# A Novel 3-D Local DAISY-Style Descriptor to Reduce the Effect of Point Displacement Error in Point Cloud Registration

Fariborz Ghorbani, Hamid Ebadi, Amin Sedaghat, and Norbert Pfeifer

***Abstract*—Three-dimensional (3-D) point clouds are widely considered for applications in different fields. Various methods have been proposed to generate point cloud data: LIDAR and image matching from static and mobile platforms, including, e.g., terrestrial laser scanning. With multiple point clouds from stationary platforms, point cloud registration is a crucial and fundamental issue. A standard approach is a point-based registration, which relies on pairs of corresponding points in two-point clouds. Therefore, a necessary step in point-based registration is the construction of 3-D local descriptors. One of the (many) challenges that will specifically affect the performance of local descriptors with local spatial information is the point displacement error. This error is caused by the difference in the distributions of points surrounding a (potentially) corresponding center point in the two-point clouds. It can occur for various reasons such as 1) distortions caused by the sensors recording the data, 2) moving objects, 3) varying density of point cloud, 4) change of viewing angle, and 5) different of the sensors. The purpose of this article is to develop a new 3-D local descriptor reducing the effect of this type of error in point cloud coarse registration. The approach includes an improved local reference frame and a new geometric arrangement in point cloud space for the 3-D local descriptor. Inspired by the 2-D DAISY descriptor, a geometric arrangement is created to reduce the effect of the point displacement error. in addition, directional histograms are considered as features. Investigations are performed for point clouds from challenging environments, which are publicly available. The results of this study show the high performance of the proposed approach for point cloud registration, especially in more challenging and noisy environments.**

***Index Terms*—3-D descriptor, coarse registration, displacement error, mobile laser scanner, terrestrial laser scanning (TLS).**

## I. Introduction

GENERATION of the three-dimensional (3-D) point clouds from various objects and surfaces has recently been widely considered. Airborne LIDAR, mobile mapping, and terrestrial laser scanners are the principal devices used to produce point clouds. Point cloud registration is a crucial and fundamental issue with this significant data source. The primary purpose of point cloud registration is to find a suitable spatial transformation to determine the conformity and overlap of the geometry of two or more 3-D sets obtained from different sensors, times, or locations. Applications such as 3-D modeling [1], 3-D change detection [2], 3-D object detection [3], 3-D positioning of robots [4], and semantic interpretations of 3-D scenes [5] are among the most important applications in which the point cloud registration is a crucial component. Generally, point clouds registration methods can be divided into three categories: 1-greedy searching-based [6], [7], 2-global feature-based [8], [9], and 3-local feature-based [10]–[13]. Greedy search-based methods select several points from the source and target point clouds and find the corresponding points based on a comprehensive search. These methods are time-consuming in large-scale and high-resolution data. Global features describe the entire model shape, while local descriptors describe only the characteristics in a neighborhood of the feature points. Since complete coverage of the two-point clouds will not always be available, the global features often suffer from the limitation of discriminative power. In contrast, local features are more suitable for aligning point clouds which partially overlap. Both global and local feature-based methods consist of two parts, coarse and fine registration [14]. According to [14], the main challenge in point cloud registration is coarse registration. The coarse registration decreases the amount of rotation and translation between the point clouds. This step is as an initial value for fine registration.

3-D local descriptors are an essential part of a coarse registration process. These descriptors were first presented in the image space. They were used in many applications such as remote sensing image matching [15], image registration [16], geospatial target detection [17], geographic image retrieval [18], etc. They allowed reliably automating a majority of these tasks. Inspired by descriptors in image space, 3-D local descriptors in a 3-D space were presented. Despite the high importance of 3-D local feature descriptors and the introduction of many methods in recent decades, the accurate, fast, and highly reliable performance of this process still faces major problems [19]. 3-D local descriptors of point clouds must be stable to various geometric transformations. Besides, they must show an

Manuscript received November 12, 2021; revised January 12, 2022 and February 10, 2022; accepted February 12, 2022. Date of publication February 16, 2022; date of current version March 21, 2022. *(Corresponding author: Fariborz Ghorbani.)*

Fariborz Ghorbani and Hamid Ebadi are with the Department of Photogrammetry and Remote Sensing, Geomatics Engineering Faculty, K. N. Toosi University of Technology, Tehran 15433-19967, Iran (e-mail: f_ghorbani95@yahoo.com; ebadi@kntu.ac.ir).

Amin Sedaghat is with the Department of Geomatics Engineering, Faculty of Civil Engineering, University of Tabriz, Tabriz 51666, Iran (e-mail: am.sedaghat@gmail.com).

Norbert Pfeifer is with the Department of Geodesy and Geoinformation, Technische Universität Wien, 1040 Vienna, Austria (e-mail: norbert.pfeifer@geo.tuwien.ac.at).

Digital Object Identifier 10.1109/JSTARS.2022.3151699

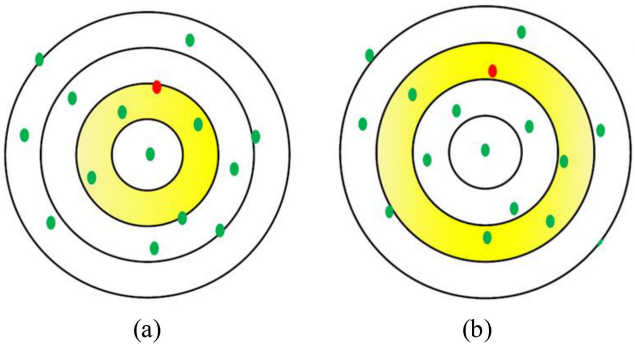



Fig. 1. Example of the belonging of the same points (red point) located at the boundary of the division to different bins (yellow bins).

appropriate performance against noise, different densities, clutters, occlusions, and missed data (e.g., due to different viewpoints). It is necessary to use descriptors with high descriptiveness in the face of these types of errors in challenging data. As discussed in [11] and [12], encoding the shape geometry, along with local spatial information, significantly improves the descriptiveness of a local descriptor.

One of the errors we encounter in most point cloud data is the point displacement error. This error has not been addressed in previous descriptors, and we define it in this study. It can occur for various reasons such as distortions caused by the sensors receiving data, moving objects, the varying density of point clouds, difference in viewing angle, and different resolution of sensors. This error is caused by the distributions of points surrounding a (potentially) corresponding center point in the two-point clouds. It is particularly problematic for descriptors encoding spatial information by binning data. Similar points in the two-point clouds, located near the boundary of those geometric arrangement descriptors, may appear in different bins. This type of error reduces the discriminative power of the descriptors. Fig. 1 shows that the same point in two-point clouds belongs to two separate bins.

3-D local descriptors can be divided into three categories [20]: 1-descriptors that use a local reference framework (LRF), 2-descriptors that use a local reference axis (LRA), and 3-descriptors which do not use any local reference systems. Descriptors without a local reference system (such as FPFH descriptors) often utilize geometric features. Their descriptiveness is usually low because spatial distribution information is not used in these descriptors [21]. Encoding the geometric information along with spatial information significantly increases the descriptor's descriptiveness. These descriptors generate spatial information by dividing the 3-D space based on either LRA or LRF. An LRF consists of three axes perpendicular to each other, while an LRA consists of only one axis. Therefore, LRF provides spatial information, including radial, azimuth, and altitude, while LRA does not include azimuth information. As a result, LRFs make the descriptor stable concerning geometric transformations and provide all the spatial information to describe the surface. Recent evaluations of standard datasets indicate the superiority of descriptors using the LRF system [20]. Nevertheless, LRF systems are more sensitive to noise, and most of them are challenged by determining the $x$-axis [20]. This is because the $z$-axis comes from the normal vector, but the $x$-axis defines the orientation within the tangent plane and is affected by noise, clutter, and occlusion [20]. As a result, providing a stable LRF can play a significant role in descriptor performance.

### A. Related Work

Point cloud registration in remote sensing is often done on large scales of data, and it is always considered a challenging topic. Various methods for point cloud registration in remote-sensing data have been presented. For example, Stamos and Leordeanu [22] used a feature-based method for point cloud registration in large-scale urban environments. They first identified the intersecting lines between adjacent planes and then estimated the transformation parameters of adjacent points using at least two intersecting lines. Weber *et al.* [23] generated the Fast Point Feature Histograms (FPFH) descriptor for each extracted keypoint and then matched the detected keypoints using a polygon-based approach. In 2016, Yang *et al.* [24] extracted lines from artificial buildings and pole-like objects and matched them using a combination of semantic and geometric features. Zai *et al.* [25] generated a covariance matrix to identify keypoints and create descriptors. They performed a matching process using a noncooperative game approach. In 2018, Dong *et al.* [26] proposed the hierarchical merging based multiview registration (HMMR) algorithm for unordered point cloud registration. They have calculated a local descriptor (the Binary Shape Context) and a global descriptor (the Vector of Locally Aggregated Descriptor). Then, the point cloud overlapping graph is generated based on the similarity between corresponding global descriptors. Finally, they performed hierarchically multiple point clouds registration using an iterative approach. In 2019, Cai *et al.* [27] proposed a method to increase computational efficiency in the TLS point cloud registration. They proposed a fast branch-and-bound algorithm based on 3-D keypoint correspondence for a 1-D rotation search. Their primary focus is to provide a method with high computational efficiency in 4 degrees of freedom registration. In 2020, Ge and Hu [28] proposed an object-based incremental registration strategy for TLS point clouds in urban environments. They increase the reliability of the 2-D transformation step by using initial lines and have achieved the global optimum by least-squares optimization. In 2021, Li *et al.* [29] proposed a method for point clouds coarse registration without the need for initial points. They decompose the seven parameters of the registration problem into three sub-problems of estimating the parameters of translation, rotation, and scale based on line vectors. The OPRANSAC algorithm was presented to estimate the translation and scale parameters in this research. And used a scale-annealing biweight estimator to estimate the rotation parameter. In 2021, Li *et al.* [30] proposed an approach to increase the coarse registration efficiency of point clouds. They proposed a polynomial time (O (N2)) outlier removal method. Their main idea was to reduce the initial volume of the input point cloud to a smaller set with a less outlier based on the bound principle.

These methods were widely used for coarse registration. However, they still have limitations. These methods depend on lines, planes, and surfaces and only work well in areas with man-made features but will not work well in natural areas with fewer structural features. On the other hand, point-based methods give good results in most environments, but these methods must have high stability to noise.

Other methods have been proposed for 3-D point cloud registration, including point cloud registration by deep learning [31]–[34]. These methods are mostly used in indoor space and small scale and, due to the limitations associated with data and complexity, do not have good results in large-scale point cloud registration [35]. The 4PCS-based methods [36]–[38] have also been used for coarse registration and can work well in the case of low overlap [35]. These methods are presented to eliminate the initial correspondences, and its direct application in the registration process, especially on a large scale, is very time-consuming. Probability-based methods are another type of approach for point cloud registration [39], [40]. These methods also consider registration as a probability estimation problem. The registration results in these methods depend on the sampling results. They cannot consider a large volume of points simultaneously.

### B. Research Aims

Although research has been done on the 3-D point cloud registration, they have been studied on a limited set of data in a specific scenario [41]. In addition, most of the research is based on data collected by themselves and designed for a particular application and not accessible to other researchers. Recently, real datasets in remote sensing and photogrammetry have been presented for point cloud registration [41], [42]. These data are produced by different methods and have a variety of environments. This type of point cloud data makes it possible to have comprehensively evaluated registration methods.

A lot of research has addressed the various challenges for making local features descriptors. They have investigated the effective factors in improving their performance [10]–[14]. Nonetheless, providing an approach to reduce the effect of displacement error can be considered as a vacancy in previous research. Motivated by these considerations, the main goal of this article is to provide an accurate and stable framework for large-scale point cloud registration to reduce the effect of the point displacement error. This method includes a descriptor with the new geometric arrangement and an improved LRF. Inspired by the 2-D DAISY [44] descriptor, we use circles with overlapping coverage in the local descriptors to reduce the effect of point displacement error. In addition, three perpendicular planes to each other are selected in the LRF to overcome the missed data problem. The points in a specific neighborhood are projected to these plans. The final descriptor is obtained after generating the directional histograms of each plane and their linear concatenation. Because this descriptor is applied directly to point clouds, it does not require heavy preprocessing such as mesh production. The proposed method is evaluated on real challenging data prepared with different scenarios.

The main contributions of this article are summarized as follows.

1) Introduce a novel descriptor inspired by the 2-D DAISY geometric structure with high robustness against displacement errors.
2) Provide an improved LRF that reduces the effect of error due to point displacements.

The rest of this article is organized as follows: Some descriptors will be reviewed in Section II. Section III provides details of the research theory. The display and evaluation results are discussed in Section IV. Finally, Section V concludes this article.

## II. Overview of Descriptors

Many methods have been presented to describe 3-D local features in point clouds. In this section, considering that the proposed method is among the descriptors with LRF, this local reference framework is examined. Furthermore, some 3-D descriptors and their geometric arrangement will be considered in more detail.

### A. Methods for Determining LRF

A wide range of LRFs has also been introduced following various methods for 3-D local feature descriptors [10], [45]–[49]. These methods are divided into two categories based on covariance analysis (CA) and geometric attributes (GA) [20].

Among the CA methods, in 2008, Novatnack and Nishino [45] used the intrinsic properties of the normal vector and the eigenvectors of the covariance matrix to construct the LRF. In this method, the $z$-axis is determined as the normal vector of the keypoints. The covariance matrix analysis is used to determine the $x$-axis, which is obtained from the neighboring points of keypoint points. The third axis is obtained by the cross-product of the $z$- and $x$-axes. The calculation speed in this method is high. But the main problem in this design is the ambiguity of the sign on all LRF axes. In 2010, Mian *et al.* [46] used a method based on analysis of the covariance matrix. In this method, the LRF unit vectors are obtained by the eigenvector of the covariance matrix. This method also has the problem of signing the LRF axes because it only signs the $z$-axis. In 2010, Tombari *et al.* [10] developed an LRF for the SHOT descriptor that signs all its axes. In this method, the weighted covariance matrix is used to determine the LRF. This method has created a balance between performance, repeatability, and stability capabilities in an area despite occlusion and clutter. Nonetheless, its main limitation is its sensitivity to changes in density in point clouds.

Among the GA-based methods, Chua and Jarvis [47] first used the attribute values to generate the $x$-axis of the LRF system. In this method, a 3-D curve is extracted by the intersection of the keypoints spherical neighborhood and the entire 3-D shape. A plane is then fitted to the points in the 3-D curve whose normal vector of that plane is defined as the $z$-axis of the LRF system. To avoid ambiguity of the $z$-axis sign, the vector whose cross-product is positive with the normal vector of the keypoint is selected. They have used a Signed-distance to determine the $x$-axis. In 2011, Petrelli and Di Stefano [48] used the normal attribute to specify the LRF. In this method, a small subset of adjacent points is used to fit a plane. They considered the normal

vector of the fitted plane as the *z*-axis. The *z*-axis sign is obtained via the inner product between it and the average normal vector over points in the support region. The point with the largest normal deviation angle to the normal of the keypoint is selected in the support region to determine the *x*-axis. The extension of the point (p) to the chosen point is defined as the *x*-axis. In 2012, Petrelli and Di Stefano [49] improved their previous work. The difference between their method and the previous work is in determining the *x*-axis. They used the Signed-distance attribute instead of the normals. Although research on LRFs has been conducted, most CA-based methods suffer from axial symptom ambiguity, and GA-based methods have low stability against high-density changes and noise [10].

### B. 3-D Local Features Descriptors

Many 3-D local feature descriptors have been introduced in recent years. This section introduces some of the most important algorithms for creating descriptors in 3-D data. In 2009, Rusu *et al.* [13] proposed the FPFH descriptor to reduce the computational burden of their previous method. This descriptor extracts geometric information in a neighborhood. Then, the SPFH is calculated for each neighboring point. The final descriptor is obtained from the sum weight of the SPFH criterion. The second category of descriptors uses spatial information along a local axis (LRA). In this category of descriptors, we can refer to spin image (SI) [12] in this category of descriptors. Normals are used as local reference axes in the SI descriptor. Then, features are generated in the neighborhood of each keypoint, according to the two distance criteria. Finally, the created features for the neighboring points are discretized into 2-D bins, and the SPIN Image descriptor is generated by connecting them. This descriptor is resistant to occlusions and clutters [12].

The SHOT descriptor was developed by Tombari *et al.* in 2010 [10]. This descriptor can be considered as a combination of geometric and spatial information. Initially, an LRF is defined with the property of disambiguation of axes sign and uniqueness. This LRF is determined for each point based on the eigenvalue decomposition of the covariance matrix in the keypoint neighborhood. A local coordinate system is formed that is segmented along the radial, azimuth, and altitude axes. Points within each bin are considered to obtain a local histogram. A histogram is produced based on the angle between normal at neighboring points and normal at the keypoint. Finally, the SHOT descriptor is created by connecting all these local histograms. Despite the distinctiveness of the descriptor, it is sensitive to changes in point clouds density. In 2013, Guo *et al.* [11] introduced the RoPS descriptor based on a new LRF. The points on the local plane rotate into the three axes (*x*, *y*, *z*). For each rotation, the points in the neighborhood are projected to three coordinate planes (*xy*, *yz*, *xz*). Then, a distribution matrix for each plane is created by dividing the plane into several parts. A histogram is produced by considering the points located in each bin.

Recently, deep learning methods have been used for local features description. These types of methods utilize a large amount of data for training. The aim is to learn high-level and highly descriptive descriptions. Deep learning methods are divided into three categories [50]: Voxel-based networks, multiview convolutional neural networks (CNNs), point-based networks. For example, PointNet [31], as a point-based network, is applied directly to unordered point clouds. This method is invariant to transformations and different permutations. PPFNet [32] gets a combination of point coordinates, normal vectors, point pair features, and neighborhood points as input. It uses the PointNet to learn geometric feature descriptions and random samples to obtain keypoints. 3DsmoothNet [33] is a voxel-based method and uses the smoothed density value representation to learn local features. SpinNet [34] employs projected points to spherical space and a spherical convolution to extract local features.

## III. Methodology

This research presents a framework for the 3-D point cloud coarse registration in challenging datasets. At first, 3-D keypoints are extracted using a 3DSIFT [51] detector. Inspired by the 2-D DAISY descriptor, a new 3-D descriptor is proposed to enhance discrimination power. The corresponding points are detected in the two-point clouds with the proposed descriptor. Then, a global transformation between the 3-D point clouds is used to eliminate wrong correspondences. Finally, point clouds registration is also performed using a suitable transformation function. In the following, the details of the proposed method are discussed.

### A. 3-D SIFT Detector

The SIFT algorithm is a computer vision algorithm used to detect and describe local features in images [52]. It uses the DoG operator to approximate the Laplacian of Gaussian filter and detects local features with specific dimensions. This algorithm has also been used for 3-D local features detection, implemented in Point Cloud Library (PCL) [51]. Detection of 3-D local features in this algorithm is as follows [53].

*Step 1:* The scale-space in a point cloud is defined as a 4-D function, *L*, with the convolution of a 3-D Gaussian function in the point cloud

$$L(x, y, z, \sigma) = G(x, y, z, \sigma) * P(x, y, z) \tag{1}$$

$$G(x, y, z, \sigma) = \frac{1}{\left(\sqrt{2\pi\sigma}\right)^3} e^{-(x^2+y^2+z^2)/2\sigma^2} \tag{2}$$

where *G* is the Gaussian function and *P(x,y,z)* is 3-D coordinates of a point cloud. The convolution is performed by voxel grid filters [54].

*Step 2:* The spatial-space DoG is generated. The DoG space is the difference between two consecutive spatial-space created in the point cloud using the Gaussian function. The DoG calculation relation is as follows:

$$\begin{aligned}&\mathrm{DoG}(x, y, z, k^i\sigma)\\ &\quad = P(x, y, z) * (G(x, y, z, k^{i+1}\sigma) - G(x, y, z, k^i\sigma)).\end{aligned} \tag{3}$$

*Step 3:* Local features in a 4-D position are obtained from the DoG function as local extrema. The extremes are found in

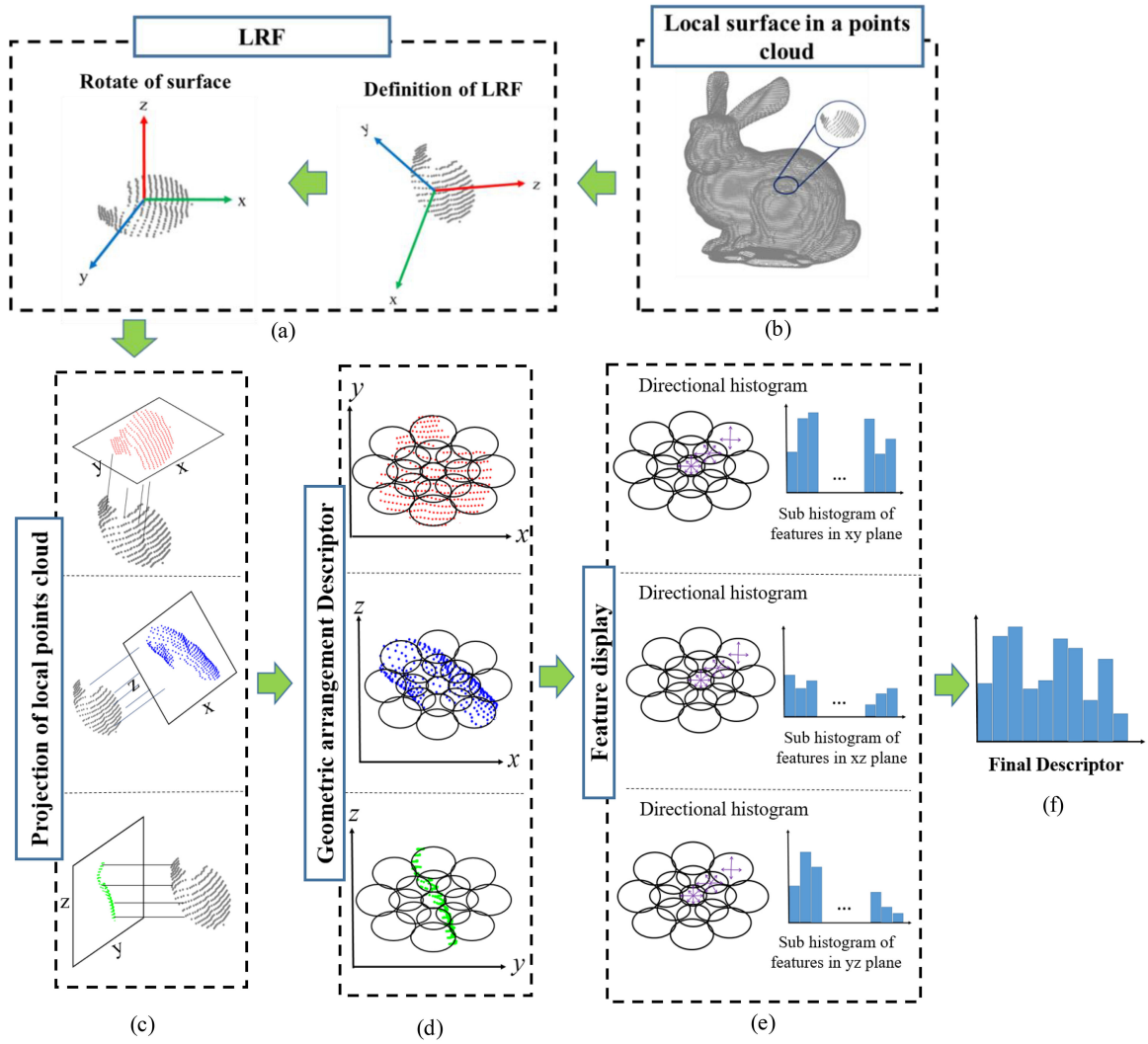



Fig. 2. Diagram of the production process of the proposed descriptor.)a) Displaying a local surface in the point cloud. (b) Definition of LRF in a local surface. (c) Projecting local points to 2-D planes. (d) Applying DAISY geometric arrangement to each 2-D plane. (e) Generating a directional histogram for each bin in 2-D planes. (f) Making the final descriptor.

the DoG space. This process is done by comparing each point with 26 neighboring points on its scale and 27 neighboring points above and below that point ($27 + 26 + 27 = 80$). A key point in this space is selected as a local feature that is larger or smaller than all its neighboring points.

*Step 4:* A threshold method is applied to remove the unstable points.

### B. 3-D Local DAISY-Style Descriptor

Methods based on local features for 3-D registration should use high-performance descriptors. Therefore, these descriptors must be stable against all types of errors. In this section, the details of the proposed method are presented for generating descriptors from local surfaces in the point cloud. At first, the developed LRF in this descriptor is introduced. The geometric arrangement and features used in the proposed descriptor are explained in the following. How to structure the proposed descriptor is shown in Fig. 2. At first, a local surface on the given point cloud is considered with the centrality of the keypoint ($p$) and the neighborhood radius R [see Fig. 2(a)]. Neighboring

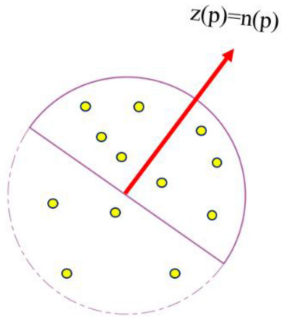



Fig. 3. How to determine the sign of LRF axes.

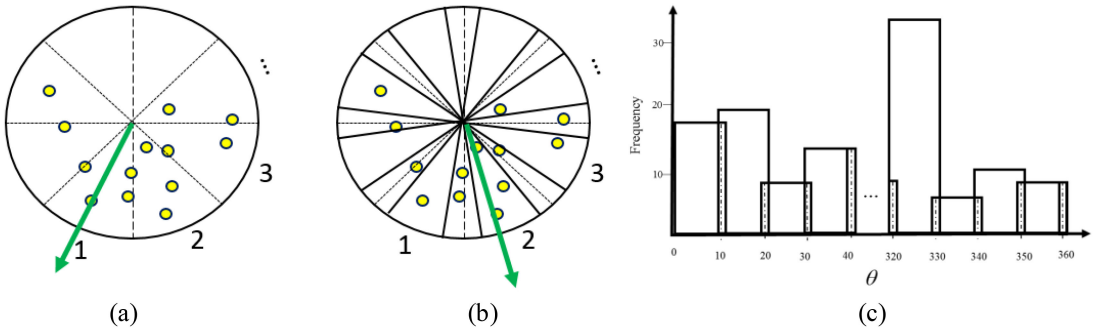



Fig. 4. Determining the dominant direction in the proposed method.

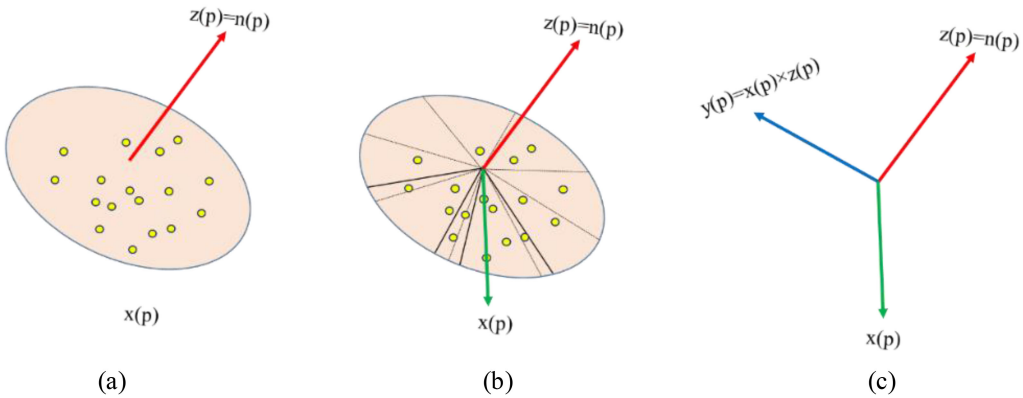



Fig. 5. How to produce the LRF. (a) Determining the $z$-axis. (b) Determining the $x$-axis. (c) Determining the $y$-axis.

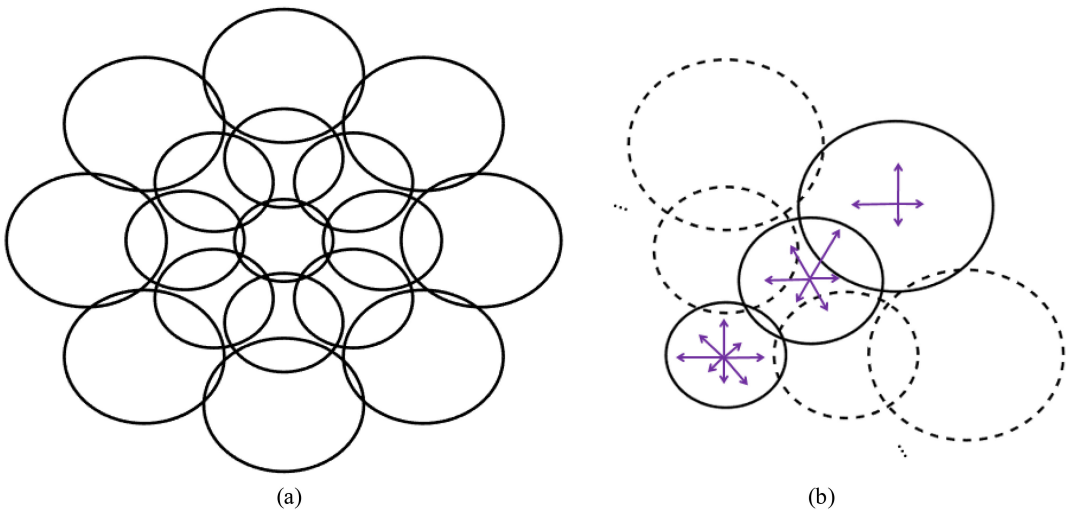



Fig. 6. (a) Geometric arrangement of the proposed descriptor into a 2-D plane. (b) Demonstration of the directional histogram used in the descriptor. The arrows represent the number of directional divisions.

points coordinate of the keypoint ($p$) are transferred to the local coordinates in the LRF, $Q = \{q_1, q_2, \ldots q_n\}$ with the respective center point, see Fig. 2(b). The descriptor becomes stable to various geometric transformations with this transfer. Three planes with different viewing angles are used due to the occlusions and clutters in the point clouds. The points in the vicinity of the keypoint are projected orthogonally to three planes obtained from the LRF, see Fig. 2(c). These are the $xy$, $xz$, and $yz$ planes, respectively. As shown in Fig. 2(d), the DAISY geometric arrangement is applied to the points projected on the 2-D planes. Then, a descriptor is generated for each 2-D plane by calculating a directional histogram, see Fig. 2(e). Finally,


Fig. 7. Demonstration of a scan of the data used. The colors are based on the height values. (a) Arch. (b) Trees. (c) Courtyard. (d) Facede. (e) ETH hauptgebaude. (f) Gazebo. (g) RESSO (7c). (h) RESSO (7d).

by linear connecting the histograms of each plane, the final descriptor is created, see Fig. 2(f).

A LRF must be defined for each keypoint *(p)* in the point cloud. The LRF is represented at *(p)* by three axes x*(p), y*(*p*), and *z(p)*. Where the *y(p)*-axis is determined by the cross product of the two axes *x(p)* and *z(p)*. The axes *x(p)* and *z(p)* must be specified to determine the LRF.

According to studies in most descriptors, normals are used as a proper and repeatable direction in the definition of LRFs [20]. In this research, the normals of the extracted key points are utilized as the $z$-axis of the LRF system. However, one of the challenges in most LRFs is the determination of the sign of the axes. The used approach in this research is to employ a hemisphere along the $z$-axis. If the number of points in the hemisphere in the initial direction is more than the other hemispheres, the positive axis sign is considered and otherwise, it will be negative. This process is as follows:

$$z(p) = \begin{cases} n(p), & \text{if } m_1(q) \geq m_2(q) \\ -n(p), & \text{otherwise} \end{cases} \tag{4}$$

where $m_1(q)$ and $m_2(q)$ are the number of points in the two hemispheres. This process is shown in Fig. 3.

The projection of neighborhood points in the plane perpendicular to the $z$-axis is used to determine the $x$-axis. A proposed method was adopted to overcome the points-displacement error in determining the $x$-axis. First, the neighboring points around

the keypoint (p) are considered with radius *r*. All neighboring points (*q*) are projected on the plane perpendicular to the normal of the key point, resulting in *q*'. This is a 2-D point with coordinates in a preliminary *xy*-system, of which the orientation within the plane is not of importance. The angle $\theta$, as an azimuth angle on a plane perpendicular to the normal vector (*z*-axis), for all projected points (*q'*) is calculated by (5), where (*n)* is the number of neighborhood points in radius (*r)* from the keypoint (*p)*. The $\theta$ angle is in [0°, 360°)

$$\theta^i_{(q')} = tan^{-1}((y^i_{(q')} - y(p))\Big/(x^i_{(q')} - x(p))) \; i = 1, 2, \ldots, n\,. \tag{5}$$

A weighted histogram determines the orientation of the *x*-axis of the LRF. The plane perpendicular to the normal is divided into 10° bins. Then, each point is placed in a bin according to (5). Due to the point displacement error, the points on the boundary of these divisions can be placed in different bins by slight movement. If it is decided about their location in binary form, it makes a mistake in determining the dominant direction, see Fig. 4(a) and (b). According to Fig. 4(a) and (b), most points are accumulated in bin number (2). However, the boundary points led to the selection of the bin number (1) as the dominant direction [see Fig. 4(a)]. In the proposed approach, the boundary points are assigned to two bins by considering the overlap (e.g., 5%) between the bins, see Fig. 4(b). Consequently, the effect of point displacement is reduced at the boundary of this division. Fig. 4(c) shows how the histogram is formed in the proposed method.

The points located in each angular bin are weighted using (6) to increase the stability of the LRF to clutters, occlusions, and missed areas. According to this equation, the closer the points are to the central point, the more effective it will be in determining the dominant direction. In this equation, *R* is the maximum radius in a neighborhood

$$wi = R - |q_i - p|\,. \tag{6}$$

After generating the histogram, the bin with the highest value is chosen for determining the dominant direction. The average projected coordinate of the points inside this bin is considered to determine the exact value of the direction. The vector obtained from the chosen point and the central point is considered as the *x*-axis of the LRF system. The *y*-axis is obtained by the cross-product of the two axes *x* and *z*. The calculation procedure of the LRF in the proposed descriptor is shown in Fig. 5.

Fig. 6(a) shows the geometric arrangement of the descriptor in a 2-D plane. Inspired by the 2-D DAISY descriptor, it consists of coated circles in different rings. This geometric arrangement will increase the robustness and discriminative of the descriptor. Methods that use mapping of 3-D points on 2-D planes have provided promising results in describing 3-D shapes [55]. These descriptors usually provide a directional histogram in each bin. We utilized an approach to creating a tradeoff between the discriminative and efficacy. In this method, directional divisions have been considered for each ring in proportion to its distance to the keypoint. The rings closer to the keypoint will be more effective, and more divisions are considered for them, and this division is reduced for farther rings, see Fig. 6(b). This process will reduce the descriptor dimensions, and increasing the impact of near points in making the descriptor.

A sub histogram of each bin is generated by considering the number of points located in each direction. By connecting these subhistograms, a histogram is created for each plane. Finally, the descriptor is obtained by linearly connecting the histograms from the planes *xy*, *xz*, and *yz*. The descriptor is normalized to [0, 1].

This descriptor has four main parameters including: descriptor radius (*R*), number of rings (*Q*), number of circles in each ring (*T*), and number of directional histogram divisions for each bin (*H*). Accordingly, the total number of circles (*S*) is equal to $S = T \times Q + 1$ and the dimension of the descriptor is $N = 3 \times \sum_{i=1}^{Q}(T_i \times H_i)$.

### C. Matching and Eliminating Wrong Correspondences

The next step after generating the descriptor is to establish a correspondence between the points. Two-point clouds are considered as the source ($\boldsymbol{P}^s$) and target ($\boldsymbol{P}^t$) and a known transformation function between them (*T)*. A descriptor is created for each keypoint in source ($p^s_i$) and target ($p^t_j$) point cloud. The closest keypoint in the target point clouds is found for each keypoint in the source point cloud. These points are considered initial correspondence points if their Euclidean distance is less than a threshold.

Initial correspondences contain incorrect correspondences and should be eliminated. The relationship between two-point clouds generated can be established using a 3-D global transformation like 3-D affine. This transformation function models geometric differences such as translations and rotations well. The M-estimator sample consensus (MSAC) [54], an extension of RANSAC, is used to estimate the parameters of the 3-D affine model. The points in the source data are then transferred to the target data space using the parameters of this transformation function, and their distance from the correspondence point in the target data is considered as the error of each corresponding pair. The root means square error (RMSE) is estimated as a criterion to determine the accuracy of the transformation of the source to the target point cloud. The RMSE value is compared with a threshold (*TG*). If the RMSE value exceeds the threshold (*TG*), the corresponding point pair with the highest error is considered incorrect and eliminated. This process is repeated until the desired threshold is reached.

## IV. Results

In this section, the results of the proposed method are presented. The results are compared with some popular traditional descriptors and the SpinNet deep learning descriptor. In the following are provided details of the results.

### A. Data Description

The performance of the proposed method is evaluated on three different benchmarks. Recently, some real challenging dataset of point clouds were made available to analyze registration algorithms. The most well-known of them are ETH

TABLE I
DETAILS OF THE PROPERTIES OF THE DATA USED (MR IS THE MEAN RESOLUTION)

| *Dataset* | *Scans* | *Mr(m)* | Environment organization | Environment locations | Spatial scale(m) |
|---|---|---|---|---|---|
| **Arch** | 5 | 0.20 | Unstructured | Outdoors | 70×110×130 |
| **Trees** | 6 | 0.25 | Unstructured | Outdoors | 45×100×100 |
| **Courtyard** | 8 | 0.15 | Unstructured | Outdoors | 40×140×140 |
| **Facade** | 8 | 0.07 | Unstructured | Outdoors | 20×55×50 |
| **ETH hauptgebaude** | 36 | 0.05 | Structured | Indoors &Outdoor | 15×20×60 |
| **Gazebo** | 32 | 0.05 | Semi-structured | Outdoors | 15×40×30 |
| ***RESSO* (7c)** | 5 | 0.20 | Unstructured | Outdoors | 20×120×160 |
| ***RESSO* (7d)** | 3 | 0.10 | Semi-structured | Outdoors | 15×90×80 |

PRS TLS dataset,[1] ETH ASL datasets and Repository,[2] and RESSO dataset.[3] These data include ground truth that allows for a proper assessment. In the following, we will review these data.

1) *ETH PRS TLS dataset*: This benchmark is provided for registration of TLS point clouds. It is collected from various scenes by Z + F Imager 5006i and Faro Focus 3D. The Arch, Trees, Courtyard, and Facade datasets are used in this benchmark.

a) *Arch dataset:* This data was generated to capture a Roman arch in five scenes around the arch. Each scan is obtained with relatively little coverage (30% to 40%) and this dataset also includes some objects such as vegetation and humans on the move.
b) *Courtyard Database:* This data contains eight scenes from the courtyard of an ancient tomb. There are no vertical objects in this data. It was generated to create DTMs from laser scanners.
c) *Trees Dataset:* It includes six scenes from a forest that also contains a lot of underwood with medium coverage.
d) *Facade dataset:* This data includes eight scenes from an urban area focusing on one specific facade. This data set has a high overlap between scans and also has artificial objects caused by moving cars and people.

2) *ETH ASL datasets Repository*: This benchmark was presented by Pomerleau *et al.* [42] to evaluate registration algorithms on the mobile laser scanner. The point clouds are produced by the Hokuyo 536 UTM-30LX as a rangefinder scanner. The ETH hauptgebaude and Gazebo datasets are used from this benchmark.

a) *ETH hauptgebaude dataset*: This dataset is collected in the main building (hauptgebaude) of ETH Zurich. It includes indoor and outdoor environments with a structured form. The purpose of obtaining this data is to investigate the robustness of the registration algorithms in the face of repetitive patterns.
b) *Gazebo dataset:* Gazebo data has a semi-structured and outdoor environment. The data was collected in a park that includes grass, scattered trees, and an asphalt road.

3) *RESSO dataset*: This benchmark is a TLS dataset presented by Chen *et al.* [43] to evaluate registering point clouds with a small overlap. The *RESSO* (7c) and *RESSO (7d)* data are used from this benchmark. These data were generated by a static laser scanner (the Leica ScanStation C10 static).

a) *RESSO (7c):* This is unstructured data prepared from the outdoor environment.
b) *RESSO (7d):* This data partially consists of curved surfaces, and planar structures are the dominant structure.

Fig. 7 shows a scan of each of the data used. Furthermore, a summary of the attributes of this data is provided in Table I. These three benchmarks cover a very wide range of scanners, scenes, resolutions, and they are therefore a good basis for evaluation of the suggested method.

### B. Evaluation Criteria

The proposed method was evaluated by three different criteria. The first criterion is suggested to measure rotational and translation error, generally used to assess the point cloud registration [26], [57]. Assume that the source point cloud $P_s$ is transferred using the $T_{s,t}$ transformation function to the target point cloud $P_t$. The remaining transformation $\Delta T_{s,t}$ is defined as follows:

$$\Delta T_{s,t} = T_{s,t}\left(T_{s,t}^{G}\right)^{-1} = \begin{bmatrix} \Delta R_{s,t} & \Delta t_{s,t} \\ 0 & 1 \end{bmatrix} \tag{7}$$

where $T_{s,t}$ is the estimated transformation function from $P_s$ to $P_t$ and $T_{s,t}^{G}$ is the true transformation function. Then, the rotation error $e_r$ and translation error $e_t$ from $P_s$ to $P_t$ based on the rotational components $\Delta R$ and the translation component $\Delta T$ are calculated as follows:

$$\begin{cases} e_{s,t}^{r} = \arccos\left(\frac{\mathrm{tr}(\Delta R_{s,t})-1}{2}\right) \\ e_{s,t}^{t} = \|\Delta t_{s,t}\| \end{cases} \tag{8}$$

where $\mathrm{tr}(\Delta R_{s,t})$ represents the trace of $\Delta R_{s,t}$ and the rotation error $e_{s,t}^{r}$ corresponds to the angle of rotation in the axis-angle representation.

[1][Online]. Available: https://prs.igp.ethz.ch/research/completed_projects/automatic_registration_of_point_clouds.html
[2][Online]. Available: https://projects.asl.ethz.ch/datasets/
[3][Online]. Available: https://3d.bk.tudelft.nl/liangliang/publications/2019/plade/resso.html

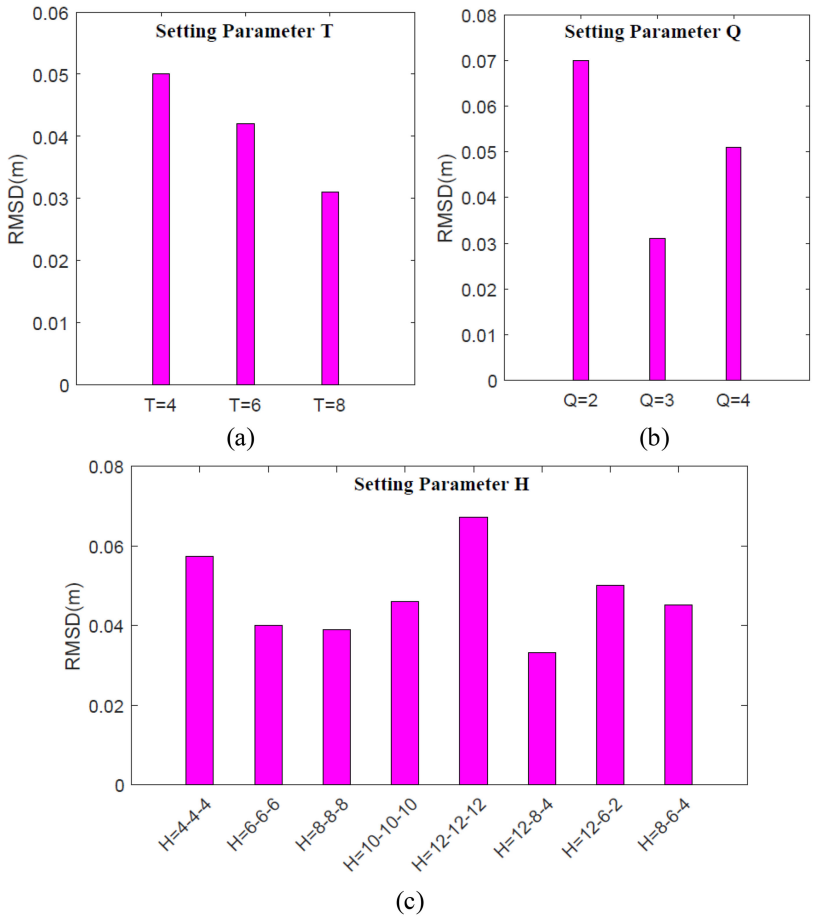



Fig. 8. Effect of $Q$, $T$, and $H$ parameters on the descriptor with RMSD.

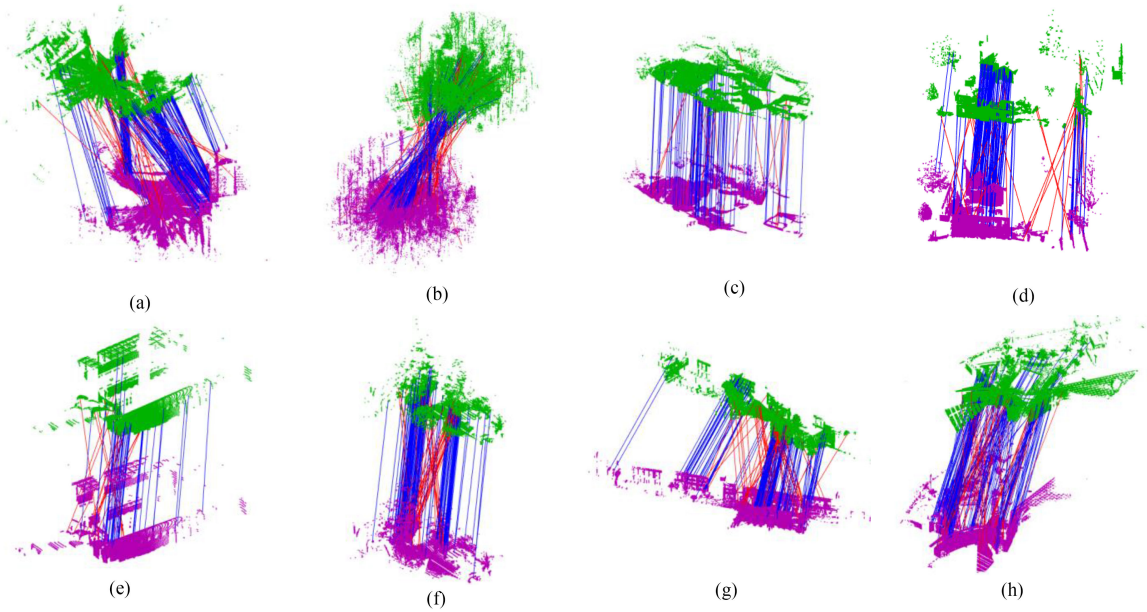



Fig. 9. Results of keypoints correspondence by the proposed method. Blue and red lines mean right and wrong correspondences, respectively. (a) Arch. (b) Trees. (c) Courtyard. (d) Facede. (e) ETH hauptgebaude. (f) Gazebo. (g) RESSO (7c). (h) RESSO (7c).

TABLE II
PARAMETERS SETTING

| | *Q* | *T* | *H* | *R(mr)* |
|---|---|---|---|---|
| setting of *Q* | 2-4 | 8 | 8 | 20 |
| setting of *T* | 3 | 4-6-8 | 8 | 20 |
| setting of *H* | 3 | 8 | 4-6-8-10-12 | 20 |

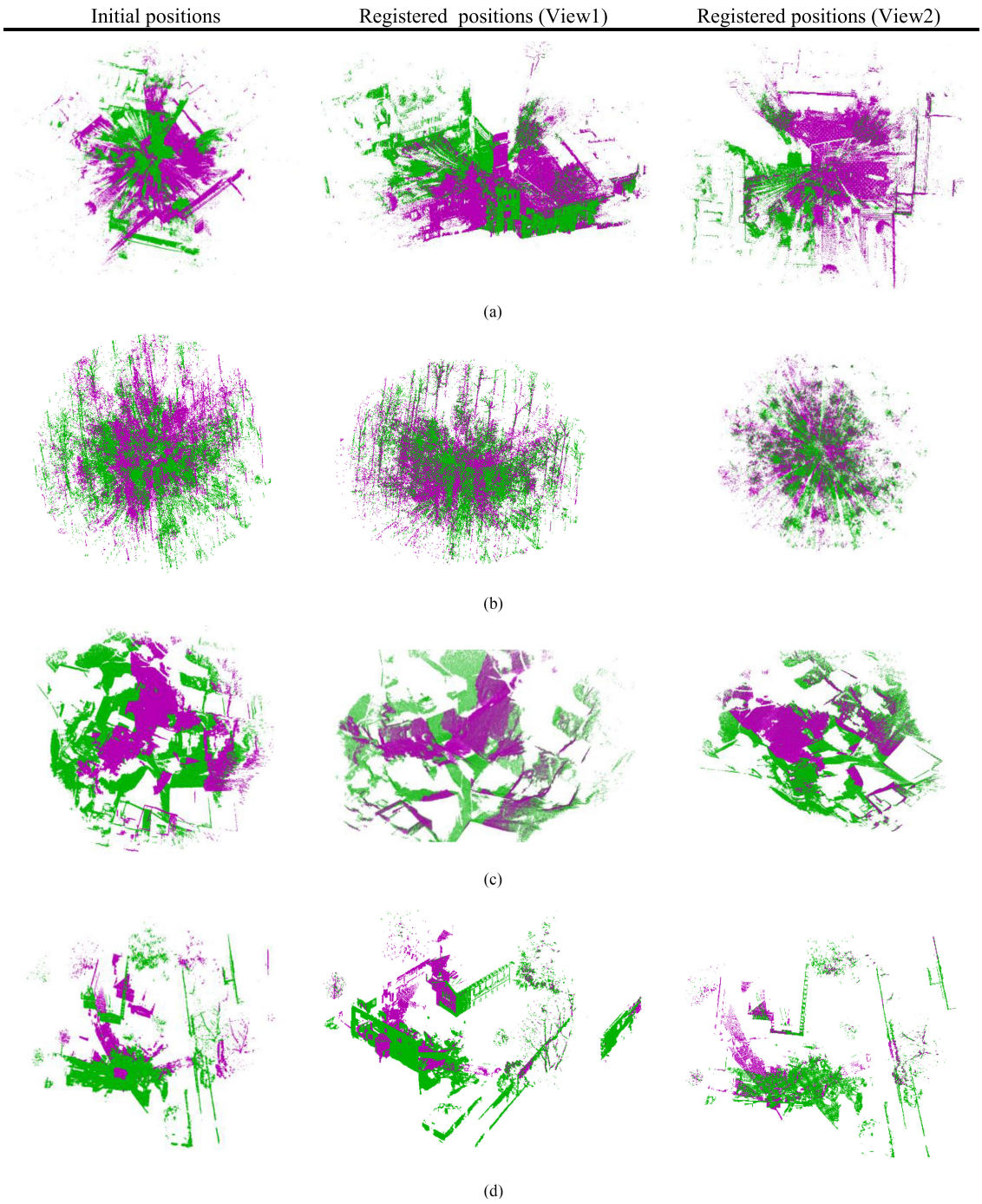



Fig. 10 Registration results for each pair of covered scans for all datasets used. (a) Arch. (b) Trees. (c) Courtyard. (d) Facede.

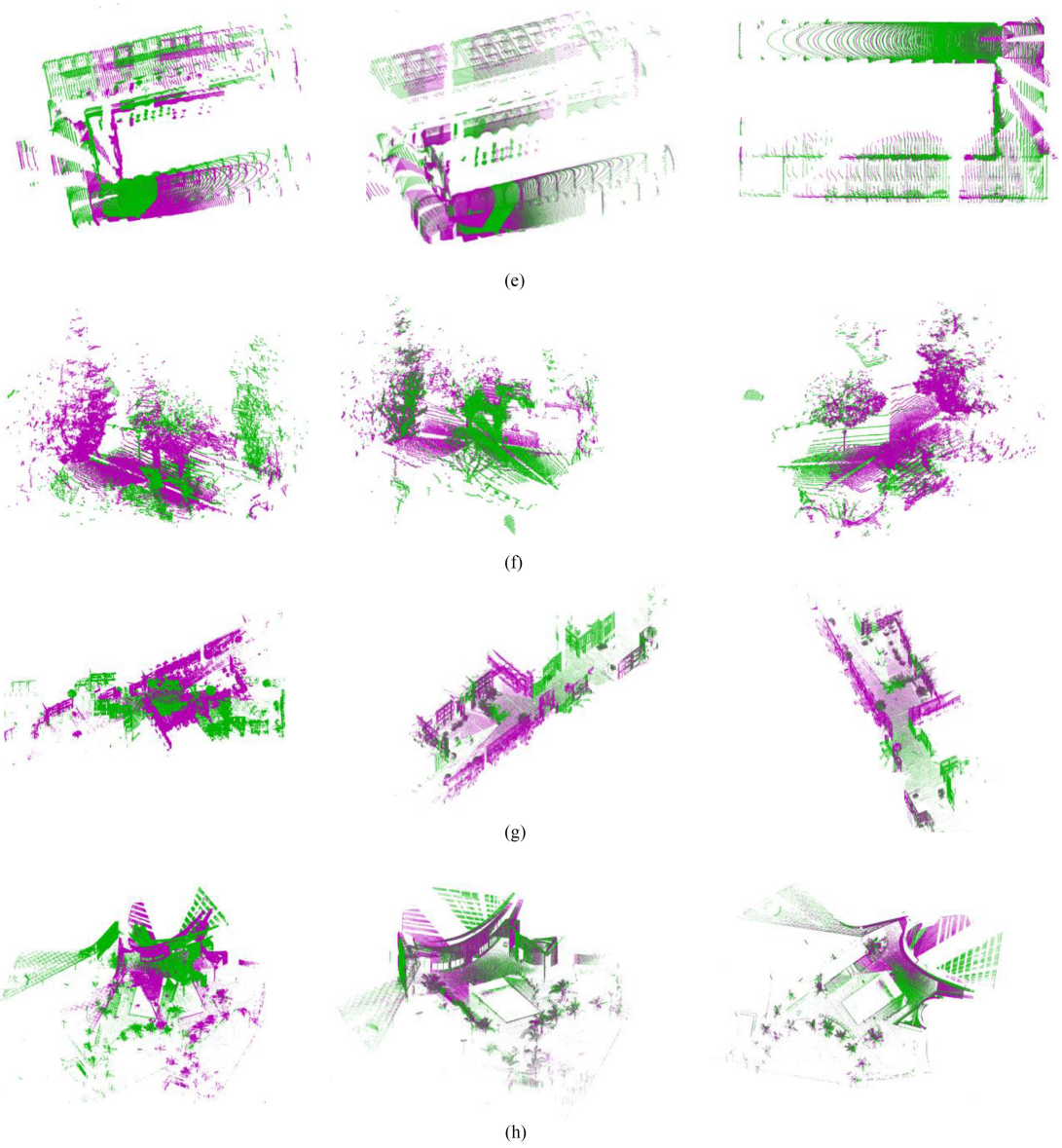


Fig. 10. *(Continued.)* Registration results for each pair of covered scans for all datasets used. (e) ETH hauptgebaude. (f) Gazebo. (g) RESSO (7c). (h) RESSO (7d).

The second criterion is a statistical criterion in which the root mean square distance (RMSD) is computed between the source point cloud after applying the registration algorithm $\boldsymbol{P}^s$ and the same point cloud in its true position $\boldsymbol{G}^s$. The RMSD is calculated as follows:

$$\text{RMSD} = \frac{\sum_{i=0}^{n} \|p_i^s - g_i^s\|_2}{n} \tag{9}$$

where $\|.\|$ is Euclidean distance and $n$ is the total number of points. It should be noted that the scale of the point clouds dataset used in this research is equal to each other.

The third criterion will be used as the percentage of completeness [24]. This criterion indicates how much of the point cloud registration was successful and whether it was possible to enter the fine registration stage. This criterion is defined as follows:

$$\text{Percentage complete} = \frac{N_a}{N} \times 100 \tag{10}$$

where $N$ is the total number of two adjacent point clouds for registration and $N_a$ is the number of correct registration.

### C. Setting the Parameters

The proposed descriptor has four parameters: neighborhood radius, number of rings, number of circles in each ring, and number of directional histograms. Neighborhood radius is an important and influential parameter that determines the descriptor scale [10]. Determining the neighborhood radius depends on the intended application. A large neighborhood radius affects computational performance and increases sensitivity to clutters and occlusions, and a small neighborhood radius will reduce

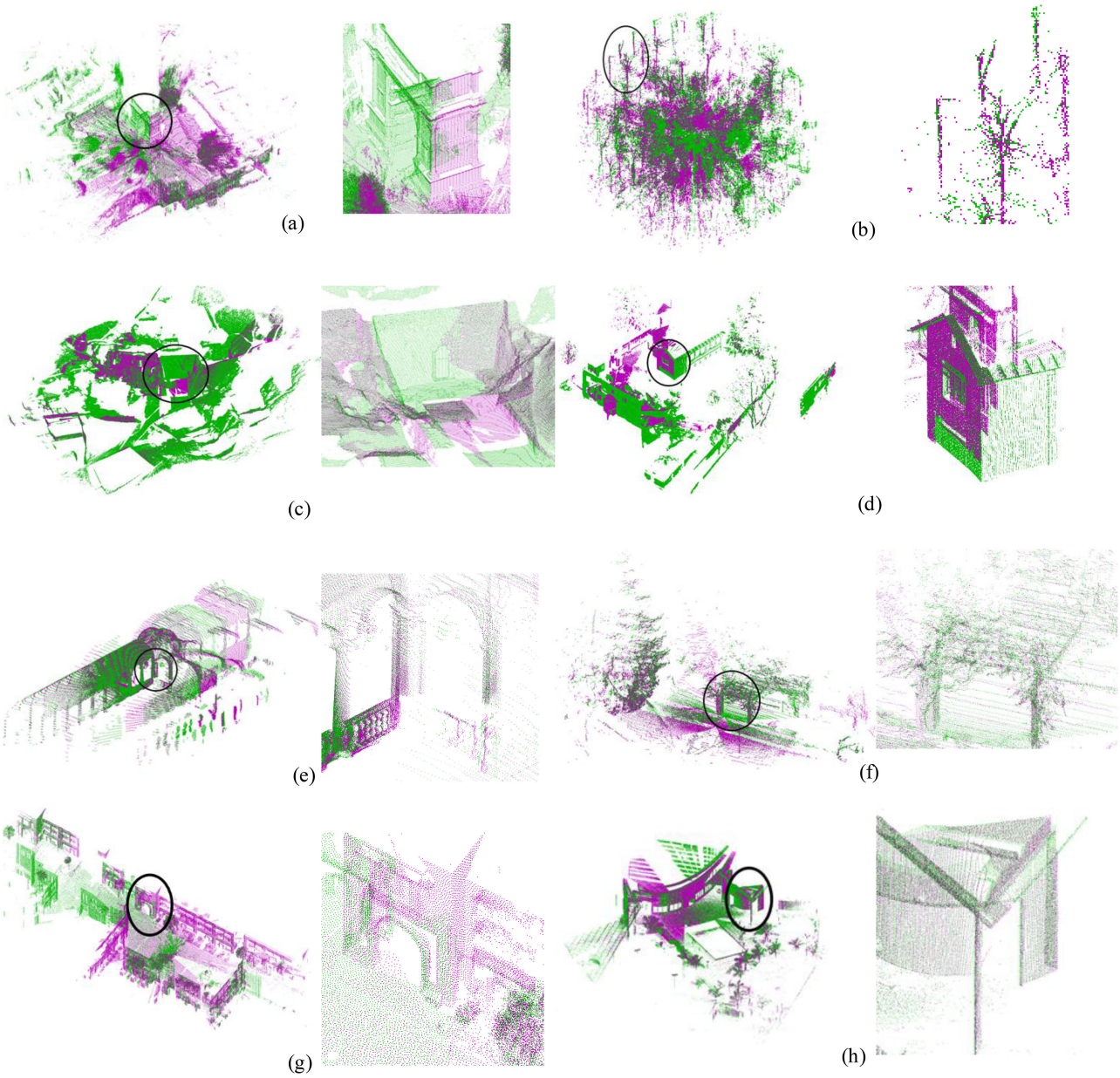



Fig. 11. Details of the point cloud registration.

the discrimination of the descriptors [11]. Many partitions for the descriptor will naturally increase the computation time and increase the amount of memory consumed, thus reducing the descriptor compression property. But, it will increase the discernment of the descriptor. If this division is considered low, many details will not be provided in the descriptor. Therefore, it is necessary to balance robustness, discrimination, compactness, and efficiency. The criterion used to evaluate the performance of the parameters is RMSD. The lower RMSD shows a higher performance of the algorithm.

In setting parameters, ETH hauptgebaude data was used and different values of a parameter were examined according to the constancy of other parameters. Details of setting the parameters are given in Table II. At first, 3-D SIFT keypoints are extracted in this data. Then the process of producing the descriptor and corresponding is done conforming to the stated approach. According to the practical experiments performed, the value of the neighborhood radius has been considered as 20mr [mr is the mean resolution (the mean distance points to the nearest point), and its unit is meter]. The results obtained from the descriptor parameters are presented in Fig. 8. Fig. 8(a)–(c) show the results of setting the $T$, $Q$, and $H$ parameters, respectively. In Fig. 8(c), the parameters represented by one value (e.g., 4-4-4) mean, that all directions divisions have the same number of bins in each ring. And for the parameters with three different values (e.g., 12-8-4), the divisions are based on the distance. The first value for the nearest ring and the last value for the farthest ring are considered. The test results show the values $Q = 3$, $T = 8$,

TABLE III
PARAMETER SETTINGS FOR FIVE FEATURE DESCRIPTORS

| | Support radius | Dimensionality | Length |
|---|---|---|---|
| **SHOT** | 20mr | 8 × 2 × 2 × 10 | 320 |
| **ROPS** | 20mr | 3 × 3 × 3 ×5 | 135 |
| **SPIN Image** | 20mr | 9 × 17 | 153 |
| **FPFH** | 20mr | 3 × 11 | 33 |
| **Proposed Method** | 20mr | 3 × (12+8×8+8×4) | 324 |

TABLE IV
AVERAGE ERRORS IN ALL DATA OF DIFFERENT DESCRIPTOR

| | Rotation Error (rad) | Translation Error (m) | RMSD(m) |
|---|---|---|---|
| SHOT | 0.7819 | 7.484 | 8.273 |
| ROPS | 0.3545 | 3.710 | 4.650 |
| SPIN Image | 0.2598 | 3.534 | 3.946 |
| FPFH | 0.9847 | 8.416 | 11.159 |
| SpinNet(3DMatch) | 0.1731 | 1.996 | 2.601 |
| SpinNet(KITTI) | 0.1182 | 1.386 | 1.689 |
| Proposed Method | 0.1045 | 1.980 | 2.039 |

TABLE V
RESULTS OF PERCENTAGE COMPLETE CRITERIA (%)

| Data / Method | Arch | Trees | Courtyard | Facade | ETH hauptgebaude | Gazebo | RESSO (7c) | RESSO (7d) |
|---|---|---|---|---|---|---|---|---|
| SHOT | 45.34 | 50.65 | 100 | 100 | 82.35 | 90.90 | 40 | 66.66 |
| ROPS | 41.65 | 39.76 | 100 | 100 | 70.58 | 83.33 | 100 | 100 |
| SPIN Image | 73.19 | 65.45 | 100 | 100 | 73.68 | 92.30 | 80 | 100 |
| FPFH | 25.43 | 0 | 100 | 90 | 54.54 | 65 | 40 | 100 |
| SpinNet | 90 | 57.5 | 100 | 100 | 89.47 | 100 | 100 | **100** |
| Proposed Method | 80.76 | 72.5 | 100 | 100 | 84.21 | 95 | 100 | 100 |

and $H = 12\text{-}8\text{-}4$ as the optimal values for the algorithm. The dimensions of the descriptor are equal to 324 according to the selected optimum parameters.

### *D. Point Cloud Registration*

The results of keypoint correspondence with the proposed method are displayed in Fig. 9, while Fig. 10 shows the coarse registration results for each pair of scans. In this figure, an image of the initial position point clouds is provided, in which purple is the source point cloud and green is the target point cloud. Then the registration results are shown in two different viewing angles. Furthermore, Fig. 11 presents the details of the point cloud registration by the proposed method. The inspected position is shown with a black circle in the overview image.

### *E. Comparison of Results With Traditional Descriptors*

The descriptor is compared with four popular traditional descriptors in 3-D point description. These descriptors include SHOT, ROPS, SPIN Image, and FPFH that are implemented in C++ (using PCL) [51]. The parameter settings for these feature descriptors are listed in Table III. The averages of all registration errors in all datasets and the results of completeness percentage criteria are presented in Tables IV and V, respectively. Furthermore, the introduced approach in [58] is used to evaluate the accuracy and precision of the results. They suggest using robust error statistics, namely the 50th, 75th, and 95th percentiles of empirical distributions of the errors, referred to as A50, A75, and A95. The results of this study are presented in Table VI. Moreover, the full distribution of rotation and translation errors in each data is shown in Fig. 12.

Table IV displays the average performance of each descriptor in the total data. For the suggested descriptor, the mean errors of rotation, translation, and RMSD are 0.1045(rad), 1.980(m), and 2.039(m), respectively. These results show that, on average, the proposed method has the highest performance in all criteria in all data among traditional descriptors. The results of Table V show that an appropriate percentage of coarse registration with our approach has been done successfully. For the courtyard, façade, RESSO (7c), and RESSO (7d) data, it shows 100% and in other data, it has high success rates compared to other traditional descriptors.

TABLE VI
QUANTILE STATISTICS OF REGISTRATION ERRORS

| | method | Arch | | | Trees | | | Courtyard | | | Facade | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | A50 | A75 | A95 | A50 | A75 | A95 | A50 | A75 | A95 | A50 | A75 | **A95** |
| Rotation Error (rad) | SHOT | 1.01 | 1.46 | 2.15 | 0.72 | 1.50 | 2.02 | 0.0022 | 0.0033 | 0.1232 | 0.0015 | 0.0021 | **0.1700** |
| | ROPS | 0.76 | 0.85 | 1.53 | 0.64 | 1.14 | 2.42 | 0.0030 | 0.0056 | 0.0075 | 0.0029 | 0.0047 | **0.0070** |
| | SPIN Image | **0.007** | **0.014** | 0.72 | 0.37 | 0.55 | 0.88 | **0.0015** | **0.0018** | **0.0035** | **0.001** | **0.0015** | 0.0020 |
| | FPFH | 0.24 | 1.30 | 2.73 | 1.72 | 2.68 | 2.75 | 0.0271 | 0.11 | 0.20 | 0.0245 | 0.048 | **2.39** |
| | Our | 0.015 | 0.016 | **0.33** | **0.013** | **0.41** | **0.78** | 0.0020 | 0.0023 | 0.0037 | 0.0017 | 0.0026 | **0.0031** |
| Translation Error (m) | SHOT | 2.49 | 12.54 | 17.63 | 9.89 | 17.51 | 18.22 | 0.066 | 0.092 | 30.23 | **0.020** | **0.026** | 0.041 |
| | ROPS | 4.21 | 11.45 | 23.33 | 7.47 | 11.49 | 15.31 | 0.061 | 0.121 | 0.151 | 0.034 | 0.054 | **0.108** |
| | SPIN Image | 0.13 | 0.21 | 25.72 | 0.28 | 13.23 | 28.14 | **0.038** | **0.067** | 0.085 | 0.021 | 0.027 | **0.043** |
| | FPFH | 3.37 | 20.22 | 23.11 | 11.29 | 14.14 | 18.82 | 0.587 | 0.874 | 1.573 | 0.315 | 0.666 | **39.25** |
| | Our | **0.12** | **0.15** | **16.30** | **0.11** | **10.11** | **15.00** | 0.047 | 0.075 | **0.081** | 0.023 | 0.036 | **0.045** |
| RMSD(m) | SHOT | 2.79 | 6.35 | 21.22 | 16.33 | 17.74 | 23.27 | 0.101 | 0.120 | 32.85 | **0.025** | **0.032** | **0.060** |
| | ROPS | 1.95 | 8.74 | 23.85 | 20.87 | 17.17 | 25.99 | 0.118 | 0.167 | 0.224 | 0.043 | 0.068 | **0.086** |
| | SPIN Image | **0.17** | 0.43 | 32.76 | 0.14 | 14.96 | 25.06 | **0.074** | **0.079** | **0.092** | 0.026 | 0.034 | 0.036 |
| | FPFH | 5.38 | 9.44 | 34.23 | 25.13 | 36.08 | 39.35 | 0.821 | 0.992 | 5.537 | 0.305 | 0.695 | **36.70** |
| | Our | 0.30 | **0.35** | **17.23** | **0.12** | **2.42** | **15.35** | 0.082 | 0.101 | 0.130 | 0.032 | 0.036 | **0.045** |
| | | **ETH hauptgebaude** | | | **Gazebo** | | | **RESSO (7c)** | | | **RESSO (7d)** | | |
| | | A50 | A75 | A95 | A50 | A75 | A95 | A50 | A75 | A95 | A50 | A75 | A95 |
| Rotation Error (rad) | SHOT | 0.0063 | 0.0082 | 0.0132 | 0.0067 | 0.0111 | 0.8444 | 1.83 | 2.9 | 2.94 | 0.143 | 0.35 | 0.55 |
| | ROPS | 0.0056 | 0.0070 | **0.0098** | 0.0096 | 0.0126 | **0.8625** | 0.0096 | 0.0312 | 0.08 | 0.0128 | 0.038 | 0.058 |
| | SPIN Image | 0.0060 | 0.0098 | 0.0257 | 0.0060 | **0.0079** | **0.961** | 0.0117 | 0.026 | 2.61 | 0.0044 | 0.006 | 0.008 |
| | FPFH | 0.0259 | **0.0657** | 0.1917 | 0.1010 | 0.2034 | **1.053** | 1.65 | 2.99 | 3.11 | 0.0054 | 0.007 | 0.009 |
| | Our | **0.0052** | 0.0075 | 0.0100 | **0.0057** | 0.0121 | **0.8520** | **0.0059** | **0.0067** | **0.0136** | **0.002** | **0.0032** | **0.0045** |
| Translation Error (m) | SHOT | 0.062 | 0.191 | 2.86 | **0.016** | 0.0755 | **4.86** | 15.98 | 16.44 | 26.22 | 0.45 | 1.41 | 2.31 |
| | ROPS | 0.094 | 2.14 | 2.85 | 0.038 | 0.10 | **7.45** | 0.237 | 0.274 | 0.350 | 0.35 | 0.66 | 0.73 |
| | SPIN Image | 0.084 | 0.789 | 2.45 | 0.020 | **0.029** | **2.12** | 0.30 | 0.34 | 10.35 | 0.071 | 0.11 | 0.162 |
| | FPFH | 0.582 | 1.42 | 2.86 | 1.22 | 1.73 | **7.32** | 11.48 | 17.2 | 23.4 | 0.095 | 0.190 | 0.284 |
| | Our | **0.057** | **0.112** | **2.16** | 0.026 | 0.064 | 1.46 | **0.169** | **0.208** | **0.262** | **0.041** | **0.042** | **0.043** |
| RMSD(m) | SHOT | 0.089 | 0.105 | 3.40 | 0.040 | 0.073 | **2.20** | 14.76 | 18.41 | 32.2 | 0.94 | 2.20 | 3.25 |
| | ROPS | 0.098 | 0.248 | 4.98 | 0.049 | 0.068 | **2.76** | 0.28 | 0.38 | 0.47 | 0.40 | 1.05 | 1.55 |
| | SPIN Image | 0.092 | 0.247 | 4.96 | 0.042 | **0.065** | 1.70 | 0.51 | 0.56 | 12.1 | 0.167 | 0.191 | 0.21 |
| | FPFH | 0.535 | 1.570 | 4.09 | 0.861 | 1.29 | **4.00** | 13.77 | 18.98 | 27.3 | 0.173 | 0.26 | 0.34 |
| | Our | **0.0822** | **0.099** | **1.47** | **0.036** | 0.067 | **1.80** | **0.28** | **0.34** | **0.41** | **0.083** | **0,10** | **0.12** |

Bold values in indicate the best performance.

The proposed algorithm has provided appropriate results, regardless of the data type. The two data, Arch and Trees, are considered the most challenging data among comparative data in which the performance of descriptors is lower than other data. The Arch dataset has relatively little coverage, and the trees dataset has a high density, high noise level, and medium overlap. According to Table VI, in the Arch data, the suggested method has the highest performance in all three 50th, 75th, and 95th percentiles in the translation error and the amount of rotation error and RMSE are also appropriate. The SPIN image descriptor performed suitable in this data, but other descriptors failed in most experiments. In the trees dataset, the proposed method has the best performance in all criteria, which shows that the performance of our approach is appropriate in noisy data. However, in the courtyard and facade data, the performance of our method was acceptable and near to other descriptors but provided less performance than other descriptors. In the courtyard data, we do not see any vertical objects, and the only existing complication is the ground surface and also there are high overlap and low noise level in these data. These cases have made this data less challenging.

The results in ETH and Gazebo data also show the high performance of our method. In the ETH data, we see duplicate structures, but the proposed method has the highest performance in all percentiles in the translation and RMSD criteria. This result means that the algorithm has performed well against repetitive patterns. Moreover, in gazebo data, we see the highest performance of our method in the *A50* percentile of the rotation

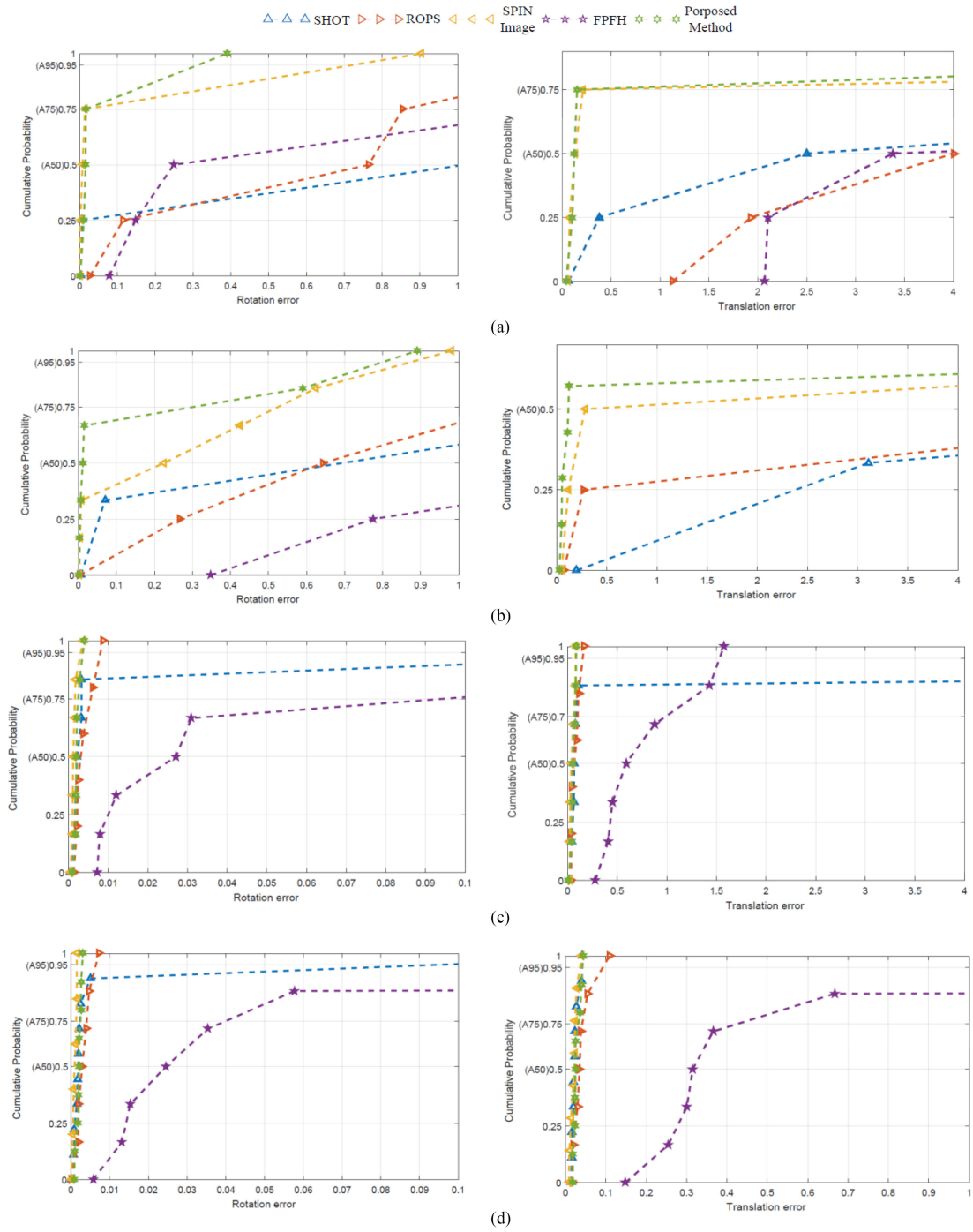



Fig. 12. Distribution of (left) rotation and (right) translation errors for all data. (a) Arch. (b) Trees. (c) Courtyard. (d) Facede.

criterion and in the *A50* percentiles of the RMSD criteria. The errors distribution in Fig. 12 shows that most descriptors failed in the Arch and Trees data, while the proposed method provided the highest performance in both data. In the other data, the descriptors performed almost closely, but in the ETH data with repetitive patterns, the performance of the proposed method performed better than the other descriptor in the translations criterion.

In RESSO (7d) and RESSO (7c) data, the best performance belongs to the proposed method and has provided the highest accuracy in all criteria. In this data, descriptor ROPS provides the second performance. The FPFH descriptor gave the weakest results and failed in many experiments. In general, this descriptor has low differentiation and is unstable to high noise.

The results show that the proposed descriptor has provided good results, especially in more noisy and challenging data. One of the reasons for the superiority of our method is the DAISY structure used in the descriptor. The geometric

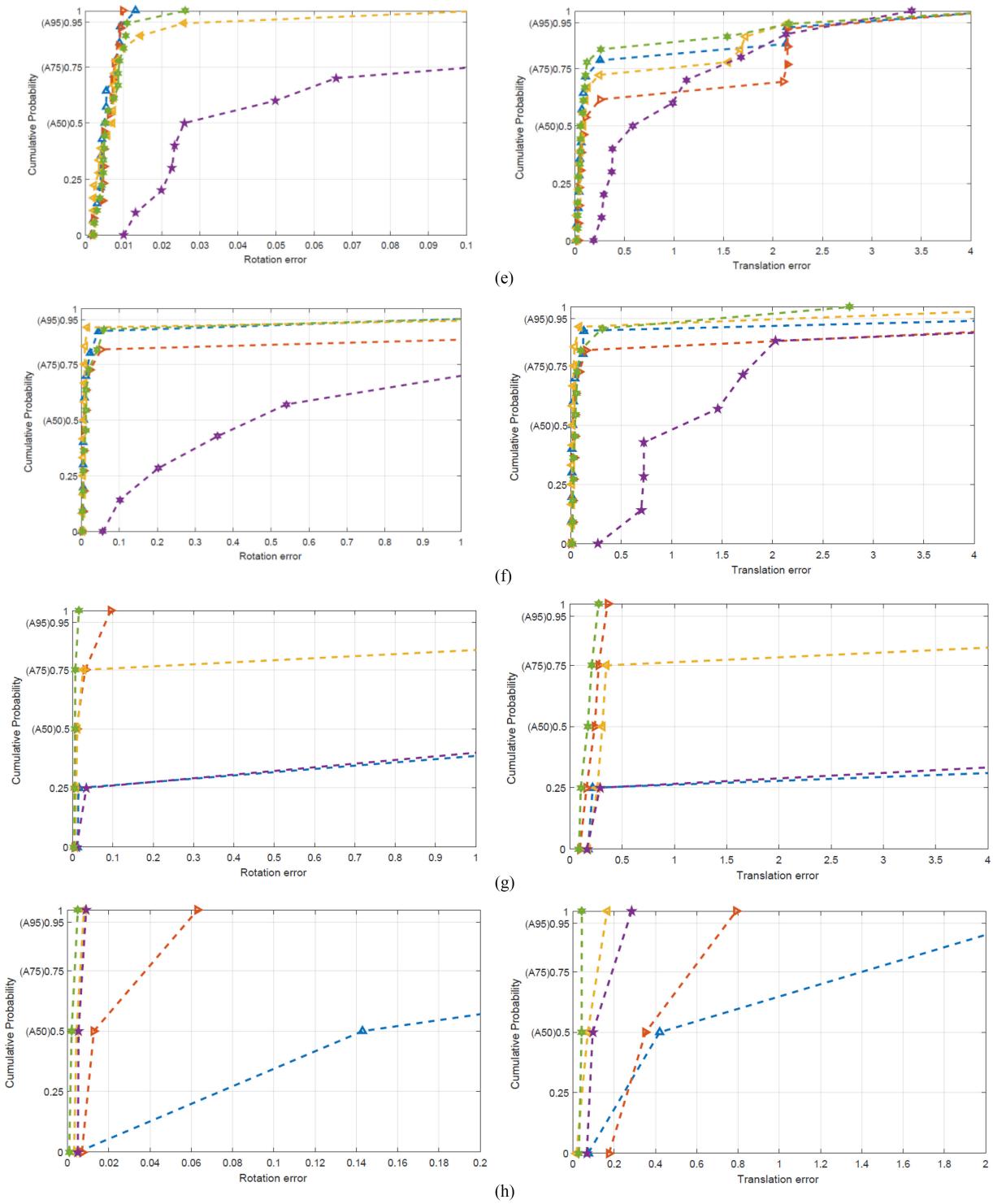



Fig. 12. *(Continued.)* Distribution of (left) rotation and (right) translation errors for all data. (e) ETH hauptgebaude. (f) Gazebo. (g) RESSO (7c). (h) RESSO (7d).

arrangement of this descriptor reduces the effect of noise and displacements of points in adjacent bins due to the use of overlapping circles. These displacements due to various distortions are usually seen in the point cloud data. On the other hand, determining the LRF is a very influential part of the production of descriptors. LRFs are also highly impacted in the face of this type of displacement errors. A partition overlay approach was adopted to define the $x$-axis to reduce this displacement error.

### F. Comparison of Results With SpinNet Descriptors

Given that deep learning methods have recently been considered, the proposed descriptor has also been compared with the SpinNet deep learning method. For this purpose, two

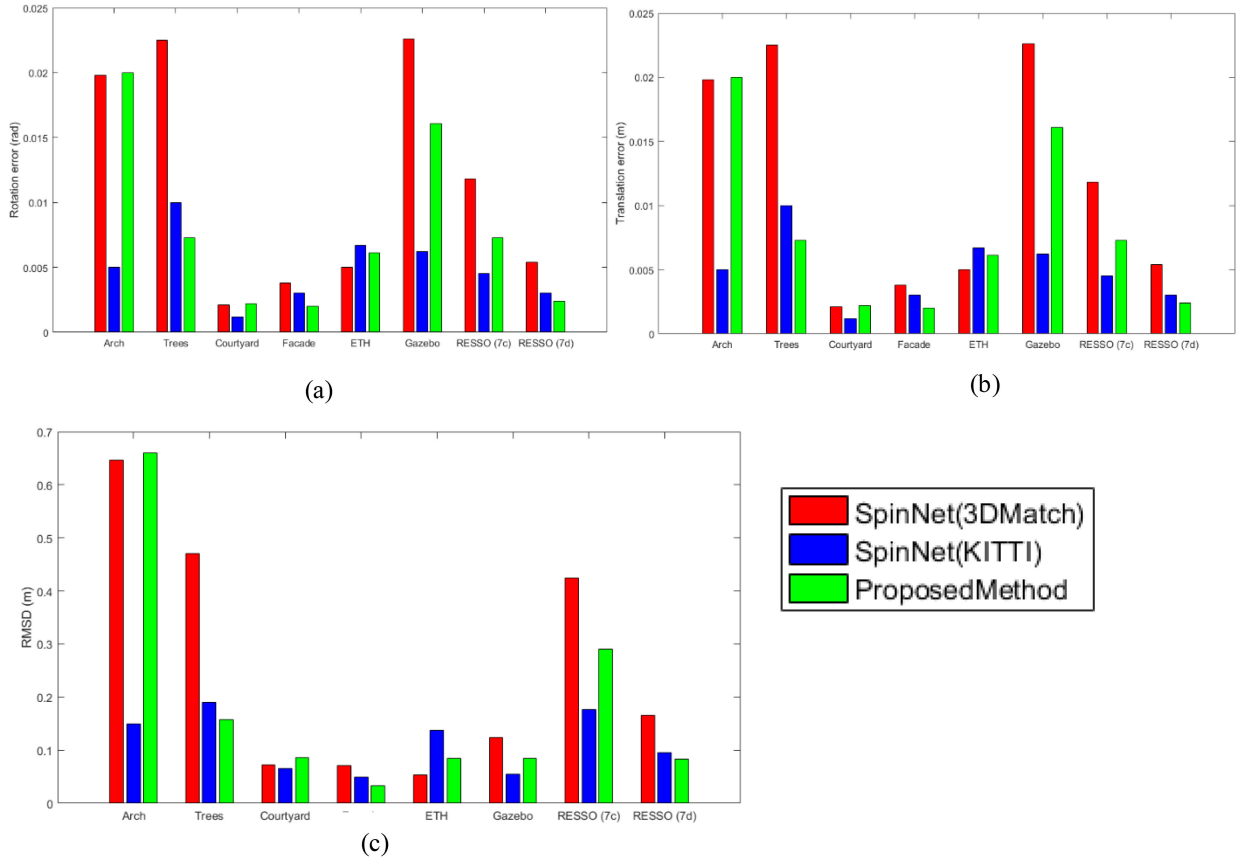



Fig. 13. Results of comparison with the deep learning method.

pretrained models have been used. The models trained on the indoor 3DMatch [59] and outdoor KITTI [60] datasets. We keep the default parameters in the SpinNet algorithm and only the radius parameter is determined due to the varying point densities in different datasets. The results obtained from comparing the performance of the proposed method with the SpinNet approach by two pre-trained models are shown in Fig. 13. This figure considers the average values of rotation error, translation error, and RMSD criteria for successful experiments in coarse registration of each data.

Fig. 13 shows the superiority of the proposed method in the three data Trees, Facade, and RESSO (7d), compared to both pretrained models of the SpinNet. The performance of the two approaches in the Courtyard and ETH data is very close, and with a slight difference, the SpinNet algorithm performed better. The proposed method performed better than SpinNet (3DMatch) in all comparisons except ETH data. But in Arch, Gazebo, and RESSO (7c) data, the SpinNet (KITTI) has been better. According to Table IV, the proposed method, on average, performed better than the SpinNet (3DMatch) method in the whole data but Compared with SpinNet (KIITI) method only has performed better in the Rotation Error criterion. In Table V, both algorithms in the Courtyard, Facade, RESSO (7c), and RESSO (7d) data have succeeded in all coarse registration experiments. The proposed method for the Tree data offers much higher performance. And in the Arch, ETH, and Gazebo data, the SpinNet is higher, but this superiority is significant only in the Arch data.

According to the obtained results, the proposed method has shown very high performance in the point clouds coarse registration without using any preknowledge. The proposed method is introduced as the superior algorithm in most experiments compared to one of the two pretrained models of the SpinNet algorithm. This means that although the deep learning approach has yielded promising results, they are highly dependent on training data. Due to the unavailability of these training data, the performance of these methods in real data requires a comprehensive evaluation.

## V. Conclusion and Future Work

This research presented an accurate, stable, and effective framework for point cloud registration. The proposed method included a novel 3-D descriptor with a new geometric arrangement inspired by the 2-D DAISY descriptor and an improved LRF. In this research, an approach was adopted to reduce the effect of point displacement error in local features descriptors. This error reduces the discriminative in descriptors that use spatial information. The local descriptor proposed was divided into circles in different rings with common overlays on 2-D plans. This geometric arrangement increased the robustness and discrimination of the descriptor. Furthermore, the directional histogram in each circle was also suggested as the feature. We considered an approach to creating a tradeoff between the discriminative and efficacy. In this method, directional divisions have been

considered for each ring in proportion to its distance to the keypoint. Furthermore, in determining the dominant direction of the $x$-axis, a partition coverage method was adopted to reduce the effect of point displacement error in determining the LRF. Based on this descriptor was proposed a point cloud registration framework. Registration results were evaluated on point clouds from three benchmarks and were compared with some popular traditional descriptors. The results of point cloud registration showed that our descriptor performed well, especially in noisy and more challenging data wherein the displacement error was more likely to occur.

In addition, the proposed method was compared with the SpinNet deep learning method with two pretrained models. The proposed method provided the best performance in some comparisons. In most comparisons, it had a better performance than one of the pretrained models. Therefore, despite the good results of the method SpinNet, it must be kept in mind that this method is dependent on training data. In contrast, the proposed method has provided admirable results without preknowledge and is completely automatic.

In future research, in addition to the robustness and discrimination of the descriptor, the efficiency of the algorithm can be evaluated. For this purpose, some approaches such as TEASER [61] or GESAC [62] can be adapted to eliminate wrong correspondences. Furthermore, the performance of the proposed descriptor can be evaluated on data with small overlaps and even data from various sensors. In evaluating the performance of the descriptor, it is suggested that other criteria such as feature matching recall be considered.

## ACKNOWLEDGMENT

The authors would like to acknowledge the help of Dr. N. Li in the sections related to deep learning.

## REFERENCES

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

[2] G. Esposito *et al.*, "Multitemporal monitoring of a coastal landslide through SFM-derived point cloud comparison," *Photogrammetric Rec.*, vol. 32, no. 160, pp. 459–479, 2017.

[3] H. Zhao, M. Tang, and H. Ding, "HoPPF: A novel local surface descriptor for 3D object recognition," *Pattern Recognit.*, vol. 103, 2020, Art. no. 107272.

[4] P. Kim, J. Chen, and Y. K. Cho, "SLAM-driven robotic mapping and registration of 3D point clouds," *Automat. Construction*, vol. 89, pp. 38–48, 2018.

[5] N. Li, O. Kähler, and N. Pfeifer, "A comparison of deep learning methods for airborne lidar point clouds classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 6467–6486, 2021.

[6] D. Fontanelli, L. Ricciato, and S. Soatto, "A fast ransac-based registration algorithm for accurate localization in unknown environments using lidar measurements," in *Proc. IEEE Int. Conf. Automat. Sci. Eng.*, 2007, pp. 597–602.

[7] J.-C. Cheng and H.-S. Don, "A graph matching approach to 3-D point correspondences," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 5, no. 3, pp. 399–412, 1991.

[8] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 766–782.

[9] R. Huang, Y. Xu, W. Yao, L. Hoegner, and U. Stilla, "Robust global registration of point clouds by closed-form solution in the frequency domain," *ISPRS J. Photogramm. Remote Sens.*, vol. 171, pp. 310–329, 2021.

[10] 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.

[11] 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, pp. 63–86, 2013.

[12] 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.

[13] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2009, pp. 3212–3217.

[14] J. Yang, Z. Cao, and Q. Zhang, "A fast and robust local descriptor for 3D point cloud registration," *Inf. Sci.*, vol. 346, pp. 163–179, 2016.

[15] A. Sedaghat and H. Ebadi, "Remote sensing image matching based on adaptive binning SIFT descriptor," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 10, pp. 5283–5293, Oct. 2015.

[16] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[17] F. Ghorbani, H. Ebadi, and A. Sedaghat, "Geospatial target detection from high-resolution remote-sensing images based on PIIFD descriptor and salient regions," *J. Indian Soc. Remote Sens.*, vol. 47, pp. 879–891, 2019.

[18] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279.

[19] X.-F. Hana, J. S. Jin, J. Xie, M.-J. Wang, and W. Jiang, "A comprehensive review of 3D point cloud descriptors," 2018, *arXiv:1802.02297*.

[20] J. Yang, Y. Xiao, and Z. Cao, "Toward the repeatability and robustness of the local reference frame for 3D shape matching: An evaluation," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3766–3781, Aug. 2018.

[21] J. Yang, Q. Zhang, and Z. Cao, "The effect of spatial information characterization on 3D local feature descriptors: A quantitative evaluation," *Pattern Recognit.*, vol. 66, pp. 375–391, 2017.

[22] I. Stamos and M. Leordeanu, "Automated feature-based range registration of urban scenes of large scale," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. II, Jun. 2003, pp. 555–561.

[23] T. Weber, R. Hänsch, and O. Hellwich, "Automatic registration of unordered point clouds acquired by kinect sensors using an overlap heuristic," *ISPRS J. Photogramm. Remote Sens.*, vol. 102, pp. 96–109, 2015.

[24] B. Yang, Z. Dong, F. Liang, and Y. Liu, "Automatic registration of large-scale urban scene point clouds based on semantic feature points," *ISPRS J. Photogramm. Remote Sens.*, vol. 113, pp. 43–58, 2016.

[25] D. Zai *et al.*, "Pairwise registration of TLS point clouds using covariance descriptors and a non-cooperative game," *ISPRS J. Photogramm. Remote Sens.*, vol. 134, pp. 15–29, 2017.

[26] 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. Photogramm. Remote Sens.*, vol. 144, pp. 61–79, 2018.

[27] Z. Cai, T.-J. Chin, A. P. Bustos, and K. Schindler, "Practical optimal registration of terrestrial LiDAR scan pairs," *ISPRS J. Photogramm. Remote Sens.*, vol. 147, pp. 118–131, 2019.

[28] X. Ge and H. Hu, "Object-based incremental registration of terrestrial point clouds in an urban environment," *ISPRS J. Photogramm. Remote Sens.*, vol. 161, pp. 218–232, 2020.

[29] J. Li, Q. Hu, and M. Ai, "Point cloud registration based on one-point ransac and scale-annealing biweight estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9716–9729, Nov. 2021.

[30] J. Li, "A practical o (N2) outlier removal method for point cloud registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Mar. 2021, doi: 10.1109/TPAMI.2021.3065021.

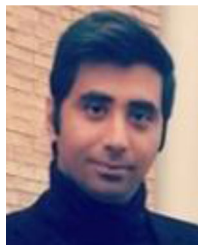
[31] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[32] H. Deng, T. Birdal, and S. Ilic, "Ppfnet: Global context aware local features for robust 3d point matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 195–205.

[33] Z. Gojcic, C. Zhou, J. D. Wegner, and A. Wieser, "The perfect match: 3d point cloud matching with smoothed densities," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5545–5554.

[34] S. Ao, Q. Hu, B. Yang, A. Markham, and Y. Guo, "SpinNet: Learning a general surface descriptor for 3D point cloud registration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11753–11762.

[35] Z. Dong *et al.*, "Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark," *ISPRS J. Photogramm. Remote Sens.*, vol. 163, pp. 327–342, 2020.
[36] 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, 2008, pp. 1–10.
[37] J. Huang, T.-H. Kwok, and C. Zhou, "V4PCS: Volumetric 4PCS algorithm for global registration," *J. Mech. Des.*, vol. 139, no. 11, 2017, Art. no. 111403.
[38] Y. Xu, R. Boerner, W. Yao, L. Hoegner, and U. Stilla, "Pairwise coarse registration of point clouds in urban scenes using voxel-based 4-planes congruent sets," *ISPRS J. Photogramm. Remote Sens.*, vol. 151, pp. 106–123, 2019.
[39] A. Myronenko, X. Song, and M. A. Carreira-Perpinán, "Non-rigid point set registration: Coherent point drift," *Adv. Neural Inf. Process. Syst.*, vol. 19, 2007, Art. no. 1009.
[40] Y. Zang and R. Lindenbergh, "An improved coherent point drift method for TLS point cloud registration of complex scenes," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 42, pp. 1169–1175, 2019.
[41] A. Walicka, N. Pfeifer, A. Borkowski, and G. Jóźków, "An automatic method for the measurement of coarse particle movement in a mountain riverbed," *Measurement*, vol. 174, 2021, Art. no. 109029.
[42] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, "Challenging data sets for point cloud registration algorithms," *Int. J. Robot. Res.*, vol. 31, pp. 1705–1711, 2012.
[43] S. Chen, L. Nan, R. Xia, J. Zhao, and P. Wonka, "PLADE: A plane-based descriptor for point cloud registration with small overlap," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2530–2540, Apr. 2020.
[44] E. Tola, V. Lepetit, and P. Fua, "Daisy: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.
[45] J. Novatnack and K. Nishino, "Scale-dependent/invariant local 3D shape descriptors for fully automatic registration of multiple sets of range images," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 440–453.
[46] A. Mian, M. Bennamoun, and R. Owens, "On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes," *Int. J. Comput. Vis.*, vol. 89, no. 2/3, pp. 348–361, 2010.
[47] C. S. Chua and R. Jarvis, "Point signatures: A new representation for 3d object recognition," *Int. J. Comput. Vis.*, vol. 25, no. 1, pp. 63–85, 1997.
[48] A. Petrelli and L. Di Stefano, "On the repeatability of the local reference frame for partial shape matching," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2244–2251.
[49] A. Petrelli and L. Di Stefano, "A repeatable and efficient canonical reference for surface matching," in *Proc. 2nd Int. Conf. 3D Imag., Model., Process., Visual. Transmiss.*, 2012, pp. 403–410.
[50] Y. Wang, B. Yang, Y. Chen, F. Liang, and Z. Dong, "JoKDNet: A joint keypoint detection and description network for large-scale outdoor TLS point clouds registration," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 104, 2021, Art. no. 102534.
[51] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 1–4.
[52] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.
[53] Z. Jiao, R. Liu, P. Yi, and D. Zhou, "A point cloud registration algorithm based on 3d-sift," in *Transactions on Edutainment XV*. Berlin, Germany: Springer, 2019, pp. 24–31.
[54] P. Stancelova, E. Sikudova, and Z. Cernekova, "3D Feature detector-descriptor pair evaluation on point clouds," in *Proc. 28th Eur. Signal Process. Conf.*, 2021, pp. 590–594.
[55] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, "Surface feature detection and description with applications to mesh matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 373–380.
[56] P. H. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *Int. J. Comput. Vis.*, vol. 24, pp. 271–300, 1997.
[57] T. Petricek and T. Svoboda, "Point cloud registration from local feature correspondences—Evaluation on challenging datasets," *PLoS One*, vol. 12, 2017, Art. no. e0187943.
[58] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Auton. Robots*, vol. 34, no. 3, pp. 133–148, 2013.
[59] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3D match: Learning local geometric descriptors from rgb-d reconstructions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1802–1811.
[60] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the Kitti vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[61] H. Yang, J. Shi, and L. Carlone, "Teaser: Fast and certifiable point cloud registration," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 314–333, Apr. 2021.
[62] J. Li, Q. Hu, and M. Ai, "GESAC: Robust graph enhanced sample consensus for point cloud registration," *ISPRS J. Photogramm. Remote Sens.*, vol. 167, pp. 363–374, 2020.



**Fariborz Ghorbani** received the B.Sc. degree in geomatics engineering from the Geomatics College, Iran National Cartographic Center, Tehran, Iran, in 2014, and the M.Sc. degree in photogrammetry engineering in 2017 from the K. N. Toosi University of Technology, Tehran, Iran, where he is currently working toward the Ph.D. degree in photogrammetry and remote sensing.

His research interests include, image processing and pattern recognition, feature extraction, machine learning, point cloud registration, classification and segmentation of point clouds, and UAV photogrammetry.

**Hamid Ebadi** received the Ph.D. degree in geomatics engineering from the University of Calgary, Calgary, AB, Canada, in 1997.

He is currently a Professor with the Department of Photogrammetry and Remote Sensing, Faculty of Geodesy and Geomatics Engineering, Khajeh Nasir Toosi University of Technology, Tehran, Iran, where he is also a member of the Center of Excellence for Geospatial Information Technology. His research interests include digital photogrammetry, information extraction from high-resolution satellite and aerial imagery, and the integration of photogrammetry and geospatial information systems.

**Amin Sedaghat** received the Ph.D. degree in photogrammetry engineering from the Khajeh Nasir (K.N.) Toosi University of Technology, Tehran, Iran, in 2015.

He is currently an Assistant Professor with the Department of Geomatics Engineering, Faculty of Civil Engineering, University of Tabriz, Tabriz, Iran. His research interests include remote sensing image processing and digital photogrammetry.

**Norbert Pfeifer** was born in Vienna, Austria, in 1971. He received the Dipl. Ing. and Ph.D. degrees in surveying engineering from Technische Universität Wien (TU Wien), Vienna, in 1997 and 2002, respectively.

From 2003 to 2006, he was Research Assistant and Assistant Professor with TU Delft, The Netherlands. In 2006, he was a Lecturer with the Department of Geography, the University of Innsbruck, Innsbruck, Austria, and a Senior Researcher with the Centre for Natural Hazard Management, alp-S, in Innsbruck. Later in 2006, he took the position of a Professor with TU Wien in Photogrammetry. He has coauthored more than 100 articles in journals and six books. His research interests include LiDAR signal processing, calibration of airborne and terrestrial laser scanning, classification and segmentation of Lidar point clouds, 3-D modeling, and application of point clouds in the environmental sciences.

Dr. Pfeifer is a Member of the International Society of Photogrammetry and Remote Sensing.