A Distinctive Binary Descriptor and Two-Point RANSACWC for Point Cloud Registration

Wuyong Tao, Shaoping Xu, Wei Huang, Shufan Hu, and Meng Pang

***Abstract*—Point cloud registration is a fundamental problem in many applications. The point cloud registration based on local shape descriptor has been widely researched. In order to further improve the performance of registration, a novel registration method is proposed in this article. First, a binary descriptor is designed to establish correspondences between two point clouds. The descriptor has high descriptiveness. Thus, more correct correspondences are established. Then, a 3-D transformation estimation technique is developed, in which multiple constraints are used to accelerate the computation. When the randomly selected correspondences do not satisfy the constraints, the iteration is skipped. Finally, the experiments are performed to analyze the descriptor and 3-D transformation estimation technique. The comparison with the existing descriptors is implemented on three datasets. The results demonstrate that our descriptor has better matching performance. As for the 3-D transformation estimation technique, the combinations of the constraints are first analyzed. The performance of different constraints is presented and the best combination is chose. The comparative results with the existing techniques demonstrate that the proposed 3-D transformation estimation technique can obtain better registration accuracy and computation efficiency.**

***Index Terms*—Constraint, local shape descriptor (LSD), point cloud registration, 3-D transformation estimation technique.**

I. Introduction

With the development of the 3-D acquisition techniques, such as laser scanner and space-time cameras, point cloud data are becoming more and more popular, and are the main data source of many applications, including 3-D model or scene reconstruction [1], [2], 3-D object recognition [3], [4], and place recognition [5]. In these applications, point cloud registration is the fundamental problem that needs to be solved. The iterative closest point (ICP) algorithm [6] and its variants [7], [8] are usually applied to perform the fine registration, but all of them need a good initial pose to avoid getting trapped into the local minimum. For the reason, this article focuses on the coarse registration, which can provide a good initial pose for the fine registration.

Manuscript received 20 July 2023; revised 5 August 2023; accepted 10 August 2023. Date of publication 15 August 2023; date of current version 22 August 2023. This work was supported by the National Natural Science Foundation of China under Grant 62162043 and Grant 62271239. *(Corresponding author: Wuyong Tao.)*

The authors are with the School of Mathematics and Computer Sciences, Nanchang University, Nanchang 430031, China (e-mail: taowuyong@ncu.edu.cn; xushaoping@ncu.edu.cn; huangwei@ncu.edu.cn; shufanhu@ncu.edu.cn; mengpang@ncu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3305229

The local shape descriptor (LSD) has been widely used to establish correspondences for point cloud registration, and many scholars have paid much attention to it for a long time. A good LSD should have high descriptiveness, strong robustness, and good computation efficiency. Descriptiveness indicates that the LSD should contain more local surface information, while robustness denotes that the LSD should be less influenced by different nuisances (e.g., noise, point density variation, and partial overlap). The existing LSDs can be classified into binary and float descriptors. Examples of the float descriptors contain the spin images (SI) [9], divisional local feature statistics (DLFS) [10], histogram of distances (HoD) [11], signatures of histograms of orientations (SHOT) [12], spatial and geometric histogram (SGH) [13], kernel density descriptor (KDD) [14], weighted height image (WHI) [15], to name a few. Among these descriptors, the SI, DLFS, and HoD are the descriptors without local reference frame (LRF). They only encode the geometric information, but ignore the spatial information. Therefore, these descriptors generally suffer from limited descriptiveness. The SHOT, SGH, KDD, and WHI are the descriptors with LRF. They first construct an LRF on the local surface around a keypoint. Both of the spatial and geometric information are encoded. Because of the LRF, the descriptors are rotation-invariant. In addition, the LRF provides a manner to fully encode the spatial information. Therefore, these descriptors are highly descriptive in general. As for the binary descriptors, the binary SHOT (B-SHOT) [16], binary HoD (B-HoD) [17], binary shape context (BSC) [18], local voxelized structure (LoVS) [19], and rotational silhouette maps (RSMs) [20] are the widely mentioned descriptors. The B-SHOT, B-HoD, and BSC are obtained by binarizing the corresponding float descriptors. During the process of the binarization, the information is inevitable to be lost. Thus, they have low descriptiveness. The LoVS and RSM are naturally binary descriptors because they are obtained without binarization. Also, the two descriptors have high descriptiveness. In comparison with the float descriptors, the binary descriptors are computationally cheap, and consume less footprint memory. In addition, the binary descriptors are compared using an XOR operation, so descriptor matching will consume less time. The binary descriptor has the significant advantages in the context of robotics and mobile phones.

The correspondences can be established by descriptor matching. A good descriptor can establish more correct correspondences, but the incorrect correspondences inevitably exist as well. How to estimate the accurate rigid transformation under the condition that many incorrect correspondences exist is the

key problem. A large number of 3-D transformation estimation techniques have been proposed. The random sample consensus (RANSAC) algorithm [21] has been applied to randomly select correspondences for point cloud registration. Numerous improved RANSAC algorithms are also proposed, e.g., one-point RANSAC [22], two-point based sample consensus with global constrain (2SAC-GC) [23], compatibility-guided sampling consensus (CG-SAC) [24], and graph enhanced sample consensus (GESAC) [25]. These algorithms improve the computation efficiency using different methods, but they are still very time-consuming. Another kind of methods attempt to remove the false correspondences and cluster the correct ones. Then, the correct correspondences are used to calculate the rigid transformation. Examples contain the geometric consistency (GC) [26], consistency voting (CV) [27], 3-D Hough voting [28], and game theory matching (GTM) [29]. The drawback of these methods is that the incorrect correspondences cannot be fully removed. Some incorrect correspondences are still remained.

Based on the above discussion, a new method is proposed to perform the point cloud registration. First, a binary descriptor is designed to establish the correspondences. The descriptor is a naturally binary one and has high descriptiveness. It encodes the local surface information by comparing the local occupied voxels, so we term it as local occupied voxel comparison (LOVC). In the LoVS, the local occupied voxels are simply encoded. Our descriptor encodes the relationship of the occupied voxels, thus making the descriptor include more local surface information. Then, a novel 3-D transformation estimation technique is proposed. We consider that the one-point RANSAC algorithm selects one correspondence at each iteration. Thus, the constraints cannot be used to skip the incorrect correspondences because the constraints are usually the relationship of at least two correspondences. If one more correspondence is selected, the constraints can be inserted into the computation at each iteration. Therefore, we randomly select two correspondences to estimate the rigid transformation at each iteration. The algorithm is termed as two-point RANSAC with constraints (two-point RANSACWC). "Two-point" means two correspondences. Because the constraints are used to skip the computation when the selected two correspondences are judged as incorrect, the computation efficiency is enhanced. In addition, the transformation calculated by using two correct correspondences is normally more accurate. In summary, the main contributions are as follows:

1) A distinctive binary LSD is designed. It has high descriptiveness and consumes less storage.
2) A novel 3-D transformation estimation technique is proposed. The algorithm is computationally efficient and can achieve more accurate transformation.
3) Based on the binary LSD and 3-D transformation estimation technique, a registration method is presented. Because of the advantages of the binary LSD and 3-D transformation estimation technique, the registration method also has good computation efficiency and high registration accuracy.

The rest of the article is organized as follows. Section II gives the review of the LSD and 3-D transformation estimation technique. The proposed LSD is presented in Section III in detail. In Section IV, the pipeline of point cloud registration is given and two-point RANSACWC algorithm is described in detail. Section V presents the experiments and analysis. Finally, Section VI concludes this article.

II. Related Work

A. Local Shape Descriptor

The float descriptors are first reviewed. The SI [9] is an early proposed descriptor. In the descriptor, two parameters are calculated for each neighboring point. The 2-D grids are generated according to the two parameters. The number of the points falling in a grid is the value of the grid. The DLFS [10] divides the local neighborhood into several subspaces along the radial direction. For each subspace, three angle features and local height are used to generate four histograms. The four weighted histograms are concatenated to form the final histogram. Although the descriptor does not construct the LRF, it encodes part of the spatial information. The HoD [11] chooses one border point as a reference point. The distance between the reference point and each neighboring point is calculated, which is then normalized. The normalized distances are used to form a histogram. The descriptor even does not calculate a local reference axis (LRA). The main drawback of the above three descriptors is that they do not fully encode the spatial information. In the SHOT [12] descriptor, an LRF method is first proposed. This is the first method that removes the ambiguity of all axes. The descriptor partitions the local neighborhood into subspaces along the radial, azimuth, and elevation directions. For each subspace, a histogram is generated according to the consine values of the deviation angle between the normal vectors of the neighbors in the subspace and the z-axis of the LRF. The descriptor only has 32 subspaces, which results in low distinctiveness. Tombari et al. [30] inserted a unique LRF into the 3-D shape context descriptor [31]; the unique shape context (USC) descriptor was proposed. The USC descriptor evenly divides the local neighborhood along the azimuth and elevation dimensions and logarithmically divides the local neighborhood along the radial dimension. The irregular division makes the descriptor sensitive to point density variation. The rotational projection statistics (RoPS) [32] applies multiview mechanism so as to encode more information. The multiple views are obtained by rotating the local point cloud. Each of the rotated point clouds is projected on the three planes of the LRF. Then, three distribution matrices are generated. The statistics of all the distribution matrices form the feature vector. The triple orthogonal local depth images (TOLDI) [33] projects the local point cloud on three planes and calculates three local depth images. The pixel values of the local depth images constitute the feature vector. The triple local coordinate images (TriLCI) [3] projects the local point cloud on the three planes and calculates three local coordinate images. Due to the 3-D-to-2-D projection, the RoPS, TOLDI, and TriLCI descriptors lost some of the local surface information [34]. In the SGH descriptor [13], the local neighborhood is evenly divided into subspaces along radial, azimuth, and elevation dimensions. For each subspace, two deviation angles are calculated for each neighboring point.

Two histograms are accordingly generated. The computation complexity of the descriptor is high. In the KDD descriptor [14], a local cubic volume is used to enclose the local point cloud. The local cubic volume is divided into voxels with the same size. A kernel density is calculated for each voxel. In the WHI descriptor [15], a simplified LRF is first constructed. The local points are projected on the *xy* plane and an image is generated. The pixel value of the image is the mean weighted heights. The dual SI [35] is an improvement to the SI descriptor. It calculates two SIs for the keypoint and one of its neighbors, which are concatenated to form the whole feature descriptor. Furthermore, the multiscale strategy is used to alleviate effect of occlusion. The multiview depth and contour signatures [36] apply multiple attributions for better encoding the local surface information. The multiview strategy is also applied to make the descriptor include more information. Yang et al. [4] proposed the voxelization in invariant distance space (VOID) descriptor. It projected the neighboring points into the 3-D space defined by one point-to-point distance and two point-to-plane distances. Because the three distances are invariant to normal-sign-ambiguity (NSA), the descriptor is also invariant to the NSA. The NSA is an unsolved problem in the existing LSDs. The VOID descriptor was obtained by voxelizing the defined 3-D space and concatenating all the voxel values.

Then, the binary descriptors are reviewed. The B-SHOT [16] is the first 3-D binary descriptor, which is the binary version of the SHOT descriptor. It successively takes four values from the SHOT feature vector. For each quaternion, five cases are designed to transform them into bit string. The B-HoD [17] is an extension of the HoD descriptor. It transforms the float values into binary values via the binary-coded decimal scheme. The BSC [18] projects the local neighboring points on the three planes of the LRF. For each plane, the 2-D grids are generated, and the weighted projection density and distance are used to calculate the grid values. Then, some pairs of the grid values are randomly selected and keep unchanged in subsequent selections. The difference test is performed to obtain the bit string. Zou et al. [37] proposed the binary rotational projection histogram descriptor, which used point density and depth to encode the local surface information. The center-symmetric local binary patterns was applied to transform the float values into binary values. These binary descriptors are obtained by performing binarization on the corresponding float descriptors. During the process of the binarization, some information is inevitably lost. Therefore, these binary descriptors have relatively low descriptiveness. Different from these descriptors, the LoVS [19] and RSM [20] descriptors are straightforward. The two descriptors obtain the binary feature without binarization. The LoVS encloses the local point cloud with a cubical volume. Then, the cubical volume is divided into uniform voxels. If some points fall in a voxel, the value of the voxel is 1, otherwise 0. The values of all voxels are concatenated to form the binary feature vector. The RSM rotates the local point cloud for multiple times. Each rotated point cloud is projected on the *xy* plane of the LRF. Then, a silhouette image is generated by judging whether some points fall in a grid. A binary string is calculated from the silhouette image. The binary strings of all the silhouette images are concatenated to get the binary feature vector.

B. 3-D Transformation Estimation Technique

In [38], the state-of-the-art 3-D transformation estimation techniques are categorized into two classifications: maximum-consistency-based and confidence-verify-based methods. The maximum-consistency-based methods struggle to cluster the correct correspondences and remove false correspondences from the initial correspondence set. Then, these correct correspondences are used to calculate the rigid transformation. The GC [26] leverages the geometric distance constraint to cluster correct correspondences. If two correspondences are correct, they will satisfy the constraint with each other. With the constraint, each correspondence finds the others that are compatible with it, and thus form a cluster. The maximum cluster is treated as the one that is composed of correct correspondences. However, due to inherent ambiguity, many incorrect correspondences are still remained. The CV [27] first selects the top-k correspondences according to the nearest neighbor similarity ratio (NNSR) scores. The compatibility measure between two correspondences is calculated using the rigidity and LRF affinity constraints. The compatibility score of a correspondence is defined as the summary of the compatibility measures between the correspondence and others. The method is sensitive to the parameter setting. It has high precision performance but pretty poor recall performance. In [3], a clustering method is proposed to find out the correct correspondences. The method uses the LRFs to compute an auxiliary point for each correspondence. The auxiliary points computed by correct correspondences will cluster together. In the method, the errors of the LRFs largely affect the performance of the clustering. The GTM [29] constructs a payoff matrix. An element of the payoff matrix is the degree of compatibility between two correspondences. In [39], a variant of the GTM is developed. The geometric constraint with an exponential form is applied to calculate the payoff matrix. However, these methods cannot find out all the correct correspondences and some false correspondences are still remained. The GTM is sensitive to symmetric and similar patch. By contrast, the confidence-verify-based methods usually calculate some plausible transformations and verify the confidence levels of the transformations. The RANSAC [21] has been widely used to calculate the rigid transformation for 3-D rigid data matching. Three correspondences are randomly selected to compute a transformation and the number of inliers is counted at each iteration. The transformation corresponding to the maximum number of inliers is the optimal one. The consistent correspondence verification [32] first calculates a transformation for each correspondence by the LRFs of the keypoints. Then, these transformations are clustered to generate a consistent set for each correspondence. Afterward, the transformation for each correspondence is calculated again by all correspondences of the consistent set. Finally, these transformations are verified by the number of inliers. The one-point RANSAC [22] leverages the LRFs of two keypoints in a correspondence to compute the rotation matrix, so only one correspondence is enough to estimate the transformation. For the method, the errors of the LRFs negatively influence the accuracy of the estimated transformation. In the 2SAC-GC [23], the LRAs of the keypoints are applied to compute the transformation, so

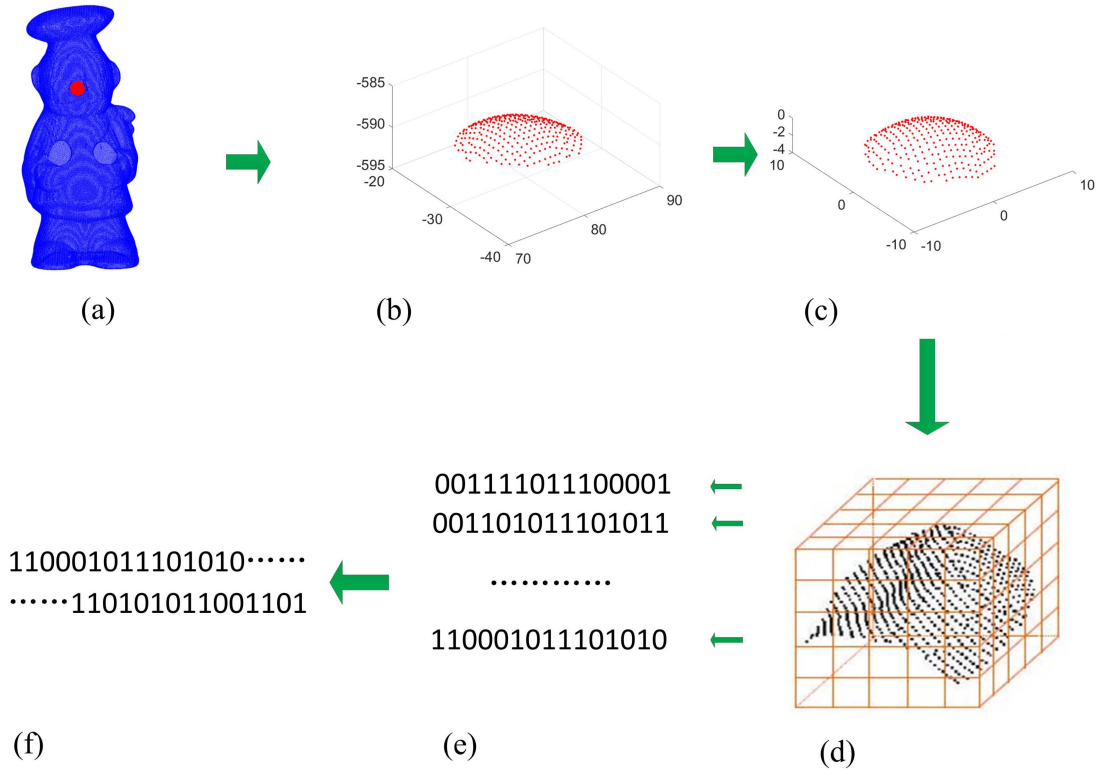


Fig. 1. Overflow of computing the LOVC descriptor. (a) Point cloud and local surface around keypoint. (b) Local point cloud is cropped from point cloud. (c) Local point cloud is transformed with respect to LRF. (d) Transformed local point cloud is voxelized. (e) Bit strings are generated. (f) Bit strings are concatenated to form binary feature vector. (a) 'Chef' model, (b) cropped local point cloud, (c) transformed local point cloud, (d) voxelization, (e) bit strings, (f) binary feature vector.

two correspondences are selected at each iteration. In addition, two constraints are used to facilitate the computation process. If the two correspondences do not satisfy the two constraints, the iteration will be skipped. The CG-SAC [24] calculates a compatibility score for each pair of correspondences using the rigidity and distance between salient points constraints. Then, the all pairs of correspondences are sorted according to the compatibility scores. The top-ranked correspondence pairs are selected to calculate the transformation. The GESAC [25] randomly selects a large subset (32 correspondences) from the initial correspondences at each iteration. A graph is used to express the subset and the false correspondences are removed by a graph matching algorithm. In [40], the reliability of a triplet of correspondences is calculated using the projection distances of the reference point to the normal vectors of the three source keypoints. All the correspondence triplets are ranked according to the reliability. The top-ranked triplets are selected to compute the transformations.

III. Proposed Local Shape Descriptor

The proposed descriptor (i.e., the LOVC descriptor) is presented in detail in this section. The keypoints are extracted from a point cloud. For a keypoint, the local point cloud is cropped from the point cloud with a sphere of radius r centered at the keypoint. The LRF method in [3] is applied to calculate the LRF for the local point cloud. The method first uses a few of neighboring points to calculate the normal vector. The z-axis is determined by disambiguating the normal vector. Then, the vectors from the keypoint to all neighboring points are projected on the tangent plane. The x-axis is calculated by integrating all the projection vectors. The y-axis is calculated as the cross-product of the x-axis and z-axis. Then, the local point cloud is transformed with respect to the LRF. This makes the descriptor invariant to rotation and translation. The entire computation process is shown in Fig. 1.

A cubical volume is used to enclose the transformed local point cloud. Because the radius of the neighborhood is r, the length of the sizes of the cubical volume is $2r$. The cubical volume is divided into voxels with same size. As shown in Fig. 2, each layer of voxels is used to calculate a bit string. The LoVS only encodes the local surface information by judging whether a voxel is occupied or not. In our descriptor, the relationship between the occupied voxels is taken into consideration. Thus, more local surface information can be included. For two symmetric voxels (e.g., the two black voxels in Fig. 2), the feature value is calculated as

$$f(v_{ij}, v_{ji}) = \begin{cases} 0 & \text{if}(\left|Q_{v_{ij}}\right| > 0 \text{ and } \left|Q_{v_{ji}}\right| > 0) \\ & \quad \text{or } (\left|Q_{v_{ij}}\right| = 0 \text{ and } \left|Q_{v_{ji}}\right| = 0) \\ 1 & \quad \text{otherwise} \end{cases} \tag{1}$$

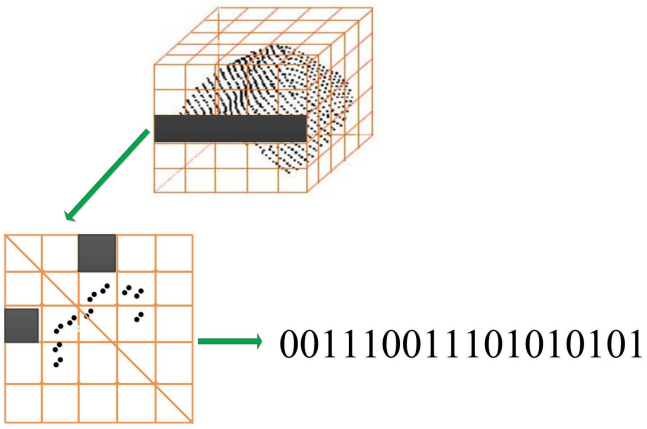


Fig. 2. Computation of bit string for one layer of voxels.

where $|\cdot|$ denotes the cardinality of a point set, v_{ij} and v_{ji} are the symmetric voxels, and $Q_{v_{ij}}$ denotes the point set in the voxel. The voxels on the main diagonal are ignored because they do not have symmetric voxels. The feature values of all pairs of the symmetric voxels are concatenated to form the bit string (see example in Fig. 2). Then, the bit strings of all layers are concatenated to get the binary feature vector, i.e., the LOVC descriptor.

If the cubical volume is divided into $w \times w \times w$ voxels, the length of a bit string is $w \times (w-1)/2$. Hence, the length of the binary feature vector is $w \times w \times (w-1)/2$, i.e., the dimensionality of the descriptor. The storage of the descriptor is $w \times w \times (w-1)/2$bit. We set the value of w as an odd number in order to resist point-level perturbations [18]. The value will be determined by experiments later.

IV. Point Cloud Registration Based on the LOVC Descriptor and Two-Point RANSACWC

In this section, the LOVC descriptor is used to perform the point cloud registration. The computation process of point cloud registration is shown in Fig. 3. In the registration method, the LOVC descriptor and two-point RANSACWC algorithm are two main contributions. This article focuses on the coarse registration of point clouds. The ICP fine registration is out of the scope of this article, so it is enclosed in a dash box. In our experiments, the step is not performed.

A. Correspondence Establishment

Giving two point clouds (source and target point clouds), the keypoints are first extracted by the 3-D Harris detector [41] because it is easy to implement and is computationally efficient. The LOVC descriptor is calculated for each keypoint. The two calculated descriptor sets for the source and target keypoints are, respectively, denoted as $\{\boldsymbol{f}_1^s \quad \boldsymbol{f}_2^s \quad \dots \quad \boldsymbol{f}_n^s\}$ and $\{\boldsymbol{f}_1^t \quad \boldsymbol{f}_2^t \quad \dots \quad \boldsymbol{f}_m^t\}$. The correspondences are established by the NNSR [2]. For each descriptor $\boldsymbol{f}_i^s$, the nearest descriptor $\boldsymbol{f}_i^t$ and the second nearest descriptor $\boldsymbol{f}_j^t$ are searched. If the distance ratio

$$\left\|\boldsymbol{f}_i^s - \boldsymbol{f}_i^t\right\| / \left\|\boldsymbol{f}_i^s - \boldsymbol{f}_j^t\right\| < \lambda \tag{2}$$

the two keypoints associated to $\boldsymbol{f}_i^s$ and $\boldsymbol{f}_i^t$ are considered as a correspondence. $\|\cdot\|$ denotes the Hamming distance and the threshold λ is usually set as 0.9 [3]. Because the calculated descriptors are influenced by the noise, point density variation, partial overlap, and so on, the correspondences established by descriptor matching are not all right. We need to search for the correct correspondences and compute the rigid transformation.

B. Transformation Estimation via Two-Point RANSACWC

The one-point RANSAC [22] only applies one correspondence to calculate the transformation, so the constraints cannot be inserted into the method. This is because the constraints are the relationship of at least two correspondences. However, many constraints have been developed and proven to be useful to accelerate the searching process of correct correspondences. Now, we have obtained the initial correspondence set. The two-point RANSACWC randomly selects two correspondences to estimate the transformation at each iteration. Thus, the constraints can be inserted. Different constraints are researched in this article. The considered constraints are as follows:

1) *Geometric distance constraint (GC)* [26]. Giving two correspondences $\boldsymbol{c}_i = (\boldsymbol{p}_i, \boldsymbol{q}_i)$ and $\boldsymbol{c}_j = (\boldsymbol{p}_j, \boldsymbol{q}_j)$, the GC is defined as

$$\text{abs}(d(\boldsymbol{p}_i, \boldsymbol{p}_j) - d(\boldsymbol{q}_i, \boldsymbol{q}_j)) < \delta_{\text{gc}} \tag{3}$$

where $\text{abs}(\cdot)$ denotes absolute value, $\boldsymbol{p}_i, \boldsymbol{p}_j$ are the two source keypoints, $\boldsymbol{q}_i, \boldsymbol{q}_j$ are the two target keypoints, and d denotes the distance between two keypoints. δ_{gc} is a threshold and is generally set as $2pr$ (pr denotes point cloud resolution, i.e., the average value of the distances between all points and their nearest points). If the two correspondences are correct, they will satisfy the constraint.

2) *Angular constraint based on local reference axes (ACLRA)* [23]. The constraint is defined as

$$\begin{aligned}&\text{abs}(\mathbf{arccos}(\text{LRA}(\boldsymbol{p}_i) \cdot \text{LRA}(\boldsymbol{p}_j))\frac{180}{\pi} \\ &- \mathbf{arccos}\left(\text{LRA}(\boldsymbol{q}_i) \cdot \text{LRA}(\boldsymbol{q}_j)\right)\frac{180}{\pi}) < \delta_{aca}\end{aligned} \tag{4}$$

where LRA denotes LRA of a keypoint, and $\mathbf{arccos}(\cdot)$ denotes inverse cosine function. δ_{aca} is a threshold and is generally set as 10°. When computing the LOVC descriptor, the LRFs have been calculated. The LRA is the z-axis of the LRF. Readers can refer to [3] for the computation of the LRA. Therefore, we just need to reuse the LRAs when using the constraint.

3) *Angular constraint based on local reference frame (ACLRF).* Similar to LRA, the LRF can also be used to construct an angular constraint. Therefore, a new angular constraint based on LRF is proposed, which is defined as

$$\text{abs}\left(\mathbf{arccos}\left(\frac{\mathbf{trace}(\boldsymbol{V}_{\boldsymbol{p}_i}\boldsymbol{V}_{\boldsymbol{p}_j}^{-1}) - 1}{2}\right)\frac{180}{\pi}\right.$$

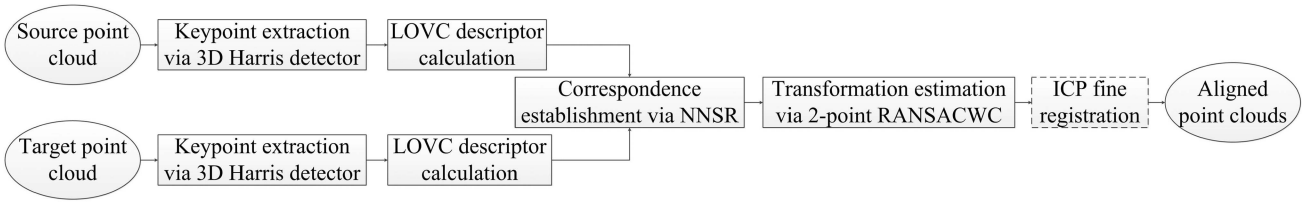


Fig. 3. Workflow of point cloud registration.

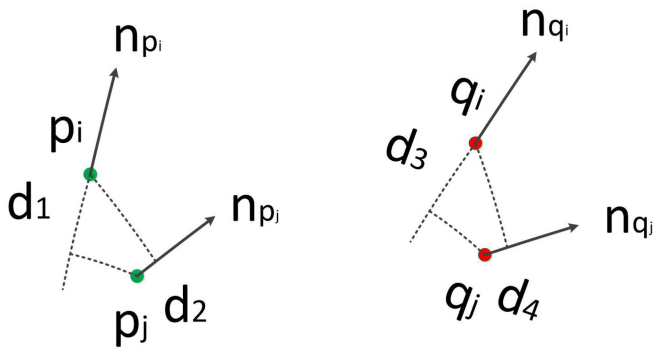


Fig. 4. Schematic diagram of PDKN.

$$-\arccos\left(\frac{\mathbf{trace}(\boldsymbol{V}_{\boldsymbol{q}_i}\boldsymbol{V}_{\boldsymbol{q}_j}^{-1})-1}{2}\right)\frac{180}{\pi}<\delta_{acf} \tag{5}$$

where $\mathbf{trace}(\cdot)$ is the trace of a matrix and $\boldsymbol{V}$ denotes the LRF of a keypoint. δ_{acf} is a threshold and is set as 10°. Because the LRFs have been calculated, we just need to reuse them.

4) *Projection distance of keypoint to normal vector (PDKN).* Based on the PDKN, we propose another new constraint.

As shown in Fig. 4, $\boldsymbol{p}_i, \boldsymbol{p}_j$ are the two source keypoints and $\boldsymbol{q}_i, \boldsymbol{q}_j$ are the two target keypoints. $\boldsymbol{n}_{\boldsymbol{p}_i}, \boldsymbol{n}_{\boldsymbol{p}_j}$ are the normal vectors of the two source keypoints and $\boldsymbol{n}_{\boldsymbol{q}_i}, \boldsymbol{n}_{\boldsymbol{q}_j}$ are the normal vectors of the two target keypoints. The projection distances are calculated as

$$\begin{aligned} d_1 &= \text{abs}(\boldsymbol{n}_{\boldsymbol{p}_i}(\boldsymbol{p}_j-\boldsymbol{p}_i)) \\ d_2 &= \text{abs}(\boldsymbol{n}_{\boldsymbol{p}_j}(\boldsymbol{p}_i-\boldsymbol{p}_j)) \\ d_3 &= \text{abs}(\boldsymbol{n}_{q_i}(\boldsymbol{q}_j-\boldsymbol{q}_i)) \\ d_4 &= \text{abs}(\boldsymbol{n}_{q_j}(\boldsymbol{q}_i-\boldsymbol{q}_j)). \end{aligned} \tag{6}$$

In principle, the projection distances are invariant to rotation and translation. Hence, the constraint is defined as

$$\text{abs}(d_1-d_3)<\delta_{pd},\ \text{abs}(d_2-d_4)<\delta_{pd}. \tag{7}$$

The threshold δ_{pd} is set as $2pr$.

These constraints are grouped into several combinations. We compare the performance of different combinations. Taking the ACLRA, ACLRF, or PDKN as a single constraint into a combination gets rather poor performance, so we did not do this. All considered combinations are listed in Table I.

TABLE I
COMBINATIONS OF CONSTRAINTS

Combination	Constraint
C1	GC
C2	GC, PDKN
C3	GC, ACLRF
C4	GC, ACLRA
C5	GC, PDKN, ACLRF
C6	GC, PDKN, ACLRA
C7	GC, ACLRF, ACLRA
C8	GC, PDKN, ACLRF, ACLRA

Algorithm 1: Two-Point RANSACWC Algorithm.

Input: The initial correspondence set
$\boldsymbol{C}=\{\boldsymbol{c}_1 \quad \boldsymbol{c}_2 \quad \ldots \quad \boldsymbol{c}_n\}$
Output: The optimal transformation
L is the maximum iteration number
Set the initial number of inliers m_0 as zero
For $i=1$ to L **do**
  Two correspondences are randomly selected.
  If the two correspondences satisfy one combination
    Calculate the transformation
    The source point cloud is transformed according to the calculated transformation
    Count the number of the points in the overlapped area between the transformed source point cloud and target point cloud, which is treated as the inlier number m.
    If $m > m_0$
      The current transformation is the optimal one
      $m_0 = m$
    End if
  End if
End for

The detailed computation steps of the proposed two-point RANSACWC algorithm are presented in Algorithm 1:

In Algorithm 1, when calculating the transformation, the keypoints $\boldsymbol{p}_i, \boldsymbol{p}_j, \boldsymbol{q}_i, \boldsymbol{q}_j$ and their normal vectors $\boldsymbol{n}_{\boldsymbol{p}_i}, \boldsymbol{n}_{\boldsymbol{p}_j}, \boldsymbol{n}_{\boldsymbol{q}_i}, \boldsymbol{n}_{\boldsymbol{q}_j}$ are applied. First, we calculate the following four points:

$$\begin{aligned} \boldsymbol{p'}_i &= \boldsymbol{p}_i + \boldsymbol{n}_{\boldsymbol{p}_i} \\ \boldsymbol{p'}_j &= \boldsymbol{p}_j + \boldsymbol{n}_{\boldsymbol{p}_j} \\ \boldsymbol{q'}_i &= \boldsymbol{q}_i + \boldsymbol{n}_{\boldsymbol{q}_i} \\ \boldsymbol{q'}_j &= \boldsymbol{q}_j + \boldsymbol{n}_{\boldsymbol{q}_j}. \end{aligned} \tag{8}$$

Then, the singular value decomposition (SVD) [42] is applied to compute the transformation. The following matrices are constructed:

$$\boldsymbol{A}_{4\times 3}=\begin{bmatrix}\boldsymbol{p}_i\\ \boldsymbol{p}_j\\ \boldsymbol{p'}_i\\ \boldsymbol{p'}_j\end{bmatrix},\ \boldsymbol{Y}_{4\times 3}=\begin{bmatrix}\boldsymbol{q}_i\\ \boldsymbol{q}_j\\ \boldsymbol{q'}_i\\ \boldsymbol{q'}_j\end{bmatrix}. \tag{9}$$

Then, the covariance matrix is calculated as

$$\begin{aligned}\boldsymbol{H}_{3\times 3}=&\left[\boldsymbol{A}_{4\times 3}-[1\quad 1\quad 1\quad 1]^T\bar{\boldsymbol{A}}_{1\times 3}\right]^T\\ &\times;\left[\boldsymbol{Y}-[1\quad 1\quad 1\quad 1]^T\bar{\boldsymbol{Y}}_{1\times 3}\right]\end{aligned} \tag{10}$$

where $\bar{\boldsymbol{A}}_{1\times 3}$ is the mean value of $\boldsymbol{A}_{4\times 3}$ and $\bar{\boldsymbol{Y}}_{1\times 3}$ is the mean value of $\boldsymbol{Y}_{4\times 3}$. Performing SVD on matrix $\boldsymbol{H}_{3\times 3}$

$$\boldsymbol{USV}^T=\text{svd}(\boldsymbol{H}_{3\times 3}). \tag{11}$$

Thus, the rotation matrix is

$$\boldsymbol{R}=\boldsymbol{VDU}^T \tag{12}$$

where $\boldsymbol{D}=\text{diag}(1,1,\det(\boldsymbol{UV}^T))$, $\text{diag}(\cdot)$ denotes a diagonal matrix, and $\mathbf{det}(\cdot)$ denotes the determinant of a matrix. The translation vector is

$$\boldsymbol{t}=\bar{\boldsymbol{q}}-\bar{\boldsymbol{p}}\boldsymbol{R}^T \tag{13}$$

where $\bar{\boldsymbol{q}}$ is the mean value of $\boldsymbol{q}_i$ and $\boldsymbol{q}_j$, and $\bar{\boldsymbol{p}}$ is the mean value of $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$. The points $\boldsymbol{p'}_i$, $\boldsymbol{p'}_j$, $\boldsymbol{q'}_i$, and $\boldsymbol{q'}_j$ are calculated using the normal vectors, so they are inevitably contaminated by the errors of the normal vectors. In order not to propagate the errors of the normal vectors to the translation vector, $\boldsymbol{p'}_i$, $\boldsymbol{p'}_j$, $\boldsymbol{q'}_i$, and $\boldsymbol{q'}_j$ are not applied for (13).

When counting the inlier number, we can also use the simplified point clouds to calculate the inlier number for decreasing the computation time. This is a generally applied strategy. This article mainly studies these combinations, so we do not use the strategy.

V. Experiments and Analysis

The experiments are performed to evaluate the LOVC descriptor and two-point RANSACWC algorithm. The performance of the LOVC descriptor is first assessed. Then, the two-point RANSACWC algorithm is analyzed.

A. Datasets

The BR [43], space time [43], UWA3M [44], and UWAOR [44] datasets are used to evaluate the performance of the LOVC descriptor. The BR dataset contains six model point clouds without noise from the Sanford 3-D Scanning Repository. The model point clouds are transformed to simulate six scene point clouds by setting a true transformation. The Gaussian noise with standard deviation of $0.3pr$ is injected into the scene point clouds and the downsampling is performed on the scene point clouds with the sampling rate of 7/10. The aim is to simulate the scene point cloud influenced by Gaussian noise and point density variation. The space time dataset is composed of eight model point clouds and fifteen real scene point clouds, which are obtained by the Space Time Stereo technology. The dataset is mainly influenced by clutter, occlusion, noise, and outliers. The UWA3M and UWAOR datasets are acquired by a Minolta Vivid 910 scanner. The two datasets are high-quality. The UWA3M dataset is mainly affected by partial overlap, while the UWAOR dataset is mainly affected by clutter and occlusion. The UWA3M dataset contains 496 pairs of point clouds and the UWAOR dataset contains five model point clouds and fifty scene point clouds. The true transformations of the space time, UWA3M, and UWAOR datasets are provided by the publishers. Fig. 5 exhibits the examples of the point clouds of the four datasets.

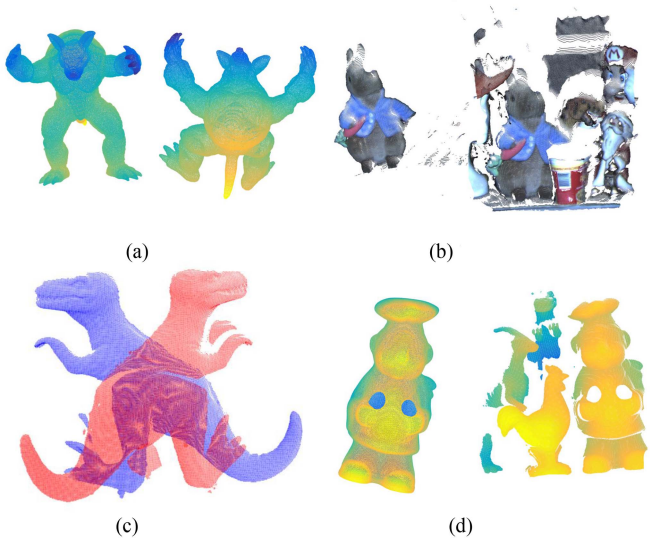


Fig. 5. Examples of point clouds of four datasets. (a) BR. (b) Space Time. (c) UWA3M. (d) UWAOR.

An indoor dataset [45] and an outdoor dataset are applied to analyze the two-point RANSACWC algorithm. The indoor dataset is obtained by an FARO Focus 3-D X330 HDR scanner. It includes the point clouds of five scenes (apartment, bedroom, boardroom, lobby, and loft). We choose two pairs of point clouds from the apartment and boardroom scenes to perform the experiments. The outdoor dataset contains the point clouds of the two city scenes (City 1 and City 2). The point clouds of the City 1 scene [46] are obtained by a Leica C10 laser scanner. The point clouds of the City 2 scene are from the Robotic 3-D Scan Repository,[1] which are obtained by a Riegl VZ-400 laser scanner. We also choose two pairs of point clouds from the two scenes, respectively. The true transformations of the two datasets are obtained by manually coarse registration and ICP fine registration. The example can be found in Figs. 9 and 10.

B. Performance of the LOVC Descriptor

1) Parameter Setting: There are two parameters that need to be determined, which are, respectively, the neighborhood radius $r$and voxel resolution w. The neighborhood radius r is generally set as $15pr$. Therefore, we only need to determine the

[1]http://kos.informatik.uni-osnabrueck.de/3Dscans/

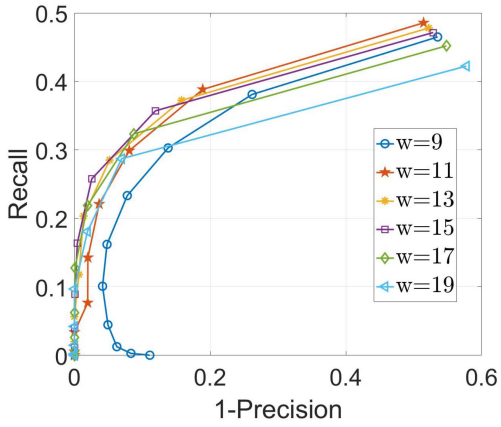


Fig. 6. RP curve under different values of w.

value of w. If the value of w is too small, the descriptor will be sensitive to noise and point density variation. By contrast, if the value of w is too large, many details will be lost. This makes the descriptor include less local surface information, decreasing the descriptiveness of the descriptor. The descriptor matching performance is assessed by the recall versus 1-precision (RP) curve. By varying the threshold λ of the NNSR from 0.1 to 1, the recall and one-precision are calculated at each threshold

$$\text{Recall} = \frac{\text{Number of correct correspondences}}{\text{Total number of true correspondences}} \tag{14}$$

$$1 - \text{Precision} = \frac{\text{Number of incorrect correspondences}}{\text{Total number of established correspondences}}. \tag{15}$$

If distance between the transformed source keypoint by the true transformation and its corresponding target keypoint is smaller than $0.5r$, the correspondence is considered as correct. If the descriptor has both the high recall and the high precision, the RP curve will fall in the left-top of the plot.

The RP curve under different values of w is shown in Fig. 6. The experiments are performed on the BR dataset. As we can see, when the value of w increases from 9 to 15, the descriptor matching performance also increases. As the value of w continues to increase, the performance begins to decline. Therefore, we set the value of w as 15.

2) Comparison With Existing Descriptors: The LOVC descriptor is compared with the SI, USC, RoPS, TOLDI, and LoVS descriptors. In these descriptors, the SI, USC, RoPS, and TOLDI are the float-valued descriptors, while the LoVS and LOVC are the binary descriptors. As suggested by [34], all the LRF-based descriptors apply the same LRF method (i.e., the LRF method in [3]) for only comparing the encoding method of local surface information. The information of the descriptors are listed in Table II.

The space time, UWA3M, and UWAOR datasets are used to perform the comparative experiments. We extract 1000 keypoints from the source point cloud, and transform these keypoints to the target point cloud by the true transformation. Then,

TABLE II
PARAMETER SETTING OF SIX DESCRIPTORS

Descriptor	Neighborhood radius (pr)	Dimension	Storage (bit)
SI	15	15×15=225 (float)	225×8
USC	15	15×12×11=1980 (float)	1980×8
RoPS	15	3×3×5×3=135 (float)	135×8
TOLDI	15	3×20×20=1200 (float)	1200×8
LoVS	15	9×9×9=729 (bit)	729×1
LOVC	15	15×15×7=1575 (bit)	1575×1

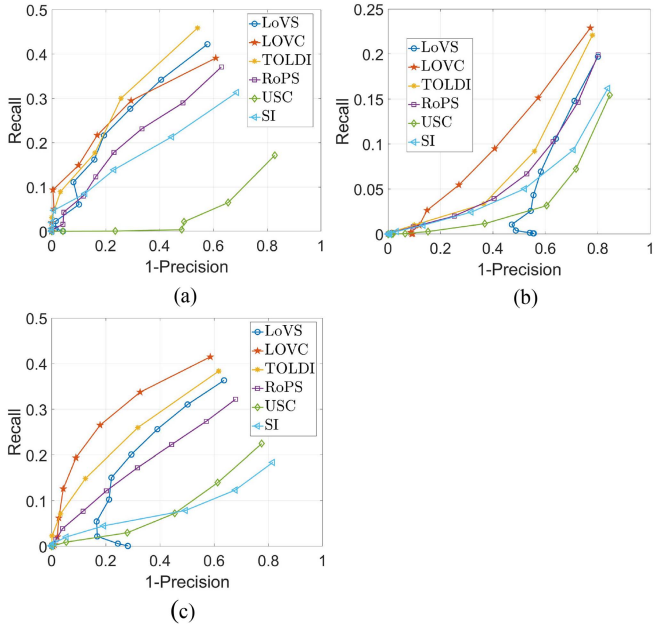


Fig. 7. RP curve of six descriptors on three datasets. (a) Space Time. (b) UWA3M. (c) UWAOR.

the closest point of each transformed keypoint is searched from the target point cloud. These closest points are treated as the corresponding keypoints in the target point cloud, obtaining the true correspondences. For the incomplete point clouds, the descriptors are only calculated on the keypoints in the overlapped area. The average RP curve of all point cloud pairs in a dataset is shown in Fig. 7. We can see that on the space time dataset, the TOLDI descriptor gets the best matching performance. The LOVC is the second best descriptor, followed by the LoVS. The RoPS, USC, and SI achieve relative poor matching performance. On the UWA3M dataset, the LOVC gets significantly better matching performance than others. The second best descriptor is TOLDI. The RoPS and LoVS obtain moderate matching performance. On the UWAOR dataset, our descriptor is still better than the others by a large margin. The TOLDI gets the second best matching performance, followed by LoVS and RoPS.

The SI is an LRA-based descriptor. It only encodes the geometric information, but ignores the spatial information. Therefore, the descriptor has poor descriptiveness. The USC irregularly divides the local neighborhood, which increases its sensitivity to noise and point density variation. The RoPS applies

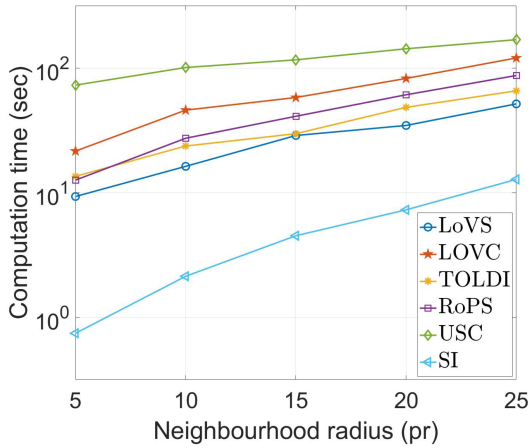


Fig. 8. Time efficiency of six descriptors. The y-axis is logarithmic axis for the best view.

the 2-D point distribution to encode the local surface information, while the TOLDI applies the local depth. The local depth is better to preserve the information than the 2-D point distribution, so the TOLDI achieves better matching performance than the RoPS. The LoVS applies a straightforward manner to encode the local surface information. Although the encoding manner is simple, it has good descriptiveness and thus gets relatively good matching performance. However, the descriptor does not take the relationship of the voxels into consideration. In order to better preserve the local surface information, the LOVC considers the relationship of the voxels and therefore achieves the best matching performance.

The time efficiency of the six descriptors is presented in Fig. 8. The neighborhood radius is varied from 5*pr* to 25*pr*. The time of computing the descriptors for 1000 keypoints is recorded. It can be seen that the USC has the poorest computation efficiency. This is because the descriptor has high dimensionality. The SI has the best computation efficiency, but note that it has very poor descriptor matching performance. All of the RoPS, TOLDI, and LoVS have good computation efficiency. The LOVC is inferior to the RoPS, TOLDI, and LoVS. Because the LOVC needs to compare the voxels and calculate more feature values, its computation time is more than that of the LoVS. However, the increase of computation time is not very much.

C. Analysis of Two-Point RANSACWC Algorithm

The performance of different combinations is discussed in the section. The indoor and outdoor datasets are used to perform the experiments. The performance is discussed from two aspects: registration accuracy and computation efficiency. The registration accuracy is measured by rotation error and translation error, which are, respectively, defined as

$$\text{error}_R = \mathbf{arccos}\left(\frac{\mathbf{trace}(\boldsymbol{R}_{\text{true}}\boldsymbol{R}^{-1}) - 1}{2}\right)\frac{180}{\pi} \tag{16}$$

$$\text{error}_t = \|\boldsymbol{t}_{\text{true}} - \boldsymbol{t}\| \tag{17}$$

where $\boldsymbol{R}_{\text{true}}$ is the true rotation, $\boldsymbol{R}$ is the estimated rotation, $\boldsymbol{t}_{\text{true}}$ is the true translation, and $\boldsymbol{t}$ is the estimated translation.

The proposed registration method is employed to align each pair of point clouds. Owing to the randomness of the two-point RANSACWC algorithm, we run the algorithm for 100 times. The mean rotation error, mean translation error, and mean computation time are recorded. The computation time does not include the time spent on the descriptor calculation. The iteration number of the two-point RANSACWC algorithm is set as 2000. Different combinations of the constraints are used in the two-point RANSACWC algorithm. The registration results are similar, so we only exhibit the registration result of one combination for each scene in Figs. 9 and 10. As we can see, the four pairs of point clouds have been successfully registered together. All the combinations can get enough good registration results, which can provide a good initial pose for the fine registration.

The mean rotation error, mean translation error, and mean computation time are listed in Table III. We can see that all of the combinations get the relatively low rotation errors and translation errors. It is observed that the rotation error and translation error of C1 are the same to those of C4. This phenomenon also happens on combination pairs (C2, C6), (C3, C7), and (C5, C8). Compared with C1, C4 has an extra ACLRA. This means that the ACLRA is not easy to reject correct correspondences. However, the ACLRA can reject incorrect correspondences because the computation time of C4 is less than C1. The computation time of C2 and C3 is smaller than that of C4. This is because some correct correspondences are also rejected by the PDKN and ACLRF. More correspondences are rejected, so the computation time is less. The rotation error and translation error of C2 and C3 are different from C1. This also indicates that some correct correspondences are rejected by the PDKN and ACLRF in comparison with the GC. Therefore, we can conclude that the ACLRA can reject incorrect correspondences but better preserve the correct correspondences. The PDKN and ACLRF can reduce more computation time but at expense of rejecting correct correspondences.

As more and more constraints are applied to reject correspondences, the computation time is generally reduced. However, the effect is decreasing when more constraints are added. Maybe four constraints are already enough. It is unnecessary to add more constraints. Therefore, we think that C8 is the best combination. Furthermore, C8 can also get enough good coarse registration results, which provide good initial pose for fine registration. *Thus, the proposed 2-point RANSACWC algorithm employs the combination C8.* We observe that C5 and C8 have similar computation time. This indicates that the effect of the ACLRA is slight. This is because most of false correspondences have been rejected by the PDKN and ACLRF. An exception happens on the City 2 scene. On this scene, C3 spends more time than C1, and C5 spends more time than C2. Therefore, the ACLRF is useless to reduce computation time. On the contrary, because of the calculation of the ACLRF, more computation time is spent. This also explains why the C8 spends more time than C6 on City 2 scene. In addition, on the City 1 scene, the computation

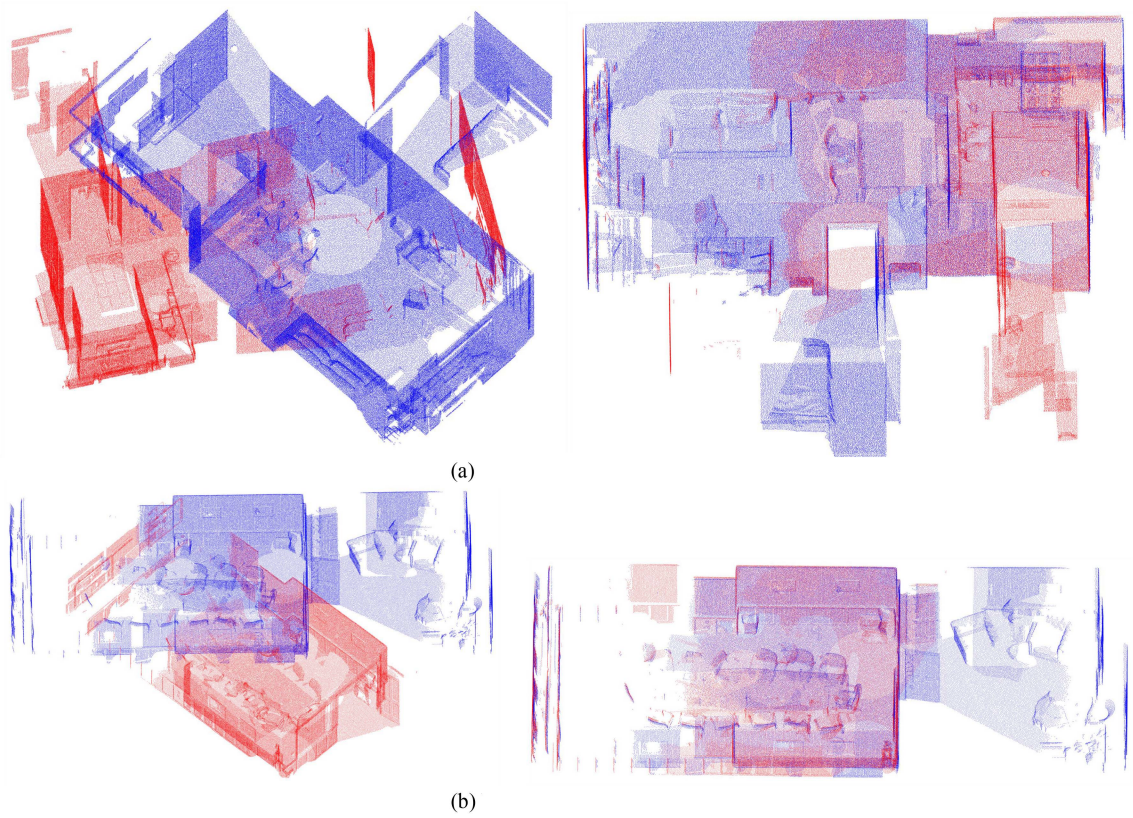


Fig. 9. Registration results of the point clouds of the indoor dataset. In each subfigure, the initial point clouds are on the left and the registered point clouds are on the right. (a) Apartment. (b) Boardroom.

TABLE III
MEAN ROTATION ERROR $\overline{\text{ERROR}_R}$, MEAN TRANSLATION ERROR $\overline{\text{ERROR}_t}$, AND MEAN COMPUTATION TIME $\bar{t}$ FOR ALL THE COMBINATIONS

Combination		C1	C2	C3	C4	C5	C6	C7	C8
Apartment	$\overline{error_R}$ (°)	0.5998	0.6580	0.7142	0.5998	0.7217	0.6580	0.7142	0.7217
	$\overline{error_t}$ (m)	0.2118	0.2254	0.2387	0.2118	0.2407	0.2254	0.2387	0.2407
	$\bar{t}$ (s)	55.5699	26.7857	27.4787	32.5419	20.3366	21.9040	22.9844	19.9330
Boardroom	$\overline{error_R}$ (°)	0.5504	0.5093	0.5504	0.5504	0.5042	0.5093	0.5504	0.5042
	$\overline{error_t}$ (m)	0.5080	0.4768	0.5080	0.5080	0.4713	0.4768	0.5080	0.4714
	$\bar{t}$ (s)	91.1255	49.2789	63.7997	77.4629	33.3173	47.0027	54.5091	34.4908
City 1	$\overline{error_R}$ (°)	1.0172	1.7683	1.3129	1.0172	1.7683	1.7683	1.3129	1.7683
	$\overline{error_t}$ (m)	0.1698	0.2775	0.1859	0.1698	0.2746	0.2775	0.1859	0.2746
	$\bar{t}$ (s)	29.6152	14.7271	11.5583	15.7221	8.4254	10.8236	10.0136	7.7558
City 2	$\overline{error_R}$ (°)	0.1546	0.1427	0.1554	0.1546	0.1427	0.1427	0.1554	0.1427
	$\overline{error_t}$ (m)	0.2016	0.1349	0.2007	0.2016	0.1329	0.1349	0.2007	0.1349
	$\bar{t}$ (s)	395.3386	298.9029	421.9873	372.0196	351.8075	278.0863	348.3775	344.2779

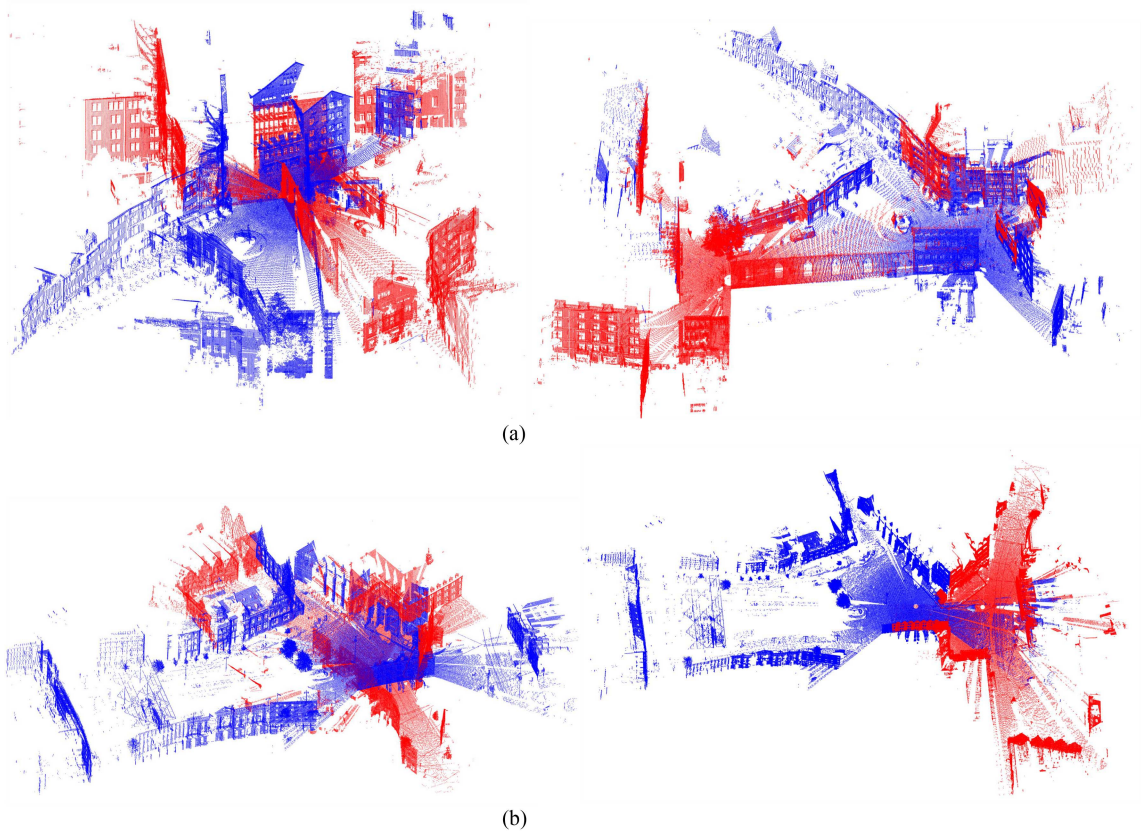


Fig. 10. Registration results of the point clouds of the outdoor dataset. In each subfigure, the initial point clouds are on the left and the registered point clouds are on the right. (a) City 1. (b) City 2.

time is pretty few. This is because the correct correspondences established by the LSD are very few on this scene. Therefore, most of iterations are not performed. A better LRF (LRA) construction method may improve the performance of the ACLRF (ACLRA).

D. Comparison With the Existing 3-D Transformation Estimation Techniques

The two-point RANSACWC algorithm is compared with the one-point RANSAC [22] and optimized sample consensus (OSAC) [47] algorithms. Considering the advantage of the one-point RANSAC algorithm is that it requires fewer iterations, its iteration number is set as 500. The iteration number of the OSAC algorithm is set as 2000. The iteration number of the two-point RANSACWC algorithm is still set as 2000. The computation results are listed in Table IV.

From Table IV, in aspect of registration accuracy, the OSAC algorithm has the best performance beside the apartment scene. The algorithm uses three correspondences to calculate the transformation, so the main error source is the point noise. Therefore, it gets better registration accuracy. On the apartment scene, the algorithm usually fails to register the two point clouds. The OSAC algorithm requires three correct correspondences to get correct transformation, while the one-point RANSAC requires one correct correspondence and the two-point RANSACWC requires two correct correspondences. Hence, the OSAC algorithm needs more iterations. This is the reason why it usually fails on the apartment scene. The one-point RANSAC algorithm uses the LRFs of the keypoints to calculate the transformation. However, the LRFs often have large errors. This leads to the poor registration accuracy. The main error source of the two-point RANSACWC is the LRA error. The LRA error is generally smaller than the LRF error because the LRA is calculated by using a small number of neighboring points. Therefore, the two-point RANSACWC is better than the one-point RANSAC. In aspect of computation time, the two-point RANSACWC algorithm has the best time efficiency. The algorithm uses the constraints to skip the iterations when the selected correspondences do not comply with the constraints. Although the one-point RANSAC algorithm requires fewer iterations, its computation time is much more than that of our algorithm. In the algorithm, all the iterations are performed, so it is still computationally expensive. The algorithm only uses one correspondence to calculate the transformation, so the constraints cannot be inserted into the iteration process. The OASC algorithm is the most time-consuming, though it first eliminates some incorrect correspondences. It is worth noting that the 2SAC-GC [23] algorithm also applies the GC and ACLRA constraints, so the two-point RANSACWC algorithm based on C4 is equivalent to the 2SAC-GC algorithm. Thus, we also prove that our

TABLE IV
MEAN ROTATION ERROR, MEAN TRANSLATION ERROR, AND MEAN COMPUTATION TIME OF THE THREE ALGORITHMS

Method		One-point RANSAC	OSAC	Two-point RANSAC WC
Apartment	$\overline{error_R}$ (°)	0.9819	14.1793	0.7217
	$\overline{error_t}$ (m)	0.2926	2.0815	0.2407
	$\bar{t}$ (s)	410.7397	1463.3368	19.9330
Boardroom	$\overline{error_R}$ (°)	0.4608	0.2088	0.5042
	$\overline{error_t}$ (m)	0.4678	0.2064	0.4714
	$\bar{t}$ (s)	200.208	983.8176	34.4908
City 1	$\overline{error_R}$ (°)	3.0889	2.4206	1.7683
	$\overline{error_t}$ (m)	0.5044	0.1722	0.2746
	$\bar{t}$ (s)	634.415	1966.7199	7.7558
City 2	$\overline{error_R}$ (°)	0.5893	0.0552	0.1427
	$\overline{error_t}$ (m)	0.1933	0.0628	0.1349
	$\bar{t}$ (s)	522.4383	2467.1893	344.2779

two-point RANSACWC algorithm is better than the 2SAC-GC algorithm.

VI. CONCLUSION

In this article, an LSD is first proposed. The LSD (i.e., LOVC descriptor) is a naturally binary descriptor, so it consumes fewer storage. The descriptor takes the relationship between voxels into consideration, so more local surface information is encoded. Then, a registration method is presented, in which the proposed LSD is used to establish the correspondences. In addition, in order to find correct correspondences and calculate accurate transformation, a 3-D transformation estimation technique (i.e., two-point RANSACWC algorithm) is developed. In this algorithm, multiple constraints are applied to facilitate the searching process of correct correspondences. The LOVC descriptor and two-point RANSACWC algorithm make the registration method have high registration accuracy and good computation efficiency.

The experiments have been performed to analyze the LOVC descriptor and two-point RANSACWC algorithm. By comparing with the existing descriptors, the experimental results well demonstrate that the LOVC descriptor has high descriptiveness. Thus, more correct correspondences can be established. Then, the two-point RANSACWC algorithm is analyzed. The combinations of different constraints are studied and the merit of the constraints is given. The performance of different constraints is analyzed. These constraints are very useful to decrease computation time. Finally, the combination C8 is employed in our two-point RANSACWC algorithm. By comparison with the existing 3-D transformation estimation techniques, the two-point RANSACWC algorithm has obviously better time efficiency and relative good registration accuracy.

DISCUSSION

This article focuses on LSD-based point cloud registration. Many scholars from the fields of remote sensing and computer vision are dedicating to point cloud data processing. The registration method can be used for reconstruction of building or urban scene, so this article is in the scope of JSTARS.

ACKNOWLEDGMENT

The authors would like to thank the Stanford 3-D Scanning Repository, University of Western Australia, and Robotic 3-D Scan Repository for publishing their datasets on the internet.

REFERENCES

[1] Z. Dong, B. Yang, F. Liang, R. Huang, and S. Scherer, "Hierarchical registration of unordered TLS point clouds based on binary shape context descriptor," *ISPRS J. Photogrammetry Remote Sens.*, vol. 144, pp. 61–79, Oct. 2018.

[2] J. Yang, Y. Xiao, and Z. Cao, "Aligning 2.5D scene fragments with distinctive local geometric features and voting-based correspondences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 714–729, Mar. 2019.

[3] W. Tao, X. Hua, K. Yu, X. Chen, and B. Zhao, "A pipeline for 3-D object recognition based on local shape description in cluttered scenes," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 801–816, Jan. 2021.

[4] J. Yang, S. Fan, Z. Huang, S. Quan, W. Wang, and Y. Zhang, "VOID: 3D object recognition based on voxelization in invariant distance space," *Vis. Comput.*, vol. 39, pp. 3073–3089, Jun. 2022, doi: 10.1007/s00371-022-02514-1.

[5] D. Xu, J. Liu, Y. Liang, X. Lv, and J. Hyyppä, "A LiDAR-based single-shot global localization solution using a cross-section shape context descriptor," *ISPRS J. Photogrammetry Remote Sens.*, vol. 189, pp. 272–288, Jul. 2022.

[6] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[7] J. Servos and S. L. Waslander, "Multi-channel generalized-ICP: A robust framework for multi-channel scan registration," *Robot. Auton. Syst.*, vol. 87, pp. 247–257, Jan. 2017.

[8] W. Tao, X. Hua, K. Yu, X. He, and X. Chen, "An improved point-to-plane registration method for terrestrial laser scanning data," *IEEE Access*, vol. 6, pp. 48062–48073, 2018.

[9] 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.

[10] B. Zhao and J. Xi, "Efficient and accurate 3D modeling based on a novel local feature descriptor," *Inf. Sci.*, vol. 512, pp. 295–314, Feb. 2020.

[11] O. Kechagias-Stamatis and N. Aouf, "Histogram of distances for local surface description," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 16–21.

[12] 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.

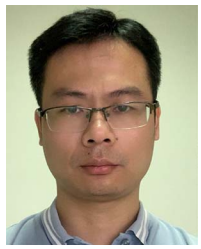
[13] S. Ao, Y. Guo, S. Gu, J. Tian, and D. Li, "SGHs for 3D local surface description," *IET Comput. Vis.*, vol. 14, no. 4, pp. 154–161, Jun. 2020.

[14] Y. Zhang, C. Li, B. Guo, C. Guo, and S. Zhang, "KDD: A kernel density based descriptor for 3D point clouds," *Pattern Recognit.*, vol. 111, pp. 107691, Mar. 2021.

[15] T. Sun, G. Liu, S. Liu, F. Meng, L. Zeng, and R. Li, "An efficient and compact 3D local descriptor based on the weighted height image," *Inf. Sci.*, vol. 520, pp. 209–231, May 2020.

[16] S. M. Prakhya, B. Liu, W. Lin, V. Jakhetiya, and S. C. Guntuku, "B-SHOT: A binary 3D feature descriptor for fast keypoint matching on 3D point clouds," *Auton. Robots*, vol. 41, pp. 1501–1520, Oct. 2017.

[17] O. Kechagias-Stamatis, N. Aouf, and L. Chermak, "B-HoD: A lightweight and fast binary descriptor for 3D object recognition and registration," in *Proc. IEEE 14th Int. Conf. Netw., Sens. Control*, 2017, pp. 37–42.
[18] Z. Dong, B. Yang, Y. Liu, F. Liang, B. Li, and Y. Zang, "A novel binary shape context for 3D local surface description," *ISPRS J. Photogrammetry Remote Sens.*, vol. 130, pp. 431–452, Aug. 2017.
[19] S. Quan, J. Ma, F. Hu, B. Fang, and T. Ma, "Local voxelized structure for 3D binary feature representation and robust registration of point clouds from low-cost sensors," *Inf. Sci.*, vol. 444, pp. 153–171, May 2018.
[20] S. Quan, J. Ma, T. Ma, F. Hu, and B. Fang, "Representing local shape geometry from multi-view silhouette perspective: A distinctive and robust binary 3D feature," *Signal Process.: Image Commun.*, vol. 65, pp. 67–80, Jul. 2018.
[21] M. A. Fischler and R. C. 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.
[22] Y. Guo, B. Mohammed, S. Ferdous, M. Lu, and J. Wan, "An integrated framework for 3-D modeling, object detection, and pose estimation from point-clouds," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 3, pp. 683–693, Mar. 2015.
[23] J. Yang, Q. Zhang, and Z. Cao, "Multi-attribute statistics histograms for accurate and robust pairwise registration of range images," *Neurocomputing*, vol. 251, pp. 54–67, Aug. 2017.
[24] S. Quan and J. Yang, "Compatibility-guided sampling consensus for 3-D point cloud registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7380–7392, Oct. 2020.
[25] J. Li, Q. Hu, and M. Ai, "GESAC: Robust graph enhanced sample consensus for point cloud registration," *ISPRS J. Photogrammetry Remote Sens.*, vol. 167, pp. 363–374, Sep. 2020.
[26] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, Jul. 2007.
[27] J. Yang, Y. Xiao, Z. Cao, and W. Yang, "Ranking 3D feature correspondences via consistency voting," *Pattern Recognit. Lett.*, vol. 117, pp. 1–8, Jan. 2017.
[28] F. Tombari and L. Di Stefano, "Object recognition in 3D scenes with occlusions and clutter by Hough voting," in *Proc. Pacific-Rim Symp. Image Video Technol.*, 2010, pp. 349–355.
[29] E. Rodola, A. Albarelli, F. Bergamasco, and A. Torsello, "A scale independent selection process for 3d object recognition in cluttered scenes," *Int. J. Comput. Vis.*, vol. 102, no. 1–3, pp. 129–145, Mar. 2013.
[30] F. Tombari, S. Salti, and L. Di Stefano, "Unique shape context for 3d data description," in *Proc. Universal Cross-Domain 3D Model Retrieval*, 2010, pp. 57–62.
[31] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *Proc. 8th Eur. Conf. Comput. Vis.*, 2004, pp. 224–237.
[32] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, Oct. 2013.
[33] J. Yang, Q. Zhang, Y. Xiao, and Z. Cao, "TOLDI: An effective and robust approach for 3D local shape description," *Pattern Recognit.*, vol. 65, pp. 175–187, May 2017.
[34] J. Yang, S. Quan, P. Wang, and Y. Zhang, "Evaluating local geometric feature representations for 3D rigid data matching," *IEEE Trans. Image Process.*, vol. 29, pp. 2522–2535, 2020.
[35] D. L. Bibissi, J. Yang, S. Quan, and Y. Zhang, "Dual spin-image: A bi-directional spin-image variant using multi-scale radii for 3D local shape description," *Comput. Graph.*, vol. 103, pp. 180–191, Apr. 2022.
[36] Z. Du et al., "MDCS with fully encoding the information of local shape description for 3D Rigid Data matching," *Image Vis. Comput.*, vol. 121, May 2022, Art. no. 104421.
[37] Y. Zou, X. Wang, T. Zhang, B. Liang, J. Song, and H. Liu, "BRoPH: An efficient and compact binary descriptor for 3D point clouds," *Pattern Recognit.*, vol. 76, pp. 522–536, Apr. 2018.
[38] J. Yang, K. Xian, P. Wang, and Y. Zhang, "A performance evaluation of correspondence grouping methods for 3D rigid data matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 1859–1874, Jun. 2021.
[39] D. Zai et al., "Pairwise registration of TLS point clouds using covariance descriptors and a non-cooperative game," *ISPRS J. Photogrammetry Remote Sens.*, vol. 134, pp. 15–29, Dec. 2017.
[40] R. Lu, F. Zhu, Q. Wu, and X. Fu, "Search inliers based on redundant geometric constraints," *Vis. Comput.*, vol. 36, pp. 253–266, Feb. 2020.
[41] R. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 1–4.
[42] Y. A. Felus and R. C. Burtch, "On symmetrical three-dimensional datum conversion," *GPS Solution*, vol. 13, pp. 65–74, Jan. 2009.
[43] F. Tombari, S. Salti, and L. Di Stefano, "Performance evaluation of 3D keypoint detectors," *Int. J. Comput. Vis.*, vol. 102, pp. 98–220, Mar. 2013.
[44] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.
[45] J. Park, Q. Zhou, and V. Koltun, "Colored point cloud registration revisited," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 143–152.
[46] B. Zeisl, K. Koeser, and M. Pollefeys, "Automatic registration of RGB-D scans via salient directions," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2805–2815.
[47] J. Yang, Z. Cao, and Q. Zhang, "A fast and robust local descriptor for 3D point cloud registration," *Inf. Sci.*, vol. 346–347, pp. 163–179, Jun. 2016.



Wuyong Tao received the master's degree in surveying and mapping from the East China University of Technology, Nanchang, China, in 2015, and the Ph.D. degree in geodesy from the Wuhan University, Wuhan, China, in 2020.

He was a Visiting Ph.D. Student with the University of Calgary, Calgary, AB, Canada, from 2019 to 2020. He is currently a Lecturer with the Nanchang University, Nanchang, China. He has authored/coauthored more than 30 research papers. He has been a Reviewer for the *Measurement and Control, photonics, Journal of Intelligent & Fuzzy Systems, Forests, Sensors*, and *Remote Sensing*. His research interests include point cloud registration, 3-D object recognition, and deep learning.

Shaoping Xu received the M.S. degree in computer application from the China University of Geosciences, Wuhan, China, in 2004, and the Ph.D. degree in mechatronics engineering from the University of Nanchang, Nanchang, China, in 2010.

He is currently a Full Professor with the Department of Computer Science and Technology, School of Mathematics and Computer Sciences, Nanchang University, Nanchang, China. He has authored/coauthored more than 50 articles in journals and conference proceedings. His research interests include digital image processing and analysis, computer graphics, virtual reality, etc.

Wei Huang received the B.Eng. and M.Eng. degrees in information engineering from the Harbin Institute of Technology, Harbin, China, and the Ph.D. degree in information engineering from the Nanyang Technological University, Singapore.

He was with the University of California San Diego, San Diego, CA, USA, and the Agency for Science Technology and Research, Singapore, as a Postdoctoral Research Fellow. He is currently a Full Professor with the Department of Computer Science and acts as the Dean of the School of Mathematics and Computer Sciences, Nanchang University, Nanchang, China. He has authored or coauthored more than 100 academic journal or conference papers, including the IEEE TRANSACTIONS ON MEDICAL IMAGING, IEEE TRANSACTIONS ON MULTIMEDIA, *MICCAI*, *CVPR*, and *ACM Multimedia*. His research interests include machine learning, pattern recognition, computer vision, and multimedia.

Dr. Huang is a Principal Investigator in studies supported by more than 20 national or provincial grants, including multiple NSF-China projects and NSF key projects in Jiangxi Province, China. He was the recipient of the Jiangxi Provincial Natural Science Award, the Best Paper Award of MICCAI-MLMI, the most interesting Paper Award of ICME-ASMMC, the Best Paper Award of ICITBE, the Best Paper Award of ICCEAI, etc. He was also designated as the Academic Leader of Jiangxi Province, China.

Shufan Hu received the bachelor's degree in geophysics from the East China University of Technology, Nanchang, China, in 2014, and the master's and Ph.D. degrees in geophysics from the Tongji University, Shanghai, China, in 2017 and 2021, respectively.

He used to be a Visiting Ph.D. Student with the Politecnico di Torino, Torino, Italy, from 2018 to 2019. He is currently a Lecturer with the Department of Computer Science and Technology, Nanchang University, Nanchang, China. His research interests include signal processing, inverse problem, and machine learning.

Dr. Hu has been a Reviewer for the IEEE TGRS, *Geophysics*, *Solid Earth*, *Journal of Geophysical Research: Solid Earth*, *Journal of Applied Geophysics*, and *Exploration Geophysics*.

Meng Pang received the B.Sc. and M.Sc. degrees in software engineering from the Dalian University of Technology, Dalian, China, in 2013 and 2016, respectively, and the Ph.D. degree in computer science from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2019.

He was a Postdoctoral Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, from 2020 to 2022. He is currently a Distinguished Professor with the School of Mathematics and Computer Sciences, Nanchang University, Nanchang, China. His research interests include image processing, artificial intelligence security, and artificial intelligence medical.