However, a sensitivity to noise makes them fail to work when the point clouds have significant noise. Aiger *et al.* proposed a RANSAC-based algorithm which obtains transformation hypothesis by matching the congruent sets of coplanar four points (4PCS) [33]. Due to the quadratic time complexity of 4PCS, Mellado *et al.* then extended it to Super4PCS and improved the time complexity to be linear [34]. Makadia *et al.* presented to use the global shape descriptor, Extended Gaussian Images (EGI), to align point clouds by maximizing correlation between the descriptors [35]. In contrast, Straub *et al.* utilized the Bayesian Nonparametric Mixtures to describe statistical characteristics of the point clouds, and searched for the optimal transformation in a BnB paradigm [36]. Recently, Yang *et al.* proposed a novel global algorithm named Go-ICP, in which the BnB over translations is nested inside the BnB over rotations and ICP is used internally to improve the BnB bounds [37]. Bustos *et al.* [38] then proposed to speed up the rotation search of Go-ICP with stereographic projections and efficient runtime is reported. Most recently, Campbell and Petersson originally formulated the notable local method GMMReg [20] into the BnB framework, and ensures that its alignment between the point clouds is globally optimal [39].

## III. APPROACH

Following our representations in the introduction, we perform uniform down-sampling to point clouds  $P$  and  $Q$ , and obtain two sparse point clouds  $P_d = \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$  and  $Q_d = \{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ . Fig. 1 shows a framework of the whole algorithm. We introduce the multiscale descriptors in Section III-A, describe the correspondence propagation mechanism in Section III-B, and discuss identification of the best transformation with the quality function in Section III-C. In addition, the time complexity of the propagation mechanism is analyzed in Section III-D.

### A. Multiscale Descriptors

For a point  $\mathbf{x}$ , we describe its local geometric structure with two kinds of information, i.e., eigenvalues and normals. Assume there are  $L$  different support radii to be considered. They are  $r_1 < r_2 < \dots < r_L$ . Within each support radius  $r_l$ , a matrix  $\mathbf{C}_l$  is established as

$$\mathbf{C}_l = \frac{1}{|\mathcal{S}_l|} \sum_{\mathbf{x}_i \in \mathcal{S}_l} (\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T, \quad (2)$$

in which  $l = 1, \dots, L$  and  $\mathcal{S}_l = \{\mathbf{x}_i \mid \|\mathbf{x}_i - \mathbf{x}\| \leq r_l\}$ .<sup>1</sup> Decomposing the  $3 \times 3$  matrix  $\mathbf{C}_l$  with Singular Value Decomposition (SVD), we obtain three eigenvalues  $\lambda_{l1} \geq \lambda_{l2} \geq \lambda_{l3}$  and their corresponding eigenvectors  $\mathbf{n}_{l1}, \mathbf{n}_{l2}, \mathbf{n}_{l3}$ . Mostly for small radius  $r_l$ , points within the support region locate actually on a flat patch. In this situation,  $\mathbf{n}_{l1}, \mathbf{n}_{l2}$  are sensitive to the distributions of the points inside, while the  $\lambda_{l1}, \lambda_{l2}, \lambda_{l3}$  and  $\mathbf{n}_{l3}$  (close to normal of the patch) are not [40], [41]. Therefore, we adopt the three eigenvalues  $\lambda_{l1}, \lambda_{l2}, \lambda_{l3}$  and

<sup>1</sup>Throughout the paper, we use  $\|\cdot\|$  to denote the euclidian distance.

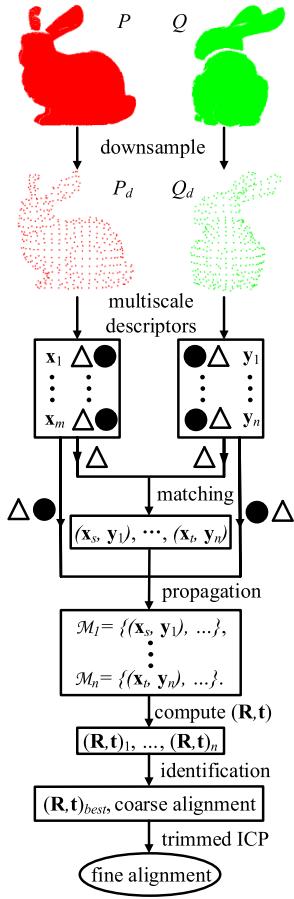


Fig. 1. Flow chart of our algorithm. The hollow triangle and the solid circle represent the eigenvalue-based descriptor  $\mathbf{D}$  and the normal-based descriptor  $\mathbf{N}$ , respectively. The correspondence of each point from  $\mathbf{y}_1$  to  $\mathbf{y}_n$  is generated by performing nearest neighbor search on  $\mathbf{D}$ . During the propagation session, both  $\mathbf{D}$  and  $\mathbf{N}$  are exploited to propagate a seed match (e.g.,  $(\mathbf{x}_s, \mathbf{y}_1)$ ) into a set of matches (e.g.,  $\mathcal{M}_1 = \{(\mathbf{x}_s, \mathbf{y}_1), \dots\}$ ). Among the numerous sets of matches  $\mathcal{M}_1, \dots, \mathcal{M}_n$ , we identify the best transformation with quality function  $E(\mathbf{R}, \mathbf{t})$ . The result is then refined with the trimmed-ICP algorithm.

the eigenvector  $\mathbf{n}_{l3}$  to describe local geometric structure of each point. Let  $\mathbf{s}_l$  be the normalized vector of eigenvalues, which is represented as

$$\mathbf{s}_l = \frac{(\lambda_{l1} \ \lambda_{l2} \ \lambda_{l3})^T}{\lambda_{l1} + \lambda_{l2} + \lambda_{l3}}. \quad (3)$$

In addition, we set the normal  $\mathbf{n}_l$  as  $\mathbf{n}_l = \mathbf{n}_{l3}$  if the eigenvector  $\mathbf{n}_{l3}$  directs towards the viewpoint, and  $\mathbf{n}_l = -\mathbf{n}_{l3}$  otherwise. With no prior knowledge, the default setting for the viewpoint is assumed to be  $(0, 0, 0)^T$ . For simplicity of notation, we combine the information from multiscale normals into a matrix  $\mathbf{N}$ , with

$$\mathbf{N} = (\mathbf{n}_1, \dots, \mathbf{n}_L). \quad (4)$$

Different from eigenvectors, eigenvalues have the good characteristic of rotation-invariant. Yet, considering the slow variation of  $\mathbf{s}_l$  across contiguous support radii, we use their differences instead of themselves to intensify the change of the geometric structure. Normalization added in Eq. (3) is to eliminate the influence of scale (support radius). As a result,

let the difference of vector  $\mathbf{s}_l$  be defined as  $\Delta \mathbf{s}_l = \mathbf{s}_{l+1} - \mathbf{s}_l$ . The eigenvalue based vector  $\mathbf{D}$  is formulated simply by concatenating these  $\Delta \mathbf{s}_l$ , which is denoted as

$$\mathbf{D} = \begin{pmatrix} \Delta \mathbf{s}_1 \\ \dots \\ \Delta \mathbf{s}_{L-1} \end{pmatrix}. \quad (5)$$

The multiscale descriptors of point  $\mathbf{x}_c$  consists of both the matrix  $\mathbf{N}$  and the vector  $\mathbf{D}$ , which is represented as  $(\mathbf{N}, \mathbf{D})$ . The descriptor  $\mathbf{D}$  can be used to find seed matches with nearest neighbour search.

### B. Correspondence Propagation

Due to the low precision of seed matches and the difficulty of recovering the transformation between point clouds with matches of low precision, we present further a mechanism named correspondence propagation to aggregate each seed match into a set of numerous matches, among which the set of matches with high precision exists and is able to provide a good coarse alignment between the point clouds. The inspiration behind our propagation mechanism is that aggregating the matches around a correct match, more correct matches can be obtained, while aggregating the matches around a false match, no correct matches will be resulted.

Let  $(\mathbf{x}_i, \mathbf{y}_j)$  be a seed match, and the initial set of matches constructed from it be  $\mathcal{M} = \{(\mathbf{x}_i, \mathbf{y}_j)\}$ . We aggregate  $\mathcal{M}$  globally by searching for other matches in  $P_d$ ,  $Q_d$  based on the descriptors  $(\mathbf{N}, \mathbf{D})$  of these points. For any  $\mathbf{x} \in P_d \setminus \mathbf{x}_i$ , and  $d = \|\mathbf{x} - \mathbf{x}_i\|$ , we define an angular vector  $\boldsymbol{\theta}$  between descriptor  $\mathbf{D}$  of point  $\mathbf{x}$  and point  $\mathbf{x}_i$ . It is represented as  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_L)^T$ , where  $\theta_l$  is the angle between normals of  $\mathbf{x}$  and  $\mathbf{x}_i$  under the support radius  $r_l$ , that is,

$$\theta_l = \cos^{-1} \langle \mathbf{n}_l^{x_i}, \mathbf{n}_l^x \rangle, \quad l = 1, \dots, L, \quad (6)$$

We search for the correspondence of point  $\mathbf{x}$  among the points in  $Q_d$  whose distance to  $\mathbf{y}_j$  are close to  $d$ . That is,

$$\mathbf{y} \in Q_d \cap \{ \mathbf{y} \mid |d' - d| < \epsilon_1 \}, \quad (7)$$

in which  $d' = \|\mathbf{y} - \mathbf{y}_j\|$ , and the threshold  $\epsilon_1$  reflects the range of search for  $\mathbf{y}$ . The angular vector between  $\mathbf{y}$  and  $\mathbf{y}_j$  is expressed as  $\boldsymbol{\theta}' = (\theta'_1, \dots, \theta'_L)^T$ . Similar to  $\theta_l$ , the calculation of  $\theta'_l$  follows

$$\theta'_l = \cos^{-1} \langle \mathbf{n}_l^{y_j}, \mathbf{n}_l^y \rangle, \quad l = 1, \dots, L. \quad (8)$$

We then define the angular dissimilarity between  $\mathbf{x}$  and  $\mathbf{y}$  as

$$\Delta \theta = \begin{cases} \frac{1}{L} \sum_{l=1}^L |\theta_l - \theta'_l|, & \text{if all } |\theta_l - \theta'_l| < \epsilon_2, \\ \text{Inf}, & \text{otherwise.} \end{cases} \quad (9)$$

Within the searched range  $Q_d \cap \{ \mathbf{y} \mid |d' - d| < \epsilon_1 \}$ , the point  $\mathbf{y}$  with the smallest dissimilarity to  $\mathbf{x}$  is considered as the correspondence of  $\mathbf{x}$  when the distance between its eigenvalue-based descriptor and that of  $\mathbf{x}$  is less than a certain threshold, i.e.,  $\|\mathbf{D}^x - \mathbf{D}^y\| < \epsilon_3$ . The set  $\mathcal{M}$  is then updated with the newly searched match  $(\mathbf{x}, \mathbf{y})$  to be  $\mathcal{M} = \mathcal{M} \cup (\mathbf{x}, \mathbf{y})$ . If  $\|\mathbf{D}^x - \mathbf{D}^y\| \geq \epsilon_3$ , the set  $\mathcal{M}$  will be kept unchanged. After

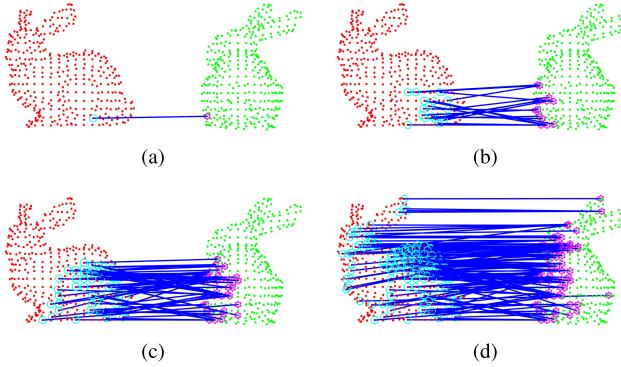


Fig. 2. An example of the correspondence propagation mechanism. We show the propagation result of a seed match  $(\mathbf{x}_i, \mathbf{y}_j)$  according to the distance  $d = \|\mathbf{x} - \mathbf{x}_i\|$ . By searching in  $Q_d$  for the correspondence of  $\mathbf{x}$  in  $P_d$  with larger and larger distances to  $\mathbf{x}_i$ , we propagate the seed match into a larger and larger match set  $\mathcal{M} = \{(\mathbf{x}_i, \mathbf{y}_j), \dots\}$ . The propagation stops when all points in  $\mathbf{x} \in Q_d$  have been traversed. (a)  $d = 0\text{cm}$ . (b)  $d < 3.90\text{cm}$ . (c)  $d < 6.29\text{cm}$ . (d)  $d = d_{\max}$ .

all points in  $P_d$  is processed, we finish the propagation of the seed match  $(\mathbf{x}_i, \mathbf{y}_j)$ , and the establishment of  $\mathcal{M}$ . Fig. 2 is an example of the propagation result of a seed match. For clarity, we show matches according to the distance  $d = \|\mathbf{x} - \mathbf{x}_i\|$  in the example. Yet, in the implementation, the propagation is actually performed following the arrangement of the data, without a sorting of their distances to the seed  $\mathbf{x}_i$ .

Assume the number of seed matches we obtain based on nearest neighbour search of  $\mathbf{D}$  is  $n$ . Thus, by applying the propagation mechanism to the seed matches each,  $n$  sets of matches are obtained, which are denoted as  $\mathcal{M}_1, \dots, \mathcal{M}_n$ .

### C. Quality Function

Based on the set  $\mathcal{M}$ , we estimate the candidate relative pose between the point clouds  $P$  and  $Q$ . To improve the robustness of our estimation to outliers, we use RANSAC [15] to find the maximum consensus set in  $\mathcal{M}$ . And the transformation  $(\mathbf{R}, \mathbf{t})$  is computed following the approach in [42]. We measure the quality of the transformation with distance errors between the down-sampled point clouds  $P_d, Q_d$ . In addition, to make the measure robust to partial overlap, only the top  $\eta$  points with smaller distances are used in the calculation. And  $\eta$  represents the overlap ratio while aligning  $P$  to  $Q$ .

In particular, let  $\hat{\mathbf{y}}_k$  be the nearest neighbour of  $\mathbf{y}_k$  in the point cloud  $\hat{Q}_d = \{\hat{\mathbf{y}} \mid \hat{\mathbf{y}} = \mathbf{R}\mathbf{x} + \mathbf{t}\}$ , and  $\mathcal{Y}_{\min}$  be the collection of  $\mathbf{y}_k \in Q_d$  whose distance to  $\hat{\mathbf{y}}_k$  is among the top  $\eta$  smallest. That is,  $|\mathcal{Y}_{\min}| = \lceil \eta n \rceil$ , and for any  $\mathbf{y}_k \in \mathcal{Y}_{\min}, \mathbf{y}_{k'} \in Q_d \setminus \mathcal{Y}_{\min}$ , we have  $\|\hat{\mathbf{y}}_k - \mathbf{y}_k\|^2 \leq \|\hat{\mathbf{y}}_{k'} - \mathbf{y}_k\|^2$ . Mathematically, the quality function is then computed as follows

$$E(\mathbf{R}, \mathbf{t}) = \sum_{\mathbf{y}_k \in \mathcal{Y}_{\min}} \|\hat{\mathbf{y}}_k - \mathbf{y}_k\|^2. \quad (10)$$

Based on the sets of matches  $\mathcal{M}_1, \dots, \mathcal{M}_n$ , we get multiple transformations between the point clouds,  $(\mathbf{R}, \mathbf{t})_1, \dots, (\mathbf{R}, \mathbf{t})_n$ . The quality of these transformations is measured with function Eq. (10). And the one which yields the smallest function value is selected as the best transformation  $(\mathbf{R}, \mathbf{t})_{best}$ . We use it to fulfill a coarse alignment of the point clouds  $P$  and  $Q$ . Next,

trimmed-ICP algorithm (with  $\eta$  overlap) [9] is applied to refine the registration result from coarse to fine.

### D. Time Complexity Analysis

In the proposed approach, the computation of  $(\mathbf{N}, \mathbf{D})$  for point clouds  $P_d$  and  $Q_d$  is very fast, which mainly requires only  $O(L(n+m))$  3D range searching and SVD decomposition.

The time complexity for correspondence propagation is higher. The total number of propagation required is  $O(n)$ . In each propagation, computing the distances of all points in  $P_d$  to  $\mathbf{x}_i$  takes  $O(m)$  time, while computing the distances of all points in  $Q_d$  to  $\mathbf{y}_j$  takes  $O(n)$  time. Following our notation in Section III-B and exploiting 1D range searching, finding  $\mathbf{y}$  with  $d' = \|\mathbf{y} - \mathbf{y}_j\|$  within range  $\epsilon_1$  of each  $d = \|\mathbf{x} - \mathbf{x}_i\|$  takes  $O(\log n)$  time, regardless of the time for reporting the results. Since the above mentioned 1D range searching has to be done for each  $\mathbf{x}$  from  $\mathbf{x}_1$  to  $\mathbf{x}_m$ , the total computation time is therefore  $O(m) \cdot O(\log n)$ . Besides, a construction of the search tree takes  $O(n \log n)$  time. As a result, the whole propagation mechanism consumes time  $T$  to be accomplished, in which  $T$  is computed as follows

$$\begin{aligned} T &= O(n) \cdot (O(n) + O(m) + O(n \log n) + O(m) \cdot O(\log n)) \\ &= O(n) \cdot O(n + m + (n + m) \log n) \\ &= O(n(n + m) \log n). \end{aligned} \quad (11)$$

In the above analysis, we arbitrarily assume  $n \leq m$ . For the situation when  $n \gg m$ , the two point clouds  $P_d, Q_d$  can be swapped to reduce the time complexity.

The computation of all transformations and a determination of the best one takes  $O(n)$  time, and trimmed ICP takes only several iterations to return a fine alignment result.

## IV. EXPERIMENTS

We evaluate performance of the proposed approach firstly with the Stanford 3D Scanning Repository.<sup>2</sup> The Stanford Repository contains scan data with full overlaps, partial overlaps as well as large transformations, and their ground truth transformations are all available. Therefore, we use it in the first place to test robustness of the proposed approach. And a comparison with many global algorithms are also provided for a better evaluation. The included algorithms are traditional descriptors combined with RANSAC [15], FPFH+SAC-IA [13], Super4PCS [34], Go-ICP [37] and BB+ICP [36]. For the traditional descriptors, we consider SHOT [14] and FPFH [13], while the required keypoints are detected with ISS\_BR detector (i.e., ISS [11] with boundary removed). Our selection of the detector ISS\_BR and the descriptors SHOT, FPFH is because they are not only effective but also efficient [17], [18]. Next, the proposed approach is applied to reconstruct the wide-baseline scans of CASTLE and CITY from [43], which are large-scale data obtained from outdoor scans and consist of merely 5 and 3 views respectively. The wide-baselines make a reconstruction of the two models very challenging. Finally, we show that the proposed approach

<sup>2</sup><http://graphics.stanford.edu/data/3Dscanrep/>

TABLE I  
ALIGNMENT RESULTS FOR POINT CLOUDS WITH SIGNIFICANT AND LIMITED OVERLAPS

	Significant			Limited			
	(a)	(b)	(c)	(a)	(b)	(c)	(d)
GT overlap	91%	79%	91%	31%	51%	52%	24%
$\eta$	90%	100%	100%	50%	60%	50%	30%
#Correct match: before	48	28	88	17	43	28	6
precision: before	0.13	0.08	0.22	0.05	0.14	0.16	0.04
#Correct match: after	163	70	104	58	65	18	20
precision: after	<b>0.60</b>	<b>0.74</b>	<b>0.57</b>	<b>0.38</b>	<b>0.72</b>	<b>1.00</b>	<b>0.87</b>
Rot ( $^{\circ}$ )/Tran (m): coarse	0.55/0.0009	1.08/0.0020	0.39/0.0008	1.42/0.0022	1.81/0.0031	1.59/0.0023	1.02/0.0016
Rot ( $^{\circ}$ )/Tran (m): fine	0.55/0.0009	0.94/0.0016	0.24/0.0006	0.90/0.0010	1.56/0.0030	0.83/0.0004	0.78/0.0016
Runtime (s)	14	12	14	15	8	3	6

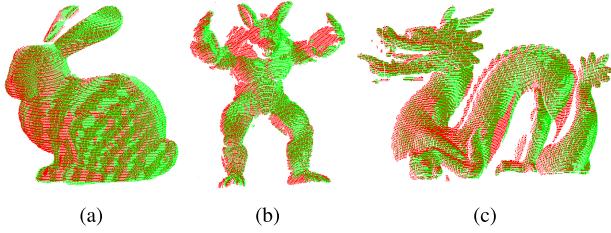


Fig. 3. Registration of point clouds with significant overlaps. (a) Bunny. (b) Armadillo. (c) Dragon.

is able to be used to register low-resolution data captured by Kinect, using the Microsoft 7-scenes dataset [44].

According to our notations in Section III-A, the maximum support radius for calculating the descriptors ( $\mathbf{N}, \mathbf{D}$ ) is  $r_L$ . It can be either manually set or automatically determined based on the mesh area of the point clouds [17], [45]. To compute  $\mathbf{s}_l$  and  $\mathbf{n}_l$  in each level, we set  $r_l = \frac{l}{L} \times r_L$ ,  $L = 4$ . Let the grid step used for uniform down-sampling be  $\rho$ , which is determined by the number of points expected to use for aligning the two point clouds.  $\epsilon_1, \epsilon_2$  and  $\epsilon_3$  are set to  $\rho/2, 10^\circ$  and 0.2 respectively in our algorithm. The whole algorithm is implemented in MATLAB on a machine with the CPU Intel Core i3-4160. We will release our code with publication of the paper.

#### A. Stanford Repository

We first demonstrate the robustness of our algorithm to point cloud pairs with limited overlap, large transformation and different noise level.  $\rho$  equals to 0.01. Then, a comparison with other global algorithms are reported. Finally, we test sensitivity to  $\eta$  of our algorithm.

1) *Robustness of the Algorithm:* The first experiment is to register selected point cloud pairs which contain significant overlaps, i.e., larger than 75%. We consider a match as a correct one if the distance between the matched point and the ground truth correspondence is within 0.02m. And the registration accuracy is measured with the rotation and translation errors. Fig. 3 visualizes the results of our alignment, with red dots representing the transformed point cloud  $P$ , and green dots representing the fixed point cloud  $Q$ . In reality, limited overlaps between point clouds prevalently exist, which can be caused by either large transformation or occlusions. Therefore, it is critical for an algorithm to be robust to these challenging

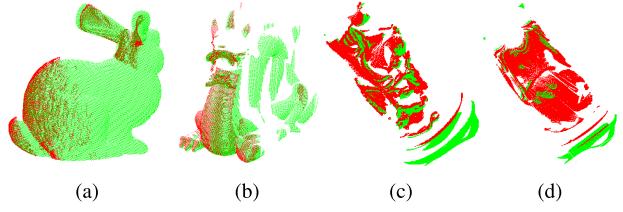


Fig. 4. Registration of point clouds with limited overlaps. (a) Bunny. (b) Dragon. (c) Happy. (d) Happy.

situations as well. We use point cloud pairs with overlaps down to 24% to test performance of the proposed approach on cases of limited overlaps. Fig. 4 shows the registration results. Table I reports more detailed results for each point cloud pairs used in the first and the second experiments, including the ground truth overlap (GT overlap), the settings of  $\eta$ , the number of correct seed matches searched using the eigenvalue-based descriptor  $\mathbf{D}$  (#Correct match: before), the precision of the seed matches (precision: before), the number of correct matches in the set  $\mathcal{M}$  which provides the best transformation ( $\mathbf{R}, \mathbf{t}_{best}$ ) (#Correct match: after), the precision of this set of matches (precision: after), the rotation (Rot) and translation (Tran) errors of the registration in degrees and metres respectively, as well as the runtime each in seconds. We define the precision of the matches as

$$\text{precision} = \frac{\text{The number of correct matches}}{\text{The number of matches}}. \quad (12)$$

The ground truth overlaps of the seven point cloud pairs vary from 24% to 100%. From Table I, it is noticed that the proposed algorithm is able to register them with high accuracy in efficient time.

In the next experiment, we use the scan data bun000 and bun090 to demonstrate robustness of the proposed approach to Gaussian noise. Firstly, the original point clouds are all normalized into the domain  $[-1, 1]^3$ , and Gaussian noise with a standard deviation  $\sigma$  is added to the normalized point clouds. Then, we recover them into their original scale. The registration results under influence of different levels of noise are shown in Fig. 5, where the alignment of the original point clouds (i.e.,  $\sigma = 0.00$ ) is also shown for better comparison. Their Rot/Tran errors in sequence are  $1.12^\circ/0.0004\text{m}$ ,  $2.46^\circ/0.0042\text{m}$ ,  $3.07^\circ/0.0067\text{m}$ ,  $5.39^\circ/0.0121\text{m}$ , which take respectively 14s, 19s, 25s, 27s to be implemented. Table II reports the matching conditions of the seed matches and the set

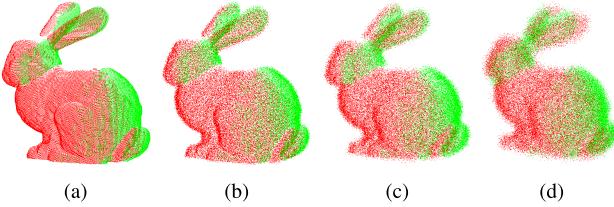


Fig. 5. Registration of bun000 and bun090 with different levels of Gaussian noise. (a)  $\sigma = 0.00$ . (b)  $\sigma = 0.01$ . (c)  $\sigma = 0.02$ . (d)  $\sigma = 0.03$ .

TABLE II  
MATCHING RESULTS BEFORE AND AFTER THE  
CORRESPONDENCE PROPAGATION

Gaussian noise	$\sigma = 0.00$	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.03$
#Correct match: before	19	20	26	30
precision: before	0.05	0.05	0.05	0.05
#Correct match: after	64	77	36	7
precision: after	<b>0.47</b>	<b>0.43</b>	<b>0.40</b>	<b>0.29</b>

$\mathcal{M}$  with which the best transformation  $(\mathbf{R}, \mathbf{t})_{best}$  is computed. It is exciting that our multiscale descriptors and propagation mechanism still provide acceptable precision of matches when the noise is significant (e.g.,  $\sigma = 0.03$ ). This follows that even for point clouds with large transformation, small overlap, and high noise, the proposed approach is still effective for recovering the transformation between them.

2) *Comparison With Other Methods:* We use implementations of ISS\_BR, SHOT, FPFH and SAC-IA in the standard PCL library. SHOT is calculated with a constant support radius of 0.03m, and FPFH is calculated with 100 nearest neighbours. As to the Go-ICP algorithm, we use the code released on the website of Yang.<sup>3</sup> The results of Super4PCS and BB+ICP are computed with the codes posted on github<sup>4,5</sup> by the authors.

The initial point clouds of Fig. 3(a) contain significant overlap. A registration of them is easy. As to the method ISS\_BR+SHOT, we search for matches with SHOT for ISS\_BR detected points, estimate the transformation with RANSAC, and then optimize the result with trimmed ICP. The method ISS\_BR+FPFH is conducted similarly. A comparison of multiple approaches with the proposed algorithm is shown in Table III. While running the methods ISS\_BR+SHOT and ISS\_BR+FPFH, the computation of keypoints and descriptors takes the main runtime, which is 121 and 167 seconds respectively. Therefore, we report their runtime as 121+, 167+. It is noticed from the table that these methods altogether show satisfactory registration results.

For the initial point clouds in Fig. 4(d), they have small transformation but significant occlusion, which results in their small overlap. We compare performance of the proposed algorithm with other methods on this pair of point clouds in Table IV. It can be seen that Go-ICP and BB+ICP fail to show acceptable results for such point clouds due to their small overlap, while the other approaches still show good registration results.

<sup>3</sup><http://iitlab.bit.edu.cn/mcislab/~yangjiaolong/go-icp/>

<sup>4</sup><https://github.com/nmellado/Super4PCS>

<sup>5</sup><https://github.com/jstraub/dpOptTrans>

TABLE III  
PERFORMANCE COMPARISON ON POINT CLOUDS  
WITH SIGNIFICANT OVERLAP

Method	Time (s)	Rot ( $^{\circ}$ )/Tran (m)
Proposed	14	0.55/0.0009
ISS_BR+SHOT	121+	1.45/0.0008
ISS_BR+FPFH	167+	0.61/0.0008
FPFH+SAC-IA	16	0.93/0.0021
Super4PCS	0.2	0.32/0.0007
Go-ICP	51	0.34/0.0006
BB+ICP	16	0.61/0.0013

TABLE IV  
PERFORMANCE COMPARISON ON POINT CLOUDS WITH SMALL  
TRANSFORMATION BUT SMALL OVERLAP

Method	Time (s)	Rot ( $^{\circ}$ )/Tran (m)
Proposed	6	0.78/0.0016
ISS_BR+SHOT	155+	1.44/0.0018
ISS_BR+FPFH	186+	1.17/0.0035
FPFH+SAC-IA	16	0.61/0.0052
Super4PCS	25	1.06/0.0027
Go-ICP	71	148.11/0.1700
BB+ICP	32	91.92/0.1880

TABLE V  
PERFORMANCE COMPARISON ON POINT CLOUDS  
WITH LARGE TRANSFORMATION

Method	Time (s)	Rot ( $^{\circ}$ )/Tran (m)
Proposed	15	1.12/0.0004
ISS_BR+SHOT	104+	100.98/0.0422
ISS_BR+SHOT+CP	104+	0.94/0.0027
ISS_BR+FPFH	156+	168.62/0.1637
ISS_BR+FPFH+CP	156+	0.37/0.0009
FPFH+SAC-IA	19	6.73/0.0096
Super4PCS	9	3.89/0.0059
Go-ICP	1235	1.11/0.0006
BB+ICP	33	78.22/0.0693

For point clouds with large transformation, such as the initial pair of Fig. 5(a), ISS\_BR+SHOT and ISS\_BR+FPFH can no longer recover the relative pose between them well. BB+ICP fails as well. And it takes too much time for Go-ICP to report good results for the point clouds. Nonetheless, we find that our correspondence propagation mechanism can be combined with ISS\_BR+SHOT or ISS\_BR+FPFH to provide good registration, which are denoted as ISS\_BR+SHOT+CP and ISS\_BR+FPFH+CP. Table V shows the results.

Although it is feasible to combine the traditional keypoint detector and local descriptor with our propagation mechanism to align point clouds with large transformation, such a combined approach tends to malfunction when the data are with high noise. This is because that the keypoint detector and the local descriptor, especially the latter which depends highly on the relative positions between points, are sensitive to noise [17], [18]. Table VI compares the results of the proposed approach, ISS\_BR+SHOT+CP, ISS\_BR+FPFH+CP, FPFH+SAC-IA and Super4PCS on point cloud pairs shown in Fig. 5(b), Fig. 5(c) and Fig. 5(d). It is noticed that ISS\_BR+SHOT+CP and ISS\_BR+FPFH+CP are unable to

TABLE VI

PERFORMANCE COMPARISON ON POINT CLOUDS WITH LARGE TRANSFORMATION AND DIFFERENT GAUSSIAN NOISE

Method	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.03$
	Rot ( $^{\circ}$ )/Tran (m)		
Proposed	2.46/0.0042	3.07/0.0067	5.39/0.0121
ISS_BR+SHOT+CP	1.62/0.0041	116.58/0.0222	92.27/0.0428
ISS_BR+FPFH+CP	1.61/0.0037	114.97/0.0215	116.98/0.0991
FPFH+SAC-IA	151.08/0.1569	123.26/0.1300	130.21/0.0736
Super4PCS	90.18/0.0001	90.18/0.0001	90.18/0.0001

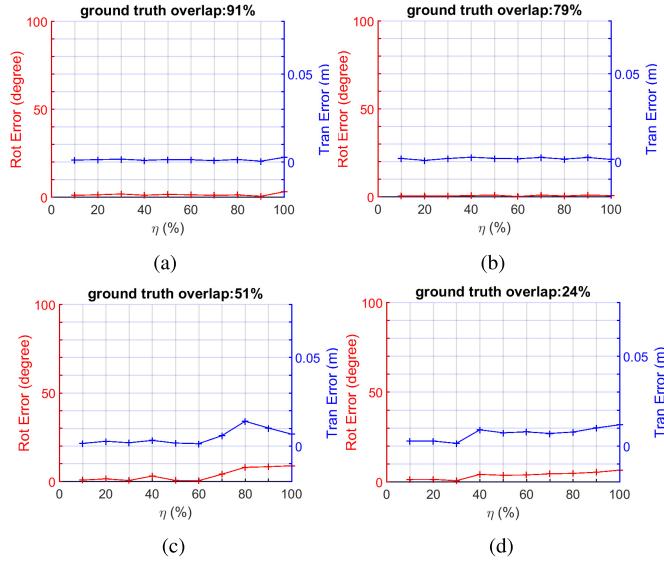


Fig. 6. Rotation error curve and translation error curve of  $\eta$ . (a) Point clouds correspond to Fig. 3(a). (b) Point clouds correspond to Fig. 3(b). (c) Point clouds correspond to Fig. 4(b). (d) Point clouds correspond to Fig. 4(d).

provide good alignment as the noise level becomes higher. In addition, FPFH+SAC-IA and Super4PCS fail to register the three point cloud pairs altogether. In particular, Super4PCS always returns an identity transformation matrix for the registration.

3) *Sensitivity to  $\eta$ :* To register point clouds with limited overlap, an estimated overlap is generally required. The estimation usually takes efforts [9], [34], [37]. Therefore, we test the sensitivity of our algorithm to different settings of the estimated overlap  $\eta$ . Four rotation error curves and translation error curves of  $\eta$  for different pairs of point clouds are provided in Fig. 6. Fig. 6(a) plots the result for point clouds in Fig. 3(a). Fig. 6(b) plots the result for point clouds in Fig. 3(b). Fig. 6(c) plots the result for point clouds in Fig. 4(b). Fig. 6(d) plots the result for point clouds in Fig. 4(d). Particularly, to plot the rotation and translation error curves of  $\eta$  on each point cloud pair, we keep the rest settings of the algorithm and increase  $\eta$  from 10% to 100% gradually. It can be seen that the algorithm performs stable around and when  $\eta$  is less than the ground truth overlap. This suggests that a small and fixed  $\eta$  could be used in our algorithm. We test the performance on four point cloud sets, Bunny, Armadillo, Dragon and Happy Buddha, with  $\eta = 0.3$  and 400 points sampled around. In total, 108 point cloud pairs with a rotation no more than  $90^{\circ}$  are selected. We show the

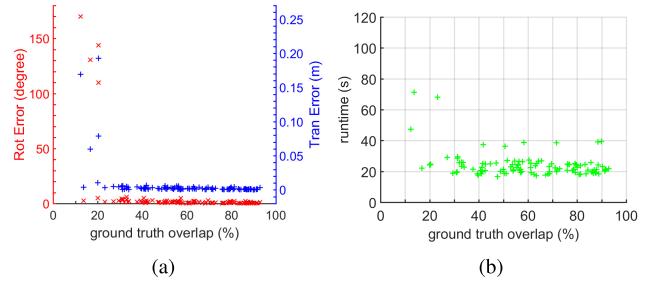


Fig. 7. Registration results on the four models with  $\eta = 0.3$ . (a) The rotation and translation errors for point clouds with different overlaps. (b) The runtime for aligning point clouds with different overlaps.

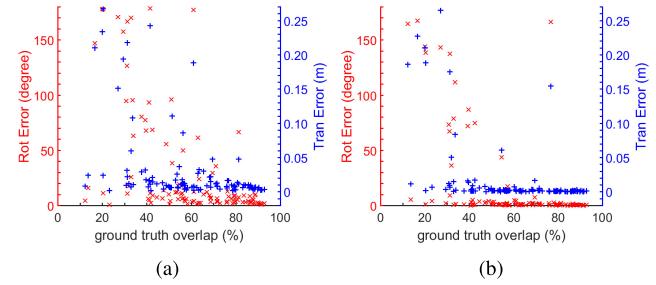


Fig. 8. Rotation and translation errors of FPFH+SAC-IA and Super4PCS on the 108 point cloud pairs. (a) FPFH+SAC-IA. (b) Super4PCS (with 400 sampled points and a constant overlap 0.3).

rotation and translation errors, and runtime with respect to their ground truth overlaps in Fig. 7. For all point clouds with their ground truth overlaps more than 0.2, our algorithm registers them satisfactorily in efficient time. The rotation and translation errors of FPFH+SAC-IA and Super4PCS on the 108 point cloud pairs are plotted in Fig. 8(a) and Fig. 8(b) as a comparison. It is noticed that both FPFH+SAC-IA and Super4PCS fail for a number of point cloud pairs with overlaps larger than 0.2.

Finally, we show two reconstructions, Dragon and Happy Buddha. The scans of both models take a  $24^{\circ}$  rotational increment around the vertical axis of its statue, with totally 15 views included. Fig. 9 shows the color coded reconstruction results of our approach, which is quite satisfactory. The minimum, average, and maximum runtime for aligning each point cloud pair is 5s, 12s, 17s on the Dragon dataset, and 7s, 10s, 15s on the Happy Buddha dataset. That is, a reconstruction of the two models each takes merely 2 to 3 minutes.

### B. Castle and City

The CASTLE and CITY scans are recently collected by Zeisl *et al.* [43], and consist of point clouds with wide-baselines, which makes the reconstruction of them quite difficult. Zeisl *et al.* reconstructed the two models by exploring information contained in both the 2D images and 3D point clouds combinationally. Instead, we fulfill the reconstruction of CASTLE and CITY by using the 3D scans singly. To facilitate the reconstruction, we down-sample each original point cloud to 25K points around in CITY and 100K points around in CASTLE, using an identical sample grid. The ground truth

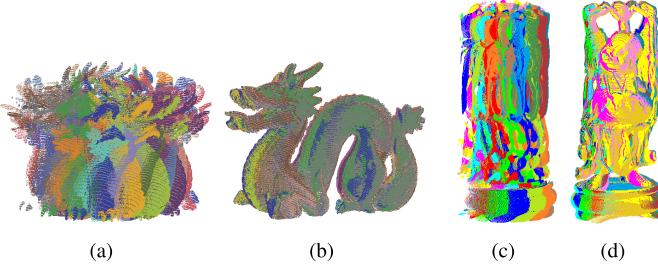


Fig. 9. Reconstruction of the Dragon and Happy Buddha models. (a) The color-coded point clouds in the Dragon dataset. (b) The reconstructed Dragon. (c) The color-coded point clouds in the Happy Buddha dataset. (d) The reconstructed Happy Buddha.

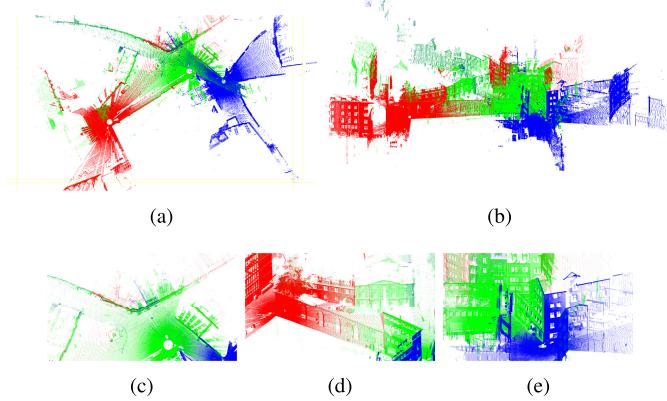


Fig. 10. The reconstruction of CITY. (a) Top view. (b) Free view. (c) Top in Detail. (d) Corner. (e) Facade.

overlaps of CITY vary from 33% to be 52%, while those of CASTLE varies from 32% to 83%. During the reconstruction, we set  $\eta$  as a constant 0.3. And around 600 points are sampled in each point cloud to fulfill the alignment of CITY. The reconstruction takes only 48s. Fig. 10(a) and Fig. 10(b) show the reconstructed CITY from the top view and a free view. In addition, further details are included in Fig. 10(c), 10(d), 10(e). And for CASTLE, around 1500 points are sampled in each point cloud to fulfill its alignment. The whole process takes 12 minutes. We show the top view and a free view of the reconstructed CASTLE in Fig. 11(a), 11(b), and further details in Fig. 11(c), 11(d), 11(e), 11(f). Despite of the challenges caused by noise, large transformation as well as limited overlaps in such data, we notice that the proposed approach successfully reconstruct the two models.

### C. Kinect Data

The data captured by Kinect are with low-resolution and non-trivial distortions exist in the depth images [46]–[49]. Hence, it is almost impossible to align point clouds collected by Kinect with high accuracy. Yet, we demonstrate on the Chess and Heads scenes of the Microsoft 7-scenes dataset [44], that the proposed approach can be used to recover the rough structures of the scenes. Because the depth and color images are captured by the depth and RGB cameras in different positions, the RGB image doesn't match the depth image strictly. Therefore, we show the alignment of the Chess scene

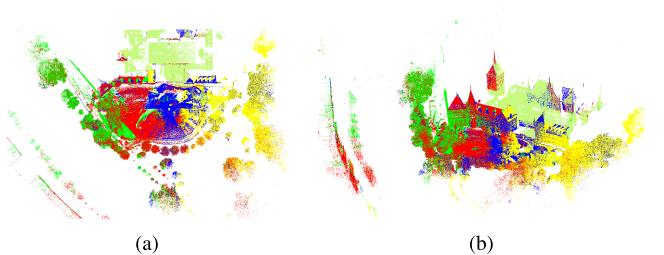


Fig. 11. The reconstruction of CASTLE. (a) Top view. (b) Free view. (c) Spire. (d) Spire. (e) Spire. (f) Ground.

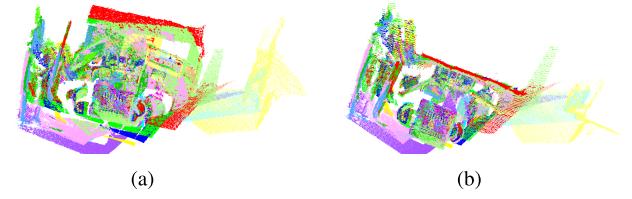


Fig. 12. Alignment results of the Chess sequence.

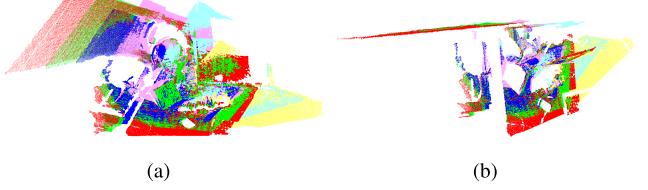


Fig. 13. Alignment results of the Heads sequence.

with the textures removed in Fig. 12. And the alignment of another scene, Heads, is also shown in Fig. 13 with the textures removed. Point clouds from the two scenes are all registered with  $\eta = 0.3$  and 1000 sampling points around. We register every 50 frames from partial sequences of the Chess scene, and every 20 frames from partial sequences of the Heads scene.

## V. CONCLUSION

In this paper, we proposed a novel approach for robust point clouds registration, which is achieved using multiscale descriptors of uniformly sampled points and correspondence propagation. Our descriptors can be constructed very fast, and the time complexity of the propagation mechanism is also pleasant. The proposed approach is applicable to align point clouds with small overlap, high noise and large transformation. We demonstrate its fine performance on the standard database, with a comparison to the traditional descriptor-based method and many other approaches. In addition, we also show with examples the promising application of the proposed approach in large-scale reconstruction, which is very time-saving. The

proposed approach can also be used to align low-resolution data collected by Kinect.

## REFERENCES

- [1] G. Blais and M. D. Levine, "Registering multiview range data to create 3D computer objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 820–824, Aug. 1995.
- [2] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM–3D mapping outdoor environments," *J. Field Robot.*, vol. 24, nos. 8–9, pp. 699–722, 2007.
- [3] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Auto. Robots*, vol. 34, no. 3, pp. 133–148, 2013.
- [4] A. S. Mian, M. Bennamoun, and R. Owens, "3D model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.
- [5] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.
- [6] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.
- [7] P. J. Besl and N. D. McKay, "A method for registration of 3D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
- [8] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image Vis. Comput.*, vol. 10, no. 3, pp. 145–155, Apr. 1992.
- [9] D. Chetverikov, D. Stepanov, and P. Krsek, "Robust Euclidean alignment of 3D point sets: The trimmed iterative closest point algorithm," *Image Vis. Comput.*, vol. 23, no. 3, pp. 299–309, 2005.
- [10] S. Holzer, J. Shotton, and P. Kohli, "Learning to efficiently detect repeatable interest points in depth data," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 200–213.
- [11] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3D object recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Sep. 2009, pp. 689–696.
- [12] I. Sipiran and B. Bustos, "Harris 3D: A robust extension of the Harris operator for interest point detection on 3D meshes," *Vis. Comput.*, vol. 27, no. 11, pp. 963–976, 2011.
- [13] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, Sep. 2009, pp. 3212–3217.
- [14] F. Tombari, S. Salti, and L. Stefano, "Unique shape context for 3D data description," in *Proc. ACM Workshop 3D Object Retr.*, 2010, pp. 57–62.
- [15] 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.
- [16] O. J. Woodford, M.-T. Pham, A. Maki, F. Perbet, and B. Stenger, "Demisting the Hough transform for 3D shape recognition and registration," *Int. J. Comput. Vis.*, vol. 106, no. 3, pp. 332–341, 2014.
- [17] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, "A comprehensive performance evaluation of 3D local feature descriptors," *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 66–89, 2016.
- [18] F. Tombari, S. Salti, and L. Stefano, "Performance evaluation of 3D keypoint detectors," *Int. J. Comput. Vis.*, vol. 102, nos. 1–3, pp. 198–220, 2013.
- [19] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3-D Digit. Imag. Modeling*, 2001, pp. 145–152.
- [20] B. Jian and B. C. Vemuri, "A robust algorithm for point set registration using mixture of Gaussians," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Oct. 2005, pp. 1246–1251.
- [21] Y. Tsin and T. Kanade, "A correlation-based approach to robust point set registration," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 558–569.
- [22] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2262–2275, Dec. 2010.
- [23] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," *J. Field Robot.*, vol. 24, no. 10, pp. 803–827, 2007.
- [24] T. D. Stoyanov, M. Magnusson, H. Andreasson, and A. Lilienthal, "Fast and accurate scan registration through minimization of the distance between compact 3D NDT representations," *Int. J. Robot. Res.*, vol. 31, no. 12, pp. 1377–1393, 2012.
- [25] D. Campbell and L. Petersson, "An adaptive data representation for robust point-set registration and merging," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 2015, pp. 4292–4300.
- [26] R. Sandhu, S. Dambreville, and A. Tannenbaum, "Point set registration via particle filtering and stochastic dynamics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1459–1473, Oct. 2010.
- [27] C. Papazov and D. Burschka, "Stochastic global optimization for robust point set registration," *Comput. Vis. Image Understand.*, vol. 115, no. 12, pp. 1598–1609, 2011.
- [28] L. Silva, O. R. P. Bellon, and K. L. Boyer, "Precision range image registration using a robust surface interpenetration measure and enhanced genetic algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 762–776, May 2005.
- [29] C. Robertson and R. B. Fisher, "Parallel evolutionary registration of range data," *Comput. Vis. Image Understand.*, vol. 87, no. 1, pp. 39–50, 2002.
- [30] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 356–369.
- [31] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust global registration," in *Proc. Eurograph. Symp. Geometry Process.*, vol. 2, no. 3, pp. 1–5, 2005.
- [32] J.-C. Bazin, Y. Seo, and M. Pollefeys, "Globally optimal consensus set maximization through rotation search," in *Proc. Asian Conf. Comput. Vis.*, 2012, pp. 539–551.
- [33] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 1–85, 2008.
- [34] N. Mellado, D. Aiger, and N. J. Mitra, "Super 4PCS fast global pointcloud registration via smart indexing," *Comput. Graph. Forum*, vol. 33, no. 5, pp. 205–215, 2014.
- [35] A. Makadia, A. Patterson, and K. Daniilidis, "Fully automatic registration of 3D point clouds," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2006, pp. 1297–1304.
- [36] J. Straub, T. Campbell, J. P. How, and J. W. Fisher, III (Mar. 2016). "Efficient globally optimal point cloud alignment using Bayesian nonparametric mixtures." [Online]. Available: <https://arxiv.org/abs/1603.04868>
- [37] J. Yang, H. Li, and Y. Jia, "Go-ICP: Solving 3D registration efficiently and globally optimally," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 2013, pp. 1457–1464.
- [38] A. Bustos, T.-J. Chin, and D. Suter, "Fast rotation search with stereographic projections for 3D registration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Apr. 2014, pp. 3930–3937.
- [39] D. Campbell and L. Petersson, "GOGMA: Globally-optimal Gaussian mixture alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5685–5694.
- [40] T. Kato, *Perturbation Theory for Linear Operators*. Berlin, Germany: Springer, 2013.
- [41] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD, USA: Hopkins Univ. Press, 2012.
- [42] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, 2nd ed. Upper Saddle River, NJ, USA: Pearson Education, 2011.
- [43] B. Zeisl, K. Koser, and M. Pollefeys, "Automatic registration of RGB-D scans via salient directions," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 2013, pp. 2808–2815.
- [44] B. Glocker, S. Izadi, J. Shotton, and A. Criminisi, "Real-time RGB-D camera relocalization," in *Proc. Mixed Augmented Reality*, 2013, pp. 173–179.
- [45] A. Zaharescu, E. Boyer, and R. Horaud, "Keypoints and local descriptors of scalar functions on 2D manifolds," *Int. J. Comput. Vis.*, vol. 100, no. 1, pp. 78–98, 2012.
- [46] D. Herrera, J. Kannala, and J. Heikkilä, "Joint depth and color camera calibration with distortion correction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 2058–2064, Oct. 2012.
- [47] J. Smisek, M. Jancosek, and T. Pajdla, *3D With Kinect*. London, U.K.: Springer, 2013.
- [48] J. C. K. Chow and D. D. Lichti, "Photogrammetric bundle adjustment with self-calibration of the primesense 3D camera technology: Microsoft Kinect," *IEEE Access*, vol. 1, pp. 465–474, Sep. 2013.
- [49] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of Kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, Feb. 2012.



**Huan Lei** received the B.S. degree from Xidian University in 2012. She is currently a Research Assistant with Xidian University. Her research interests include computer vision, machine learning, and object recognition.



**Long Quan** received the Ph.D. degree in computer science from INPL, France, in 1989. Before joining the Department of Computer Science with The Hong Kong University of Science and Technology in 2001, he has been a French CNRS Senior Research Scientist at INRIA in Grenoble. His research interests focus on 3-D reconstruction, structure from motion, vision geometry, and image-based modeling.



**Guang Jiang** (M'03) received the B.S., M.S., and Ph.D. degrees from Xidian University, Xi'an, China, in 1993, 1998, and 2003, respectively. He was with the Hong Kong University of Science and Technology, The Chinese University of Hong Kong, and the Xian Research Institute of Navigation Technique. He is currently an Associate Professor with the School of Telecommunications Engineering, Xidian University. His current research interests include image processing and computer vision. His main research goals are to develop approaches to analyze, understand and code the images or videos of real world objects and scenes based on vision geometry. He serves as a Reviewer for journals of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, *Internal Journal of Computer Vision, Image and Vision Computing*, and *Machine Vision and Application* and main conferences in computer vision.