

Review Article

A Tutorial Review on Point Cloud Registrations: Principle, Classification, Comparison, and Technology Challenges

Leihui Li ^{1,2}, Riwei Wang,³ and Xuping Zhang ²

¹*School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China*

²*Department of Mechanical and Production Engineering, Aarhus University, Aarhus, Denmark*

³*Wenzhou University Oujiang College, Wenzhou, Zhejiang, China*

Correspondence should be addressed to Xuping Zhang; xuzh@mpe.au.dk

Received 26 March 2021; Accepted 8 July 2021; Published 29 July 2021

Academic Editor: Paolo Spagnolo

Copyright © 2021 Leihui Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A point cloud as a collection of points is poised to bring about a revolution in acquiring and generating three-dimensional (3D) surface information of an object in 3D reconstruction, industrial inspection, and robotic manipulation. In this revolution, the most challenging but imperative process is point cloud registration, i.e., obtaining a spatial transformation that aligns and matches two point clouds acquired in two different coordinates. In this survey paper, we present the overview and basic principles, give systematical classification and comparison of various methods, and address existing technical problems in point cloud registration. This review attempts to serve as a tutorial to academic researchers and engineers outside this field and to promote discussion of a unified vision of point cloud registration. The goal is to help readers quickly get into the problems of their interests related to point cloud registration and to provide them with insights and guidance in finding out appropriate strategies and solutions.

1. Introduction

With the advent of low-cost three-dimensional (3D) imaging devices and the development of professional 3D acquisition techniques, more and more research efforts have been attracted toward the 3D point cloud from industry to academia in recent years. In contrast to a 2D image, a 3D point cloud data contain much more information of an object or a scene, and hence, it can provide a better understanding and description of the real world. Specifically, the point cloud can be used to measure the distance between objects in the current frame, achieve an accurate representation of the shape of an object, and so on. Therefore, 3D point clouds have been widely produced and processed to provide accurate and fast descriptions of the 3D geometries of objects in various applications [1], such as 3D model reconstruction, geometry quality inspection, and robotic manipulation.

However, it is impossible to obtain all the point cloud data of an object or a scene at once in practice. Therefore, we

usually move the 3D acquisition device around a target object to finish a throughout scan, such that retrieving multiple point clouds at different viewpoints for better rebuilding the 3D environment or recovering the 3D shape of an object. It is critically important to capture these point clouds in such a way because (1) the Field of View (FoV) of a depth camera usually is not big enough to capture the massive scene, such as the forest and city, and (2) the quality of a single-achieved point cloud is generally not acceptable. For example, due to the low resolution of a camera, we need to fuse more frames into a global scene. In addition, to overcome the holes in the point cloud that are mainly caused by the reflection of the light on an object's surface, multiple frames at different viewpoints are required. In short, to obtain an accurate point cloud, the practical approach is to collect multiple point clouds in an appropriate manner and integrate them together. Furthermore, it is required to know the orientation of a scanned object represented by its point cloud data in its local coordinate. We have to match a standard point cloud in the dataset with the scanned point

cloud. The key technology that makes several point clouds fused into a complete point cloud with a common coordinate system is called point cloud registration, also called surface registration, in which point clouds are analyzed to represent the real-world surface of the object. It is a key step for applications and tasks, such as autonomous driving [2–5], 3D reconstruction [6–10], simultaneous localization and mapping (SLAM) [11–14], and virtual and augmented reality (VR/AR) [15–17]. In the application of point cloud registration, each point cloud acquired from different viewpoints is measured within their local coordinates. Therefore, the objective of registration is basically to calculate the transformation (translational and rotational) between point clouds to transfer these point clouds into a common mapping coordinate system. According to the number of the input data, point cloud registration can be divided into pairwise registration and multiview registration [18]. The pairwise registration particularly focuses on two point clouds and estimates the transformation between both of them, while the multiview registration, also called groupwise [19], performs more than two point clouds simultaneously, and the transformation of each scan to the reference data is calculated. According to the accuracy performance of registration, it can be divided into two stages: coarse registration and fine registration in the task of point cloud registration [20]. The coarse registration is intended to provide a rough transformation in terms of accuracy with an arbitrary initial position, and the fine registration aims to provide a relatively more accurate result for two point clouds so that the point clouds have more common regions where points become closer. Depending on whether the object of registration is deformed, the registration methods are also divided into rigid registration and nonrigid registration [21]. On the other hand, according to the types of the theoretical solutions to point cloud registration, point cloud registration can mainly be split into five categories: iterative closest point (ICP)-based methods, feature-based methods, learning-based methods, probabilistic methods, and others [22–25]. The ICP and its variants are classic solutions. They can provide an accurate and reliable transformation between two point clouds through an iterative algorithm of correspondences with a good initial position and rotation. However, the final registration results are very sensitive to the initial position and rotation [26]. Feature-based methods aim to extract the meaningful and robust local geometrical descriptors as correspondences to estimate a transformation between two point clouds. The extracted descriptors can be subdivided into local descriptors [27–29] and global descriptors [30–33]. Local descriptors are extracted from the interest part of the point cloud, and the global descriptors are generated by encoding the geometric information of the whole point cloud. Although the global descriptors have a good performance on the data with obvious geometric attributes, it is difficult to be robust on point cloud containing outlier or flat surfaces.

Learning-based methods provide a relatively more robust transformation between two arbitrarily point clouds by invariant features generated by machine-learning techniques. Thanks to machine learning, feature extraction in

registration can be more invariant because the learned descriptors may contain more detailed features than geometric characteristics or other handcrafted descriptors. The probabilistic method models input point cloud as a density function, such as Gaussian mixture model, and then optimize a statistical discrepancy among probabilistic correspondences to estimate final transformation. One of the advantages of this method is that it can be applied to rigid and nonrigid registration. Other solutions include methods with auxiliary data, in which transformation information is provided from other modules and techniques, and the 4-points congruent sets algorithm based on random sample consensus (RANSAC). The registration algorithms discussed in this work could be used in the pairwise or groupwise/multiview registration and could be applied in a fine or coarse stage with respect to accuracy. Although some of them can be used in nonrigid registration, we will primarily focus on rigid registration in this paper.

Several valuable reviews so far have been published on this topic in the last decade. For instance, Pomerleau et al. [34] presented a general framework based on ICP to classify the existing solutions and algorithms during the last twenty years and mainly focused on the use cases for mobile robotics applications that cover different kinds of platforms, environments, and tasks. It also introduced several realistic situations and applications based on point cloud registration, such as search and rescue tasks, automation of inspection, shoreline monitoring, and autonomous driving. Tam et al. [35] gave a comprehensive summary on rigid and nonrigid registration of point cloud and mesh. Cheng et al. [20] reviewed the existing registration methods based on laser scanning point cloud mainly in photogrammetry and remote sensing. Li et al. [36] gave a good statement of the registration problem and presented a novel framework for globally solving the problem grounded on the Lipschitz global optimization theory with unknown point correspondences. Ben et al. [37] provided an overview of the most important rigid registration algorithm on the estimation of the optimal rigid transformation and discussed their common mathematical foundation. Zhen et al. [38] conducted a comprehensive survey on terrestrial laser scanner (TLS) point cloud registration methods and presented a benchmark dataset containing various scenes. Li et al. [39] made a comprehensive evaluation on the ICP algorithm in terms of overlap ratio, angle, distance, and noise factors. Jiang et al. [40] and Ma et al. [41] provided a comprehensive and systematic review of image registration, and briefly introduced several typical applications in which 2D image registration was discussed in more detail. Specifically, Ma introduced the existing methods of feature-based matching from handcrafted to trainable ones with a comprehensive review. Jiang presented two general frameworks for area and feature-based pipelines and briefly introduced and analyzed typical applications to reveal the significance of multimodal image matching.

Our contributions are to (1) split the point cloud registration problem into the main five categories based on the research publication in the last decades, which is beneficial for new researchers and potential users of registration; (2)

systematically summarize point cloud registration techniques and analyze the similarities and differences of the currently existing methods; and (3) introduce up-to-date achievements, challenges, and further research efforts on the applications based on point cloud registration.

The structure of this paper is organized as follows: Section 2 describes the conception of point cloud registration and the issues in this area. In Section 3, the best-known solutions in ICP-based, feature-based, learning-based, probability-based methods, and others are reviewed and discussed, respectively. Finally, the conclusion and outlook on point cloud registration is given in Section 4.

2. Basic Concepts and Principles of Point Cloud Registration

Point cloud registration is a fundamental technology in computer vision and robotics and is considered as an important prerequisite in various applications that use point cloud data as the input. In this section, we first briefly introduce the source of the point cloud, namely, the point cloud acquisitions. Then, the definition of point cloud data is provided, and it is followed by the introduction of the problems on point cloud registration, and other issues and constraints in this field.

2.1. Point Cloud Acquisitions. Point clouds can be generated by a 3D/depth camera directly or calculated by photogrammetric techniques. Typically these depth cameras can be divided into two categories [42]: passive method based and active method based. A passive method obtains the depth information to generate point cloud data through computing the relationship among multiple view images or a sequential set of images taken at different viewpoints. For instance, structure from motion (SfM) and multiview stereo (MVS) [43] can be integrated to produce point cloud data with high quality in terms of resolution and accuracy [44, 45]. Given a series of images of an object captured at different views (image i acquired at position i), the SfM can utilize these images to create a sparse point cloud based on the correspondence between two images (see Figures 1(a) and 1(b)). For a dense point cloud, the MVS technique is usually considered [44, 46], in which images are taken with the pose as the input provided by SfM, for finally producing a point cloud nearly comparable in accuracy to those generated by laser scanners. The PMVS is one of the well-known methods. It begins by feature detection and matching, and then the expansion and filtering step aims to reconstruct and remove erroneous patches repeatedly. The flowchart of PMVS is illustrated in Figure 1(c). In Figure 1, point clouds generated from SfM and PMVS are shown, respectively. Stereo-depth camera [47] is another type of camera with two or more image sensors spaced a small distance apart so that it is allowed to simulate the human binocular vision to perceive depth information by the determination of correspondence between points in images and the computation of depth values based on point correspondences [48].

The primary advantage of these approaches is that they function well in most outdoor environments. Therefore, there are many outdoor applications of scanning surrounding scene and environment, such as building [49, 50], streets [51], and city [52, 53], using the passive method.

On the other hand, the active devices, such as the time-of-flight (ToF) camera, also called LiDAR, and structured light camera, usually emit or project light and utilize their properties to calculate depth information. Since the speed of light is known, the ToF cameras can calculate the distance between the camera and target objects by measuring the round trip time of a modulated light signal. The speed of generating the final point cloud data can reach 120 fps [54] so that it is widely used in real-time applications, such as autonomous self-driving [55] and gesture recognition [56]. Moreover, depending on the wavelength and power of the emitting light, the ToF camera can provide depth value at significant distances and high-resolution. A structure light camera projects the known patterns, such as phase lines, onto the surface of an object or scene to calculate depth information based on these deformed patterns [57, 58]. As illustrated in Figure 2, the vertical straight stripes are projected onto the object and be distorted because of the surface profile. Eventually, 3D shape information can be obtained by the analysis of these images. Because the structured light camera can obtain high-resolution and speed at a relatively short range [59], it is widely used in fields that require high quality of point cloud in terms of accuracy and resolution, such as 3D shape measurement field [59] and medical field [60].

The basic principle and its applicable scenes of point cloud acquisitions are summarized in Table 1.

2.2. Basics of a Point Cloud. As shown in Figure 3, a point cloud is a collection of points, i.e., $S = \{p_1, p_2, \dots, p_n\}$, and these points describe the shape and surface of an object or scene in 3D. Generally, a point p at least contains the x , y , and z values that define basic geometric coordinates for the target objects, so that $S\{p_1, p_2, \dots, p_n | p \in \mathbb{R}^3\}$. However, when some properties of a single point in the point cloud are available, such as curvature k , normal vector n , and the color information (r, g, b) , the point p could be represented by a longer vector. An example of the point cloud provided by the KITTI dataset [61] is illustrated in Figure 3.

2.3. Registration of Point Cloud. The point cloud registration consists of rigid and nonrigid registration. Rigid registration assumes that point clouds retrieved from the 3D acquisition are related by a rigid transformation that only consists of a translation and rotation. Nonrigid registration allows deformation in input data so that it provides an affine transformation describing translation, rotation, scaling, and shear.

2.3.1. Rigid Registration. Rigid registration assumes an environment in which the transformation can be decomposed as a rotation and a translation so that one point cloud is

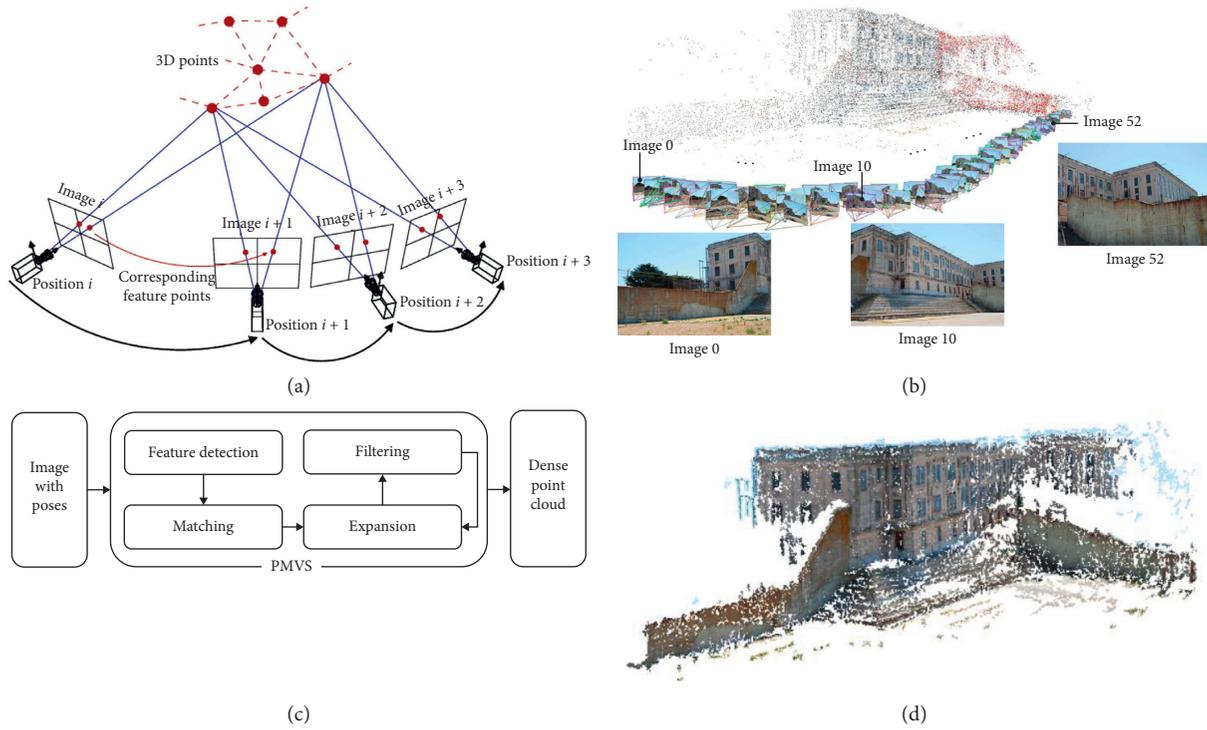


FIGURE 1: Dense reconstruction from SfM to MVS. (a) Structure from motion. (b) An example of point cloud generated by SfM. (c) Flowchart of PMVS algorithm, a patch-based multiview stereo algorithm [44]. (d) An example of dense point cloud generated by PMVS.

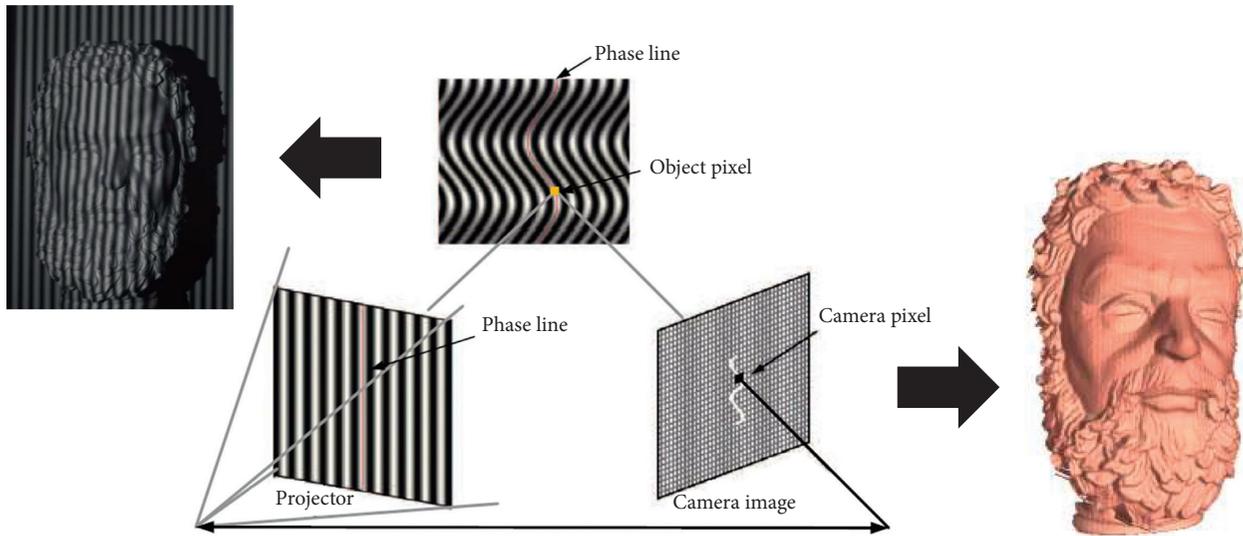


FIGURE 2: A basic principle of the structured light camera.

TABLE 1: A overview of point cloud acquisitions.

Type	Approach	Properties	Application
Passive	SfM-MVS	Take multiple view images as input to generate a sparse point cloud, and then make it dense.	Outdoor/indoor
	Stereo camera	Simulate human binocular vision to determine the position of a point to generate a point cloud.	Outdoor/indoor
Active	ToF camera	Compute a round trip time of a modulated light signal from target object's surface to produce a point cloud.	Outdoor/indoor
	Structured light camera	Use distorted patterns projected to the surface of object to make a point cloud.	Indoor

	x	y	z
p_1	18.323	0.048	0.828
p_1	18.343	0.105	0.828
...
p_n	3.967	-1.473	-1.856

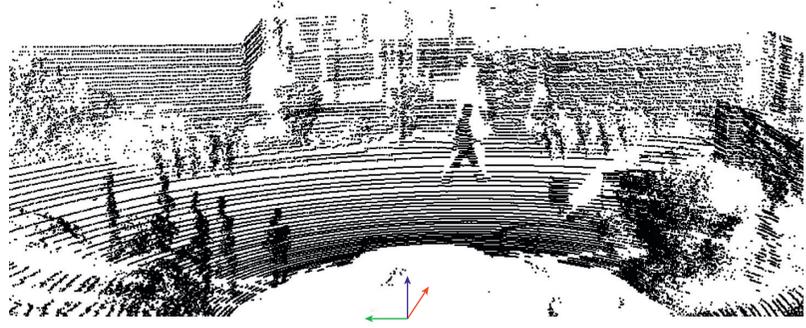


FIGURE 3: A basic principle of the structured light camera.

mapped to the other point cloud while keeping the same shape and size after an appropriate rigid transformation. The rigid transformation is widely used in computer vision since target objects only change their location without altering their shape or size in most applications [62]. Mathematically, we consider two point clouds $P = \{p_1, p_2, \dots, p_n | p \in \mathbb{R}^3\}$ and $Q = \{q_1, q_2, \dots, q_m | q \in \mathbb{R}^3\}$. The task of rigid registration is to find a transformation to align Q with P , and the transformation consists of a rotation matrix R and translation matrix T . The mathematical process can be briefly presented as

$$\begin{aligned} \text{algorithm: } R, T &= r\text{Reg}(P, Q), \\ \text{finalresult: } Q' &= \{q'_i\} = \{R \cdot q_i + T, \quad i = 1, 2, \dots, m\}, \end{aligned} \quad (1)$$

where $r\text{Reg}$ denotes a function that estimates the optimal R and T of a rigid registration based on a certain relationship between two point clouds. This function is also the focus of this paper and will be further detailed in the following sections. After the final step, the final result Q' should be in the same coordinate system as P , and the objects represented by Q and P will be more completed which means the objects have richer information and are more detailed. For example, to compute the transformation between two models formed with points, a transformation is applied for a reading point cloud to align a reference point cloud, as illustrated in Figure 4. If it puts them together without alignment, the point clouds will not be aligned because their coordinates are different (see Figure 4(b)). To obtain a complete scene of the kitchen, multiple point clouds at various viewpoints are retrieved, as shown in Figure 5(a). Point clouds fused into the common coordinate system without registration are shown in Figure 5(b), and point clouds with an appropriate registration solution are shown in Figure 5(c).

2.3.2. Nonrigid Registration. In nonrigid registration, a nonrigid transformation is established to warp the scanned data to the target point cloud. The nonrigid transformation contains reflection, rotation, scaling, and translation instead of only translation and rotation in rigid registration. The use of nonrigid registration is mainly motivated by two reasons: (1) the nonlinearities and calibration errors of data acquisitions can cause low frequency warps on the scans of rigid objects [63, 64]; (2) the registration is performed for the

deforming and moving scene or object that changes its shape over time. For example, handwritten character recognition or a point cloud with warps or deformation is aligned to a model, as illustrated in Figure 6(a). As shown in Figure 6, the person and camera are moving, while the initial and incomplete data are increasingly denoised and detailed in real-time [65, 66].

However, most of the registration solutions to the point cloud so far have focused on rigid registration. The development of nonrigid registration has been relatively tardy compared with rigid registration. To the authors' best knowledge, the estimation of correspondences and optimization techniques is also critical in most nonrigid registration. Once the correspondences are given, the nonrigid transformation will be easier to estimate [67, 68]. Meanwhile, these crucial steps also are essential for rigid registration that we mainly focus on in this paper. Furthermore, in the real world, many applications that need nonrigid registration also involve more or less rigid transformation due to the great complexity of the high dimensionality of nonrigid transformation. Therefore, in this article, we will primarily study the rigid registration and hope it will help those who want to have a comprehensive picture of point cloud registration.

2.4. Issues and Constraints. The technique of point cloud registration has been used in different fields and applications such as 3D object scanning and 3D localization. Meanwhile, a point cloud is often combined with a traditional camera [69] to effectively address real-world problems such as object tracking, object recognition, and object detection. Therefore, the specific algorithms or solutions to solve the problem of point cloud registration are various in different fields because the choice of these algorithms or solutions generally depends on several characteristics, such as accuracy and computational complexity, and even on data used and the environment itself. Point cloud registration is a challenging problem due to the reasons specified as following.

2.4.1. Variant Overlap. Because of the limitations of three-dimensional scanning technology, multiple point clouds usually are captured from different viewpoints, such that each is associated with a different coordinate system, in the sense that overlap region between two point clouds that we

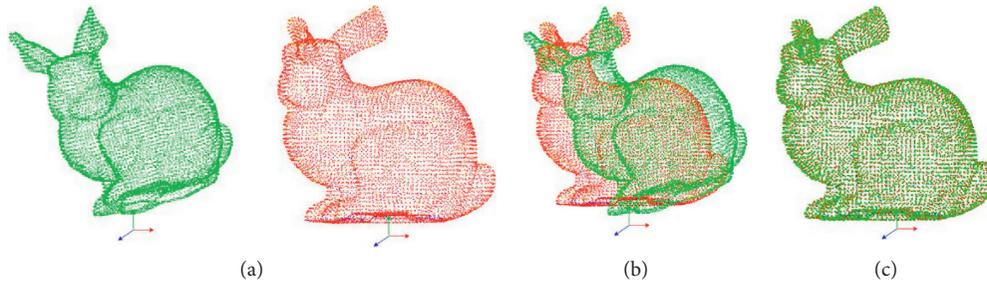


FIGURE 4: An example of rigid registration: (a) two point clouds: a reading point cloud (green) and reference point cloud (red); point clouds fused into the common coordinate system without (b) and with (c) the rigid registration algorithm.

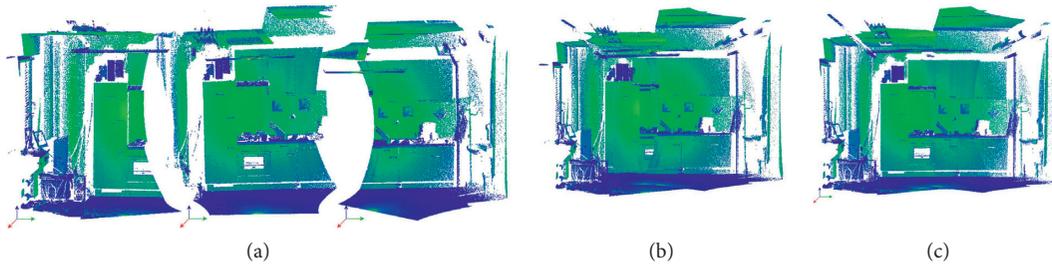


FIGURE 5: An example of rigid registration: (a) point clouds retrieved at different positions; point clouds fused into the common coordinate system without (b) and with (c) the rigid registration algorithm.

retrieved consecutively might be large or small. However, most of the existing solutions often calculate the possible transformation by determining the correspondences based on the common region between point clouds. Therefore, it will be a challenge to process varying overlap during the point cloud registration.

2.4.2. Noise and Outliers. In contrast to triangle meshes represented by a set of triangles that are connected by their common edges or corners, a point cloud acquired from low-cost sensors and time-of-flight cameras [70], such as the Microsoft Kinect camera, is scattered, always suffers from noise contamination, contains outliers, and may have a considerable variation in accuracy [71, 72] due to unavoidable deviation of camera theoretically [73], the lighting or specular reflections [74], and the measurement noise of the acquisition devices [75]. A scanned raw point cloud usually contains sparse outliers and isolated or nonisolated clustered outliers and hence may subsequently result in detrimental effects on practical applications. Generally, a good filtering algorithm should be employed to eliminate the noise from the point cloud while preserving the geometric features of objects or scenes.

2.4.3. High Computational Cost. With the emergence of the high-resolution dataset and cameras, the massive point clouds often bring a great number of troubles to the registration algorithm because the distribution of these point clouds is unstructured and irregular [76]. Moreover, these point clouds obtained by 3D cameras cannot be directly used

in applications such as the iterate-over operation of image pixels. Some procedures, such as removing noise, sampling, and normal estimation, usually first require the computation of the K-nearest neighborhoods (KNNs) for every point in the point cloud and then to visit each of them [77]. To make indexing and searching of point cloud fast, researchers designed many mathematical models such as traditional KNN algorithms and improved algorithms such as building point cloud topology for triangular mesh structure and constructing kd-tree on GPUS [78]. In the research works [77, 79–81], the faster searching algorithm and better storage management of point cloud can support the process of registration. However, with the development of computer power and digital storage and the demand for high-resolution 3D cameras, the effective operation and analysis of massive point clouds is a challenging task for applications.

2.4.4. Various Indicators for the Success of Registration. For point cloud registration, no matter what kind of solutions or algorithms we use, a good result conversion should be defined. Since the most popular classification for this problem is expectation-maximization (EM), the significant step is to find an effective minimization procedure. The well-known solution is the iterative closest point (ICP) algorithm which consists of selection, matching, and minimizing errors [82]. Meanwhile, many researchers in the past decades proposed significant modifications of ICP, such as giving different error metrics (i.e., point-to-point and point-to-surface metric) and point selection strategies, in order to improve the robustness, convergence speed, and accuracy of registration result. Although most variants of ICP achieved a

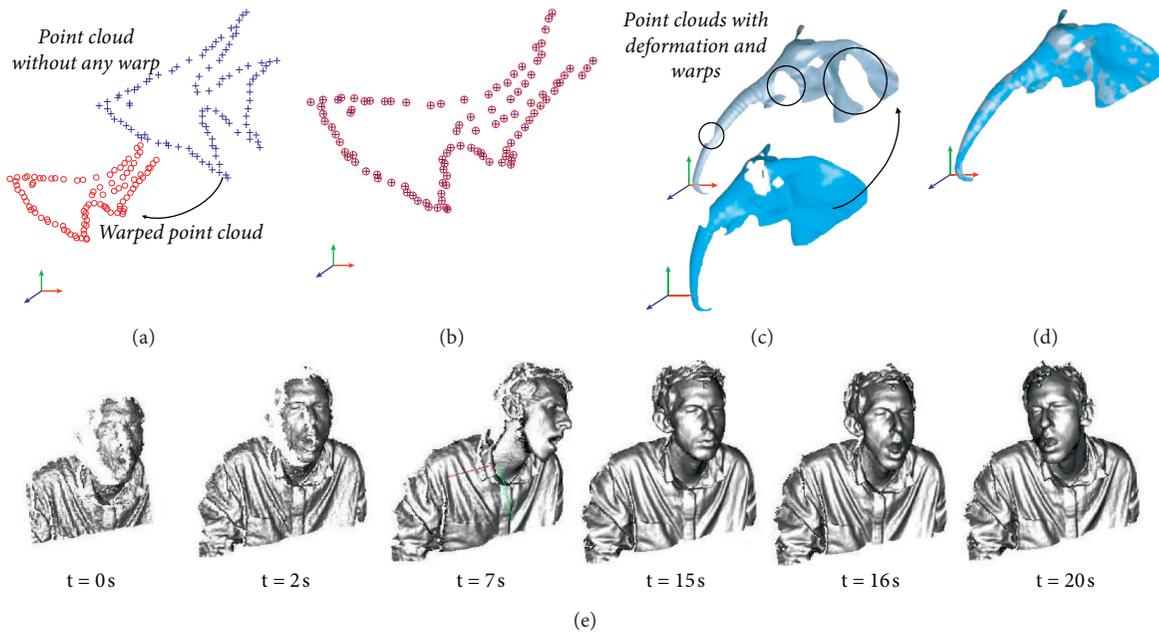


FIGURE 6: Examples of nonrigid registration: in the figures of a fish and elephant, the goal of nonrigid registration is to align point cloud to the target model. (a) Two point clouds with warps. (b) Final registration. (c) Two point clouds with a part of deformation and warps. (d) Final registration. (e) Both the person and camera are moving, and the initial data are detailed and updated in real time (left to right).

good result in the ICP-framework and in their dataset because their methods based on their proposed error metric or termination criteria are better compared with existing methods, what exactly a successful registration result is was rarely discussed. To our best knowledge, Dirk et al. [83] suggested that we could check the percentage of the overlapping surface of the two point clouds and check if the overlapping region has enough inliers to guarantee a successful result. Based on this suggestion, we will discuss the impact of the existing error metric or other criteria in terms of good registration in this paper. Liang et al. [20] gave a list of three error analysis methods: based on the comparison among existing registration algorithms, based on the target point in reference data, and based on a common region between input data. The first method compares existing methods to either improve the current solutions or incorporate the existing approaches into a complete workflow, hence achieving a better result. The second method measures the result of registration by analyzing specific points in reference data. The third one is a widely used method assuming that common points and regions are occurring in two point clouds to be registered.

3. Registration Methods/Algorithms

Over the past decades, growing amount and diversity of methods have been proposed for point cloud registration, from the classic ICP algorithm to the solution integrated with the deep learning techniques. In this section, five categories of the point cloud registration solutions are introduced: ICP-based methods, feature-based methods, learning-based methods, probability methods, and others.

3.1. ICP-Based Methods. Among existing point cloud registration methods, the iterative closest point (ICP) algorithm has received significant attention due to its simplicity and efficiency. The ICP algorithm can obtain a more accurate and robust transformation from one point cloud, named as the reading point cloud, to the other one, called as the reference data. Although it has many drawbacks, such as falling into a local minimum, it will be easily to overcome the drawbacks and to achieve a better result via integrating with other algorithms that can provide a rough transformation as the input to ICP. Therefore, many modified ICP algorithms were proposed and applied in applications. In this section, the ICP algorithm and its excellent variants are reviewed.

3.1.1. ICP Algorithm. The ICP algorithm is an iterative algorithm that can ensure the accuracy, convergence speed, and stability of registration in an ideal condition. In a sense that the ICP can be regarded as the expectation-maximization (EM) problem [84, 85] such that it computes and updates a new transformation based on correspondences, and then is applied to reading data until an error metric converged. However, there is no guarantee that ICP reaches a global optimum. The ICP algorithm can be approximately split into four steps [83, 86]: point selection, point matching, point rejection, and error metric minimizing as shown in Figure 7, and then each of them is introduced in the following sections.

3.1.2. Point Selection. This step aims to select a subset of points representing the current point cloud, because the raw point cloud is usually quite large and requires high

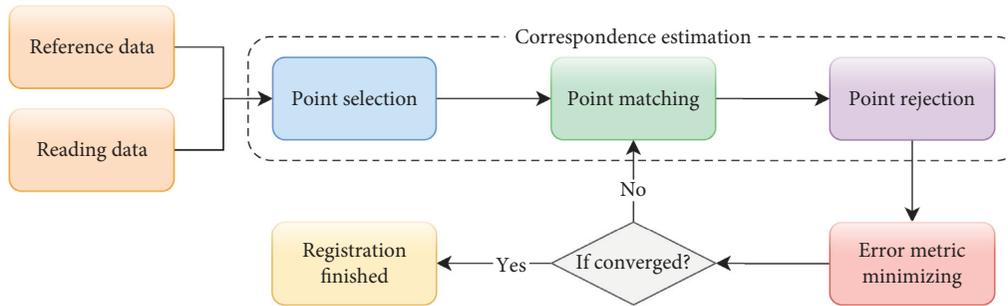


FIGURE 7: The diagram of ICP algorithm: point selection, matching, and rejection can be considered as correspondence estimation.

computational complexity, resulting in unbearable computing time. Meanwhile, depending on the sensors and applications, too much data also bring redundant or unnecessary details into the task of registration. Therefore, point selection as a preprocessing step to reduce the number of points is imperative in registration with high computational efficiency and accuracy.

Therefore, many downsampling strategies were proposed and studied [87, 88] for point selection, such as random, distance limit, and uniform downsampling. In a random strategy, specified points are picked randomly from neighborhoods. When setting a threshold distance in the distance limit strategy, for every point, it picks points from the original cloud so that points in the output cloud keep the specified distance at least between each other. The bigger this distance threshold is, the less number of points is kept, as illustrated in Figure 8. In the uniform strategy, a voxel grid is created to split the point cloud data into a set of small 3D boxes spatially, and then the spatial average of the point distribution in a voxel grid is selected as a real point. An example is shown in Figure 9.

In uniform subsampling, the computational cost of partition and centroids calculation will linearly increase when dense grids are required to achieve high-resolution [89, 90]. Therefore, the distance limit and the random subsampling are widely used as a filter or data reduction step in many applications [90]. On the other hand, these methods can avoid a false offset performance in which generated points are not raw data that might not represent the surface of the object accurately. Moreover, for speeding up the operation in the point cloud, we can use a spatial representation [76], such as kd-tree, as a data structure to efficiently store and manage discrete data of the point cloud.

3.1.3. Point Matching. The purpose of this step is to discover the matching of every point in the reading point cloud onto the reference point cloud so that the correspondences are established between the two point clouds. We can find the correct translation and rotation between the two point clouds after we know the appropriate correspondences.

Correspondences, i.e., point pairs, typically are generated by searching the closest neighbor points in reading data for each reference point with respect to the Euclidean distance. As shown in Figure 10, point clouds are approximately represented by blue points in Figure 10(a),

correspondences are established through finding closest points and marked as blue lines in Figure 10(b). However, the searching operation in matching is a greedy approximation and has a significant computational complexity with $O(N * M)$, where N and M denote the number of points in the respective point cloud. To avoid the exhaustive searching of all the reference points, many scholars improved the performance with the assistance of input data structure. For example, multi-z-buffer [91], kd-tree [92, 93], and octree [94] were developed to store, manage, and operate in order to accelerate the matching process. Specifically, Wei et al. [95] used kd-tree to optimize the calculation of the normal in the point cloud. Meng et al. [96] used a combination of kd-tree and extrapolation to improve the efficiency of registration. Moreover, we can utilize different strategies to find correspondences instead of a closest-based method. For example, Yong et al. [97] introduced a point-to-projection technique to estimate correspondences by projecting reading points onto a destination surface from the point of view of the destination [98]. Pulli [99] studied a point-to-plane technique through the normal vector of the reading point cloud, and so on. These approaches differ from point-to-point, i.e., a closest-based strategy, largely determine the further steps, including point rejection and minimization of error metric.

Additionally, the matching process can be accelerated by hardware. For example, Yong et al. [100] proposed a real-time 3D registration technique in which a point-to-plane based matching process was implemented entirely in CUDA, an up-to-date GPU programming technique developed by NVIDIA for general computing on graphical processing units. Peng et al. [101] introduced a GPU-based ICP method that implemented the GPU-based nearest-neighbor searching in the traditional ICP.

3.1.4. Point Rejection. The purpose of this step is to reject and eliminate invalid correspondences obtained from the matching process, because erroneous pairs of points can negatively affect the estimation of transformation result between point clouds [102]. Additionally, the rejection of invalid correspondences can tremendously increase the robustness against noise and outliers [103].

The widely used methods to reject erroneous point pairs are based on Euclidean distance. One of these criteria is a fixed maximum distance: given a maximal distance as the

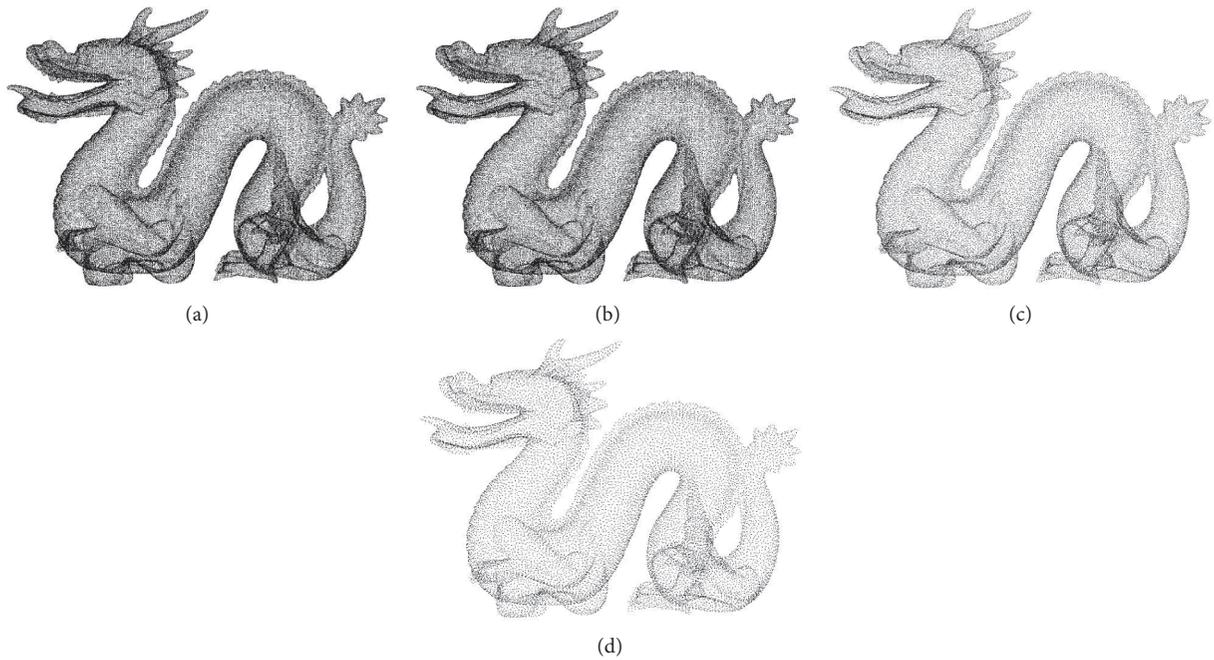


FIGURE 8: Point selection based on distance limit: (a) the original point cloud, and others are the downsampled result, where a distance threshold and a number of selection points are shown, respectively, in parentheses, (b) (0.0005, 91145), (c) (0.001, 38483), and (d) (0.0015, 19773).

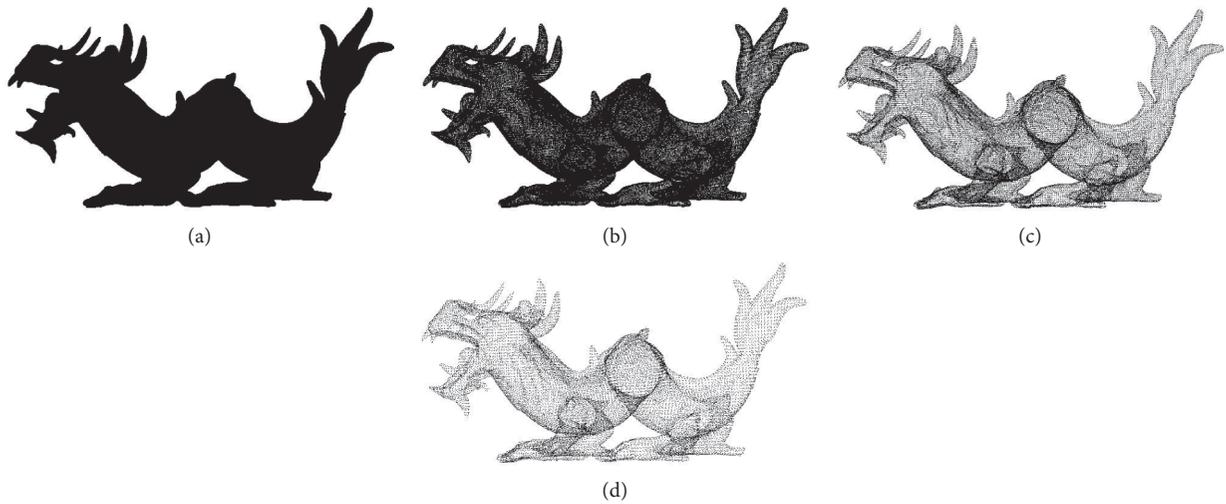


FIGURE 9: Point selection based on uniform strategy: (a) the original point cloud, and others are the downsampled result, where a voxel size and a number of selection points are shown, respectively, in parentheses, (b) (0.5, 218475), (c) (1.0, 56121), and (d) (1.5, 24945).

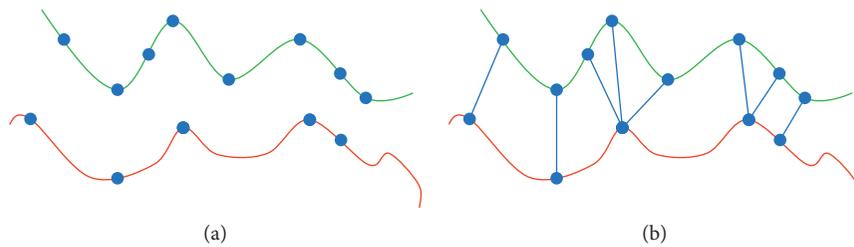


FIGURE 10: A result of point marching: for the sake of clarity, the figure provides examples in a 2D domain. The correspondences represented by a blue line are established between reading point cloud (green) and reference point cloud (red). (a) Original point cloud approximately represented by blue points. (b) Correspondences established by finding closest points.

threshold value, point pairs with the distance greater than this specific value will be rejected [102]. As illustrated in Figure 11(b), the dotted pairs will be removed because their distance is larger than a specific value. It works well for filtering outside overlap regions induced by noises and outliers. Similarly, based on this fixed distance, a specific threshold between a pair of points, correspondences can also be rejected with the worst $n\%$ of pairs [99], or with multiple of the standard deviation of distances [104].

However, rejection based on a fixed percentage or threshold usually is inflexible in dealing with incomplete point clouds [105]. Therefore, other rejection strategies were introduced, such as median distance [106], dynamic threshold [99], RANSAC-based [107], normal compatibility [108], and duplicate matching [109] methods. RANSAC-based rejection can randomly pick up subsets of correspondences to estimate relatively the best transformation, and it is effective to avoid converging into local minima [83]. The duplicate matching focuses on preserving a single pair with removing redundancy pairs between the reading and reference point clouds. As shown in Figure 11(c), the blue lines will be preserved because the pairs connect to the same point with the shortest distance in the reference data. The rejection based on normal compatibility rejects the point pairs that have inconsistent normal information. Specifically, the angle between the normal of pairs is larger than a given threshold, or the normal direction of the point pair is inconsistent [110]. As depicted in Figure 11(d), the arrows show each normal of points, and red pairs ought to be rejected because the angle of their normal is larger than a given value. Furthermore, experiments showed that weighted correspondences, by a weighting function that can be considered as an adjustment of correspondences after point matching, have a highly robust performance for point clouds involving larger rotation and translation [111, 112].

3.1.5. Error Metric Minimizing. This step aims to apply an appropriate cost function and metric that, in a sense, can be used to determine whether a good transformation result has been obtained between pairwise point clouds and then minimize the error metric through iteration with a new transformation. Three approaches [113, 114], point-to-point, point-to-plane, and point-to-projection, are widely used for rigid point cloud registration.

Point-to-Point. Besl et al. [115] and Arun et al. [116] proposed a point-to-point strategy, in which each point in the reading point cloud is paired with the closest point based on Euclidean distances in the reference point cloud to form correspondence pairs, as discussed in the step of point matching. The error metric is the sum of squared differences between the matched pairs as shown in equation (2). Then, for the entire point cloud data, the goal is to find optimal R and T that can achieve a minimum value of equation (3), in which R, T can be assumed a final result. Eggert et al. [117] proved that it could get good performers by evaluating it in terms of numerical stability and accuracy:

$$L(P, Q) = \sum_{i=1}^n \|p_i - q_i\|^2, \quad (2)$$

$$\hat{L}(R, T) = \arg \min_{R, T} \sum_{i=1}^n \|p_i - (Rq_i + T)\|^2, \quad (3)$$

where $\|\cdot\|^2$ denotes Euclidean length, p_i and q_i are the i -th point pair of N pair correspondences from reference point cloud P to the reading point cloud Q , and $R \in SO(3)$ and $T \in \mathcal{R}^3$ represent the rotation and translation part of transformation, respectively. The reading point cloud will be updated with this new transformation and proceed to the next computation, including the point matching and rejection if \hat{L} does not meet the required threshold and converges. The error metric in the ICP algorithm converges and terminates when the difference between the current \hat{L} and the next \hat{L}_{next} is less than a given threshold δ , i.e., $|\hat{L} - \hat{L}_{\text{next}}| < \delta$.

Point-to-Plane. Chen et al. [118] and Bergevin et al. [119] proposed a point-to-plane ICP, searching the intersection on the surface of the reference point cloud from the normal vector of the reading point cloud. This method takes into account searching the nearest point from the reference point cloud and the normal of the searched point. Consequently, it is less sensitive to outliers and noise and has a more accurate performance and faster convergence speed compared with point-to-point based ICP. The error metric of point-to-plane is to minimize the sum of the squared distance from a point to a plane between each correspondence as

$$\hat{L}(R, T) = \arg \min_{R, T} \sum_{i=1}^n \left((Rq_i + T - p_i) \cdot n_{p_i} \right)^2, \quad (4)$$

where q_i is transformed by R and T , and n_{p_i} is the unit normal vector at p_i , and all other variables are the same as in equation (3). For obtaining an optimal R and T , the linear least-squares problem can be formatted and solved with a linear optimization, such as singular value decomposition (SVD) method [116], computing orthonormal matrices [120], and unit quaternions [121], or a nonlinear solver, such as Levenberg–Marquardt method [122]. The performance of these methods is similar in terms of accuracy and stability to minimize the error metric function [117].

Point-to-Projection. The point-to-projection approach was proposed by Blais et al. [98] and Neugebauer et al. [123], and it is fast but less accurate than priors. It obtains correspondences through projecting the points from the reading point cloud onto the adjacent points in the reference point cloud along point normals (i.e., normal shooting approach), in a sense that it uses a different strategy to establish correspondences. Followed by rejection to achieve valid correspondences, the error metric in point-to-projection is the same as equation (3).

In addition, for achieving an ICP algorithm with more precise and fewer iterations in different applications, the other error metrics were proposed, such as point-to-line

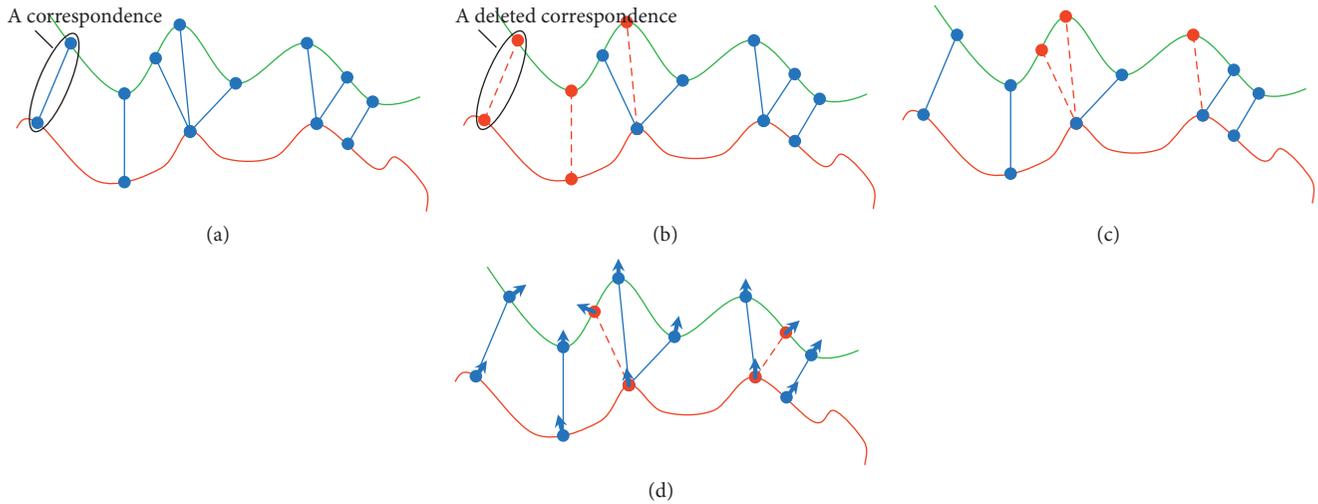


FIGURE 11: Methods to reject invalid correspondences: (a) original point pairs that are obtained in the matching process; (b) a case of the fixed maximum distance strategy; (c) a case of rejection based on duplication matching; (d) an example of rejection based on the angle of normal.

based ICP [124], and colors or curvatures based method [125] that considers the compatibility of more properties of points. Among these error metrics, the widely used are point-to-point and point-to-plane. The point-to-plane approach has been shown to converge faster than the point-to-point algorithm [82], especially for those applications that need to scan large planar surfaces, while the point-to-point has better robustness and convergence precision [39]. Moreover, there are many variants based on point-to-plane and point-to-point for achieving good performance. For example, Rusinkiewicz et al. [82] introduced simple and computationally efficient methods based on point-to-plane. Forstner et al. [126] presented a computationally efficient solution to determine the relative transformation in real-time. Segal et al. [127] introduced generalized-ICP (GICP) that attaches a probability model to the minimization of error metric in order to reduce iterative computation.

3.1.6. ICP Variants and Summary. Over more than two decades, many ICP variants were proposed with respect to two key steps: searching valid correspondences and conducting efficient minimization of error metric. They have grown into sophisticated tools that can be applied to a custom application. In this part, we introduce several state-of-art ICP variants and practices for real-world applications.

First of all, the mainly used ICP algorithms are point-to-point based and point-to-plane based. Therefore, many specific algorithms have been implemented by researchers, these algorithms are often used to compare with other approaches, for instance, Open3D [128], an open-source library that processes 3D data, Point Cloud Library (PCL) [129], an open-source project contains a part of large-scale point cloud processing, and Libpointmatcher [130], a highly configurable chain of ICP and filters, and all of them are cross-platform; that is, they can run on the mainstream platforms, such as Windows and Linux.

Secondly, some modifications of the original ICP can also be used to achieve attractive performance in registration tasks. For example, Go-ICP introduced by Yang et al. [131] searches the entire transformation space to reach a globally optimal estimation by a branch-and-bound (BnB) scheme, avoiding falling into local minima. Generalized-ICP (GICP) studied by Segal et al. [127] handles the minimization of error metric in a probabilistic model. It enhances registration to be more robust with incorrect correspondences. Although these methods still need to integrate with the traditional techniques of ICP for higher accuracy, it can provide a good result of registration.

The processes presented in this section are summarized in Table 2, in which the modules and implement details of the ICP algorithm are listed.

3.2. Feature-Based Methods. As we can see in the ICP-based algorithm, the establishment of correspondences is critical before the transformation estimation. The final result can be guaranteed if we obtain appropriate correspondences describing the correct relationship between two point clouds. Therefore, we can stick landmarks on the scanned object or manually pick equivalent point pairs in postprocessing to calculate the transformation of interesting points (picked points), and this transformation can finally be applied to the reading point cloud. As illustrated in Figure 12(c), point clouds are loaded in the same coordinate system and painted in different colors. Figures 12(a) and 12(b) show two point clouds captured at different viewpoints, point pairs are selected from reference and reading data, respectively, and the registration result is shown in Figure 12(d).

However, these methods are neither friendly to the measured object that cannot be attached with landmarks, nor can it be applied in the applications that require automatic registration [132]. Meanwhile, to minimize the search space for correspondences [26] and avoid the assumption of initial

TABLE 2: List of processing blocks that is available in the modern ICP algorithm.

Step of ICP algorithm	Modules	Implementation details
Point selection	Random subsampling Distance limitation subsampling	
Point matching	Uniform subsampling Based on closest Euclidean distance	Accelerated by kd-tree, octree, multi-z-buffer
Point rejection	Fixed maximum distance Duplication matching Normal angle Weighted rejection	
Error metric minimizing	Point-to-point Point-to-plane Point-to-projection Point-to-line	Solved by singular value decomposition, computing orthonormal matrices, unit quaternions, Levenberg–Matquardt, branch-and-bound

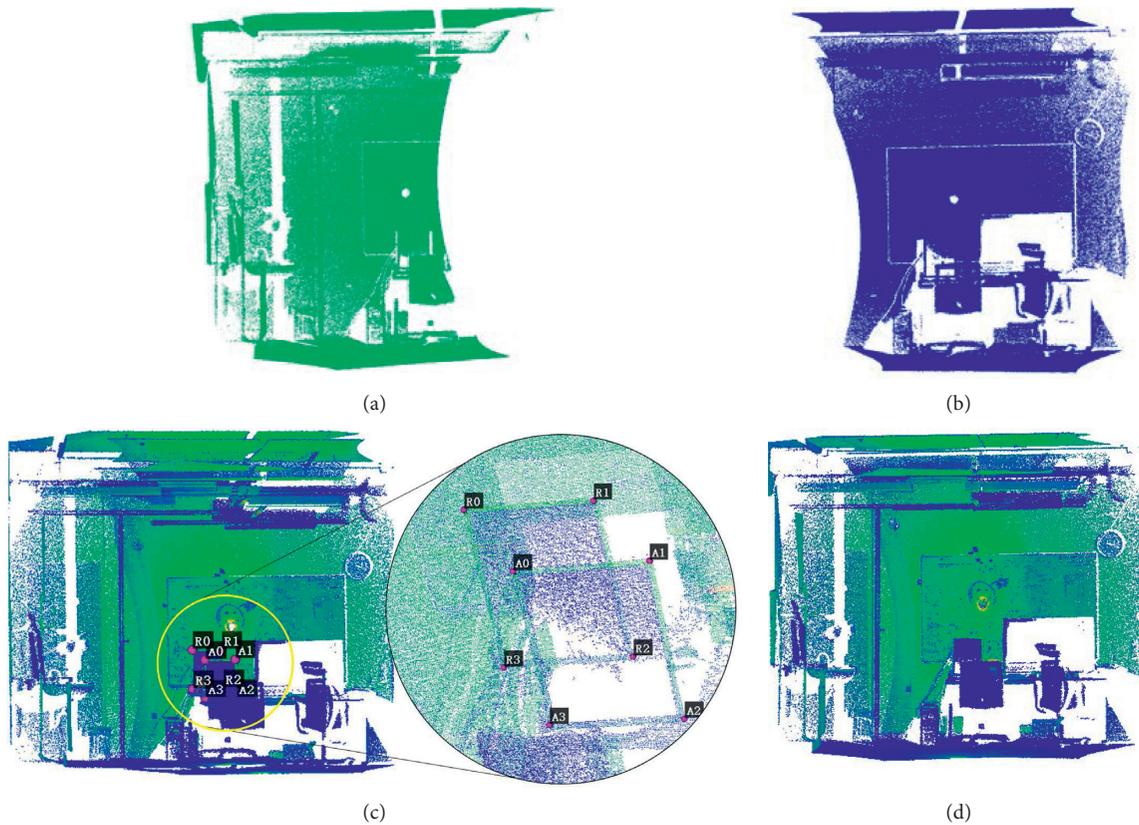


FIGURE 12: Manually picking four equivalent point pairs in postprocessing. (a) The reference data. (b) The reading data that are used to register to reference data. (c) R_* and A_* points are manually picked, and $R0; 1; 2; 3$ corresponds to $A0; 1; 2; 3$, respectively. (d) Result of registration using these four pairs.

transformation in the ICP-based algorithm, feature-based registration was introduced in which key points designed by researchers are extracted. In general, the keypoint detection and correspondence establishment are primary steps in this method [133].

Depending on the applications, these features are generated regarding geometric information of the point cloud. For instance, features generally can be highly correlated with shape information, such as, curvature, normal [134], colors [135], and intensity [136]. In [26], curvature and surface

normal was utilized to express the characteristics of the point cloud, and the correspondences were found with these conditions:

$$\begin{cases} \|h(p_i^s) - h(p_j^s)\|_2 \leq \tau_{\text{curvature}}, \\ \|w(p_i^s) - w(p_j^s)\|_2 \leq \tau_{\text{normal}}, \end{cases} \quad (5)$$

where p_i^s and p_j^s are features in the reading data and reference data, respectively, h and w are the surface curvature and normal angle of the neighbor point, and $\tau_{\text{curvature}}, \tau_{\text{normal}}$ are the threshold for curvature and normal angle. For some applications that primarily collect road or driving point cloud data, line and surface information are important because road networks and roof patches might be in common [137].

These approaches were unexpected and suggested that descriptors should be descriptive and invariant, such that it is capable of encapsulating the predominant information from the point cloud, and better insensitive to noises, i.e., disturbed points, so that the correspondences can be effectively established between pairwise point cloud, to achieve more robust rigid transformation finally. These handcrafted features, referring to some properties of features that are manually engineered by the researchers, have a long history of designing and studying. Many 3D descriptors for point clouds have been introduced and applied to the task of point cloud registration. Meanwhile, comprehensive reviews for point cloud descriptors and comparison in terms of computational efficiency and accuracy were provided by [138–142] as well. According to the way of descriptor extraction, the descriptors can be approximately classified as local, global, and hybrid descriptors. Local descriptors, for every point, are calculated through spatial distribution or geometric attributes from their neighbors, and it is produced by a histogram representing the descriptors for each point. It is less discriminating exactly, but it is a good choice for cluttered scenes. On the other hand, a global descriptor, for the entire point cloud, is produced with a single descriptor vector by integrating a set of features. Global-based features are often used in 3D object recognition and 3D object categorization, while some global descriptors can be used for pose estimation of the objects because they contain local descriptors, such as Viewpoint Feature Histogram (VFH) and Local-to-Global Signature (LGS) descriptor. Moreover, global-based methods are less computation compared with the local features and better to describe the whole point cloud but are affected generally by occlusions and clutter [143]. Hybrid-based descriptors incorporate local and global descriptors together with most of the advantages from them.

In this section, two local-based methods: Point Feature Histogram (PFH) [28, 144] and Signature of Histograms of Orientations (SHOT) [145] are introduced, respectively, for point cloud registration.

3.2.1. Point Feature Histogram (PFH). PFH can be classified as local feature descriptors that rely on the local neighborhood of each point, while the global feature descriptors estimate geometrical information as a vector through the

whole point cloud. In the keypoint detection, PFH exploits the geometrical properties of the point's neighborhood to estimate multivalue features. Then, the feature descriptions are created by binning these features into a histogram. Followed by the establishment of correspondences in multiple-space by ks-tree, the transformation (translation and rotation) between matched pairs is constructed and applied to the reading point cloud as an initial parameter for ICP.

Specifically, in the k -neighborhood around point p_s , PFH descriptor is defined by four features as

$$\begin{cases} f_1 = \langle v, n_t \rangle, \\ f_2 = \|p_t - p_s\|_2, \\ f_3 = \frac{\langle u, p_t - p_s \rangle}{f_2}, \\ f_4 = a \tan(\langle w, n_t \rangle, \langle u, n_t \rangle), \end{cases} \quad (6)$$

where u , v , and w represent the Darboux frame coordinate system, as shown in Figure 13, and f_1 , f_3 , and f_4 are in practice the difference between p_s and p_t (distance is excluded from histogram). Subsequently, PFH is formed by binning these three features into a div^3 histogram, where div is the number of subdivisions in the range of feature values. In the implementation of Point Cloud Library (PCL), div is set to 5 such that each point with 125 bins, followed by establishing correspondences through these histograms that are represented by a high-dimensional space. The correspondences between two point clouds are depicted in Figure 14(a), the specific histograms at p_{290} and p_{6916} are shown, and the final registration result by PFH is shown in Figure 14(b).

The computation complexity of PFH is $O(n^2)$ where n is the number of points. Fast PFH (FPFH) that has a computational complexity of $O(n)$ was proposed in [27], in which a Simplified Point Feature Histogram (SPFH) is generated and weighted to construct the FPFH.

3.2.2. Signature of Histograms of Orientations (SHOT).

SHOT is a local-based descriptor that combines the benefits of both signature- and histogram-based methods. Signature-based methods describe the 3D points with their neighborhood by encoding geometric information in a defined invariant local reference frame. Histogram-based methods describe the neighborhood by compressing this information into bins following a specific quantized domain. Specifically, it builds a robust local reference frame for every point in the point cloud, where angle θ_q is calculated individually between normal n_q and local z axis, and then 3D descriptors are encoded by a histogram within around points in the support. Finally, for the signature structure, an isotropic spherical grid divided into 32 volumes in the experiment is used to encompass partitions along the radial, azimuth, and elevation axes. Figure 15 provides a graphic description of

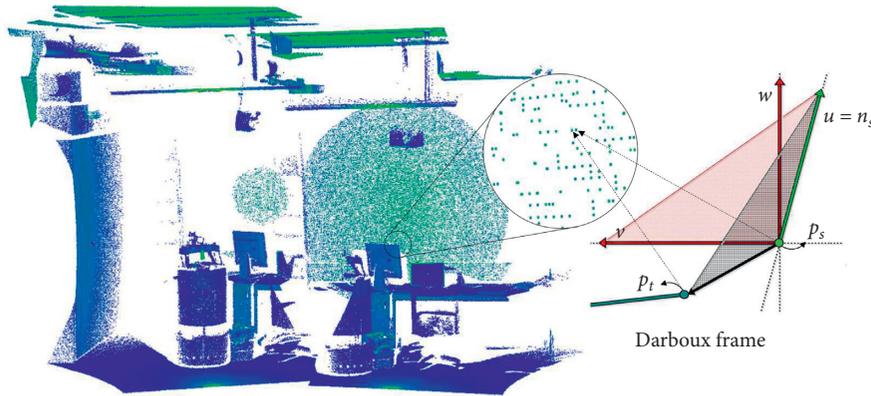


FIGURE 13: The Barbour frame created by vector u , v , and w places at points. p_s .

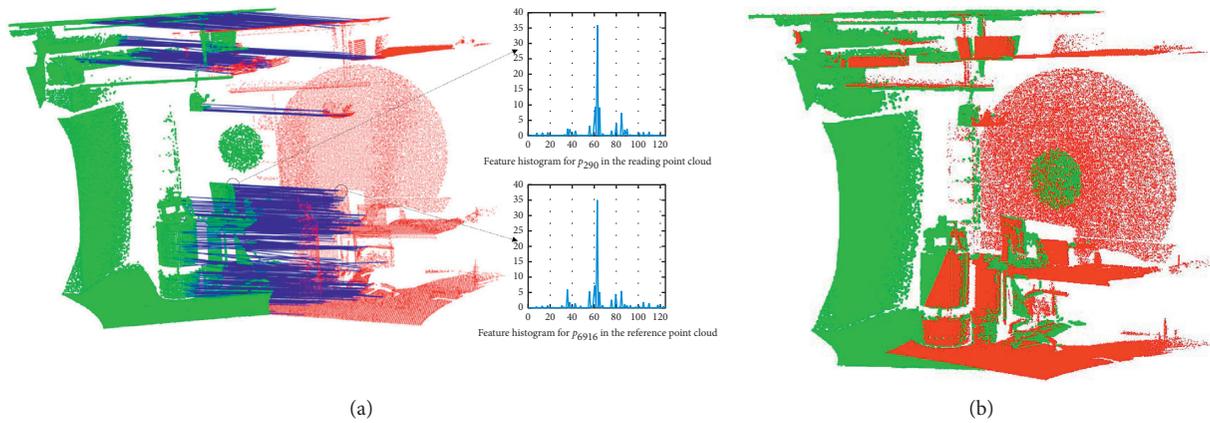


FIGURE 14: An example of registration using PFH features: two point clouds are colored by green and red, respectively. (a) Correspondences represented by blue lines between point clouds. (b) The final result applied with a transformation through PFH features.

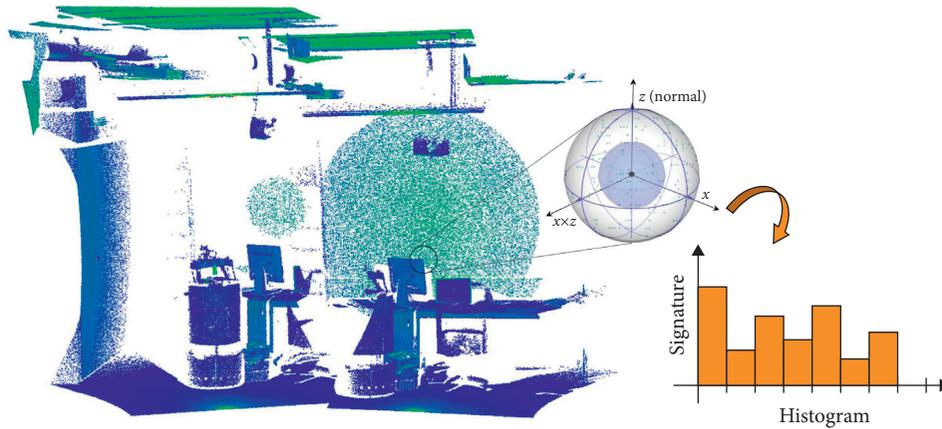


FIGURE 15: A graphic description for SHOT.

the signature structure for SHOT, where for better clarity, volumes of the grid are 16 because of 2 elevation divisions, 2 radial divisions, and 4 azimuth divisions instead of 8.

Apart from the computation of descriptors, the next step for point cloud registration is to establish correspondences between features extracted from reading and reference data.

A threshold is set for finding the nearest neighbor to match point-to-point correspondence. As shown in Figure 16(a), p_{313} and p_{5210} are a pair with under a specific distance in feature space. In the registration experiment, a coarse registration is obtained by matching SHOT descriptors, in which a transformation is calculated based on the area of

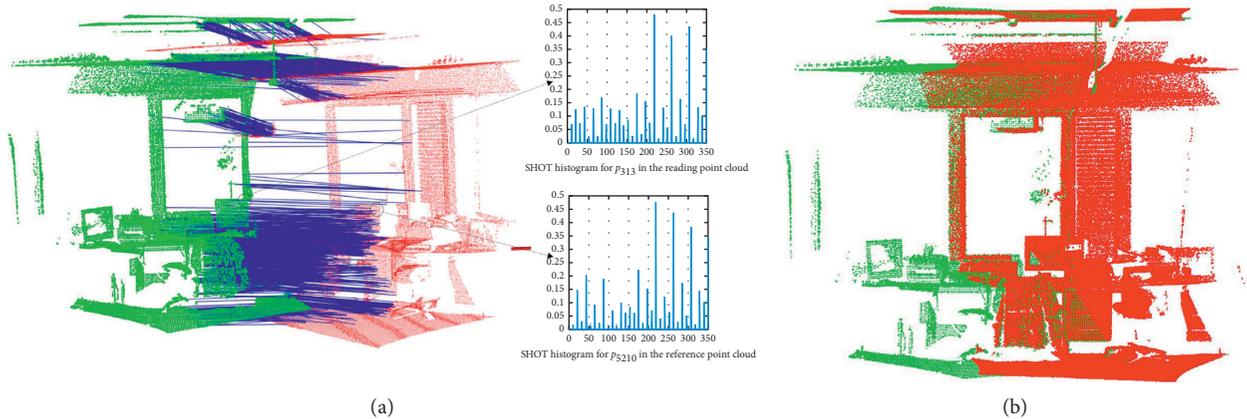


FIGURE 16: An example of registration using SHOT features. (a) Correspondences matched by blue lines between point clouds. (b) The final result estimated by SHOT features and ICP algorithm.

overlap where correspondences can be extracted. Then, the transformation leads into an initial guess into the ICP algorithm to achieve a fully automatic registration point cloud, as illustrated in Figure 16(b).

3.2.3. Further Steps after Establishment of Correspondences. This section introduces the further steps after obtaining a large set of correspondences. To our knowledge, given a large set of correspondences, a popular strategy was proposed [146, 147] and consisted of two steps: (1) building a set of putative correspondences containing outliers and inliers; (2) removing the outliers and estimating the transformation using inliers.

For some point cloud registration tasks, the point-to-point correspondences are always established through associating each descriptor to its nearest neighbor [141] in the reference point cloud in the descriptor space. The transformation produced by these correspondences can be directly considered as a rough result and then improved by other algorithms such as ICP. However, this strategy might not be accurate enough, because the correspondences obtained from two point clouds include inliers and outliers generally, and the ICP are sensitive to noises and outliers. Additionally, the real-world data contain noises, outliers, occlusion, and even flat surfaces with few apparent features, such that the registration performance is severely degraded because the extracting of traits from the point cloud is affected.

Therefore, designing an algorithm to remove outliers and estimate transformation based on inliers efficiently is important. A key of this task is to remove mismatches from given putative feature correspondences, leaving the true matches to calculate transformation. To this end, a simple way is to reject random correspondences based on the hypothesize-and-verify framework [148]. However, the notable drawback of it is it requires a large amount of trials, especially when the inlier ratio is low and the expected confidence to find correct matches is high [149]. To eliminate the mismatches fast, a mismatch removal method was proposed [150] that preserves local neighborhood structures of feature points from thousands of correspondences in few

milliseconds, resulting in a quick initialization result for registration. Ma et al. [147] considered constructing a putative set of correspondences and rejecting matches due to sufficiently different feature descriptor vectors. In addition, for nonrigid and rigid registration, a solution that robustly estimated the transformation from correspondences was proposed [151, 152] using L_2 -minimizing estimate [153], but it is severely biased by outliers. A set of putative correspondences could be purified by semisupervised learning [154] or a mixture of densities [155]. However, these feature-based methods compute transformations solely based on a set of putative correspondences instead of the entire input data.

3.3. Learning-Based Methods. In recent years, learning-based approaches have received significant attention in many fields related to computer vision, especially object detection, classification, semantic segmentation, etc. In applications that use a point cloud as an input, traditional strategies to estimate feature descriptors heavily rely on distinctive geometric properties of the object in the point cloud, as we discussed in the last section. However, the real-world data often vary from object to object that may contain flat surfaces, outliers, and noise. Moreover, the removed mismatches usually contain useful information and can be used for transformation learning.

Learning-based techniques can be suitable to encode semantic information and be generalized in specific tasks. Specifically, most registration strategies integrated with machine-learning techniques are faster and more robust than classical methods and flexibly extended to other tasks, such as object pose estimation and object classification. Similarly, a key challenge in point cloud registration based on learning is how to extract features that are invariant to the point cloud's spatial change and more robust to noise and outliers. In this section, we will describe and discuss several state-of-art approaches integrated with machine learning.

3.3.1. Major Methods. PointNet introduced in [156] is a deep network architecture and is fed by an unstructured point cloud (n points) directly. It outputs either a global

feature represented by a single vector or the local features for each point through combining the global feature with each of the point features, for the tasks of object classification and semantic segmentation separately. PointNet is composed of three parts: (1) symmetry function for unordered input; (2) local and global information aggregation; and (3) joint alignment network, as shown in Figure 17. It should be noticed that the alignment in the network is used to align both input points and point features, and this alignment can make sure that the learned representation is invariant to the geometric transformation of the point cloud. Although PointNet for 3D descriptors [157] mainly focuses on 3D classification and segmentation, they indeed provide research studies with more inspirations on the registration task because the descriptors used in segmentation task and 3D object classification bring a different perspective for correspondence establishment.

As the pioneering effort that directly processes raw point cloud, PointNet++ [158] developed from the PointNet can learn local features by exploiting metric space distance. PointNet++ is a hierarchical neural network architecture that consists of a set of learning layers to combine features from multiple scales, whereas the traditional PointNet is to learn from spatial information of points and then aggregate these features into a global signature of the point cloud. This hierarchical structure includes a bundle of set abstraction levels which are composed of a sampling layer, a grouping layer, and a PointNet layer. The sampling layer focuses on selecting a set of points from input data. The grouping layer focuses on constructing local region sets for each point. The PointNet layer aims to produce feature vectors by encoding the local region pattern. A hierarchical architecture of PointNet++ is given in Figure 18, N denotes the points with d dimension coordinate and C dimension feature, and K is the number of centroid points' neighborhood.

Point Cloud Registration Network (PCRNet) [159]. It presents a framework combines PointNet features to find the transformation between point clouds. In particular, it uses five multilayer perceptrons (MLPs) which are similar to the PointNet architecture and are arranged in a Siamese architecture, to encode the point cloud as global feature vectors that contain the information on the geometry and orientation of point clouds. Followed by estimating the transformation with a single forward pass in the fully connected (FC) layers, which is different from PointNetLK [160] that uses a traditional algorithm to calculate the transformation. An architecture of PCRNet is illustrated in Figure 19. Point clouds first are fed as input into five MLPs (64, 64, 64, 128, 1024), and the output is global features generated from symmetric max-pooling function ϕ . These features are concatenated and provided as input to fully connected layers, and the final result that includes translation and rotation values will be calculated in an FC with (1024, 1024, 512, 512, 256, 7). Experiments showed that iterative PCRNet is similar to Go-ICP with respect to accuracy but much faster based on ModelNet40 [161].

Deep Global Registration (DGR) [24]. It consists of three modules: a 6D convolutional network for estimating the veracity of each correspondence, a weighted Procrustes

solver for a closed-form rigid registration, and an optimization module that fine-tunes the registration generated from the previous step. Specifically, it first extracts features from the pairwise point cloud using fully convolutional geometric features (FCGFs) [162], and then a translation invariant convolutional network is used to identify the correct correspondences. Finally, the weighted mean squared error is minimized by the Procrustes method [163] to generate rotation R and translation T , as shown in Figure 20. To improve the accuracy of registration, a fine-tuning module is proposed finally to minimize a robust loss function, in which the optimization space is processed smoothly by removing discontinuous. Experiments showed that it can achieve a good performance on real-world data, 3DMatch [164].

Deep Closest Point (DCP) [165]. To overcome the intrinsic drawbacks of ICP, i.e., converging to spurious local optima, DCP was introduced by using learned embedding to guarantee a better marching. It has three parts: a point cloud embedding network, an attention-based module to predict a soft-matching that is generated from a probabilistic approach, and an SVD layer to estimate rigid transformation. In more detail, the raw point cloud is embedded into high-dimensional space by PointNet or dynamic graph convolutional networks (DGCNNs) [166], followed by an attention and pointer generation module to learn contextual information and establishment of soft correspondences. Finally, the transformation between point clouds is calculated in a differentiable SVD layer. As shown in Figure 21, \mathcal{F}_x and \mathcal{F}_y are feature embeddings, respectively, for \mathcal{X} and \mathcal{Y} , and Φ_x and Φ_y are new embeddings that have contextual information. $m(x_i, \mathcal{Y})$ is a soft pointer from each x_i into the points of \mathcal{Y} . Finally, we can get pairs containing x_i to y_i over all i . In particular, this is a DCP-v2 integrated with an attention module while the DCP-v1 does not. Additionally, because PointNet and DGCNN both can extract features from the point cloud, but PointNet learns global descriptors while DGCNN learns local descriptors. In this article [165], the proposed method based on DGCNN empirically performs better than the PointNet in terms of accuracy on the dataset ModelNet40 [161]. However, this method is not accurate enough. Thus, it can be further polished by the ICP algorithm after a better initialization provided by DCP to converge to the global optimum.

Partial Registration Network (PRNet) [23]. It tackles partial correspondence problems by learning geometric priors with self-supervised learning. PRNet is composed of an appropriate geometric representation and a keypoint detector that focuses on finding common points and establishing correspondences. Moreover, the PRNet is designed to be iterative like ICP, such that it is capable of performing coarse-to-fine refinement. Specifically, PRNet first detects keypoint of point cloud X and Y and then predicts a mapping and rigid transformation (m_p) from X to Y based on the keypoints, as shown in Figure 22. Finally, the obtained transformation is applied to X , and the process is executed repeatedly with the new transformation as input. However, to be sharper and approximately differentiable than traditional methods [167], PRNet can be considered as

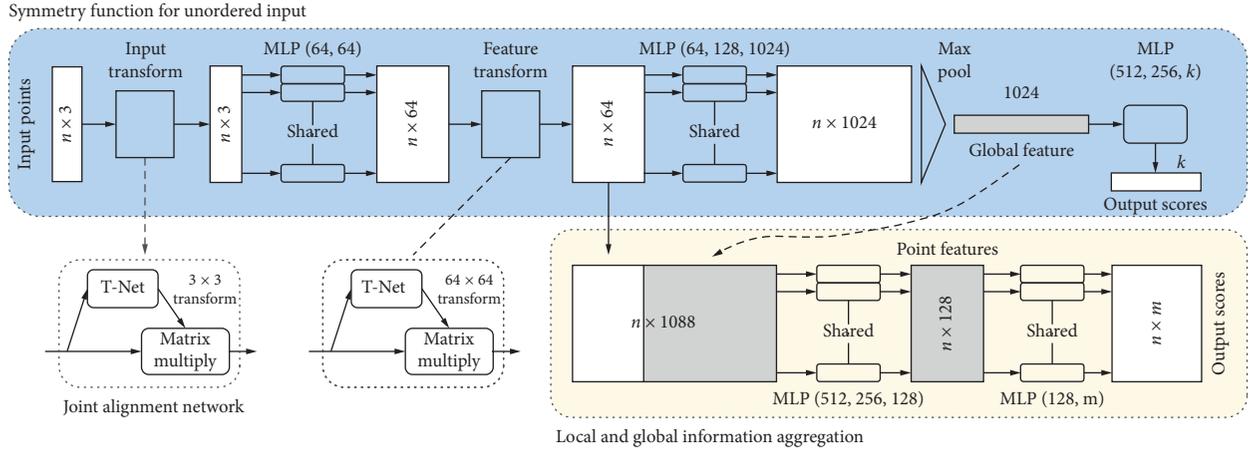


FIGURE 17: Architecture of PointNet, in symmetry function for unordered input, the output for classification is score for k classes.

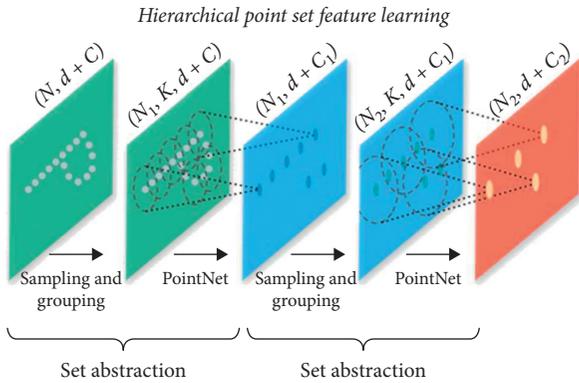


FIGURE 18: Hierarchical structure of PointNet++ in which an $N \times (d + C)$ is the input of a sampling layer.

an actor-critic method (ACM), using Gumbel-Softmax [168] to sample a matching matrix for predicting a mapping of keypoints from X to Y . Experiments showed that PRNet is more robust to noise on the ModelNet40 [161].

Robust Point Matching (RPM-Net) [169]. It builds upon the previous iterative framework RPM [170] that uses optimization technique and soft assignment strategies to solve point matching problem. RPM-Net has mainly three parts: (1) computing the hybrid features combined with spatial coordinate and geometric properties of points; (2) a differentiable Sinkhorn [171] layer and annealing algorithm [172] is applied to get soft assignments of correspondences from these features; (3) the weighted SVD is used to solve the rigid transformation. In particular, features for every point from the point cloud are extracted by PointNet and FPFH [27], followed by an annealing algorithm for soft assignment and Sinkhorn normalization that produces the final correspondences M^i , during which a secondary network is optional used to predict and adjust the annealing parameters (α, β) during iteration. Finally, the rigid transformation is computed by SVD while each correspondence \tilde{Y} is weighted. The overview of RPM-Net is shown in Figure 23. The result data obtained in experiments suggest that it is robust to noise data on the dataset ModelNet40 [161] based on mean isotropic rotation and translation errors.

PointNetLK [173]. It builds a deep neural network framework derived from PointNet and is integrated with the Lucas–Kanade (LK) [174] algorithm which is classic techniques for image alignment. To this end, PointNetLK begins by producing K -dimensional vector descriptor from PointNet function ϕ , and then ΔG as current transform is represented by an exponential map and estimated in the LK layer, estimating the alignment without costly computation of point correspondences. The final transformation G_{est} is obtained by all incremental estimates computed during the iterative loop. This model is shown in Figure 24. PointNetLK evaluated on the dataset ModelNet40 [161] demonstrates more robust to noise data and is insensitive to initial position than the ICP algorithm.

3DRegNet [175]. Given a set of noisy point correspondences, it focuses on classifying inliers and outliers because inliers mainly contribute to the estimation of an optimal pose. To end this, a deep neural network (DNN) for registration is designed to compute the transformation parameters $(R$ and $T)$. Specifically, 3DRegNet starts by passing a set of point correspondences through a fully connected layer, during which 128 dimensional features for every correspondences are calculated and will be fed into ResNets [176] to obtain weighted $w_i \in [0, 1)$ correspondences (see Figure 25(b)). Subsequently, DNN is presented to extract and process meaningful features by pooling and context normalization. Then, the transformation is estimated at the end registration block (see Figure 25(c)). The architecture of 3DRegNet is shown in Figure 25(a). Moreover, an additional 3DRegNet can be used to compute a smaller transformation for a more precise result. A series of experiments based on the dataset ICL-NUIM [177] and SUN3D [178] suggest that a deep learning approach indeed performs better under the condition of lower quality of point correspondences.

3.3.2. Comparison and Summary. In this section, the learning-based registration methods are summarized and compared in Table 3. For a better understanding, approaches mentioned in Table 3 are arranged in chronological, such that PointNetLK was introduced in 2019, whereas DGR was

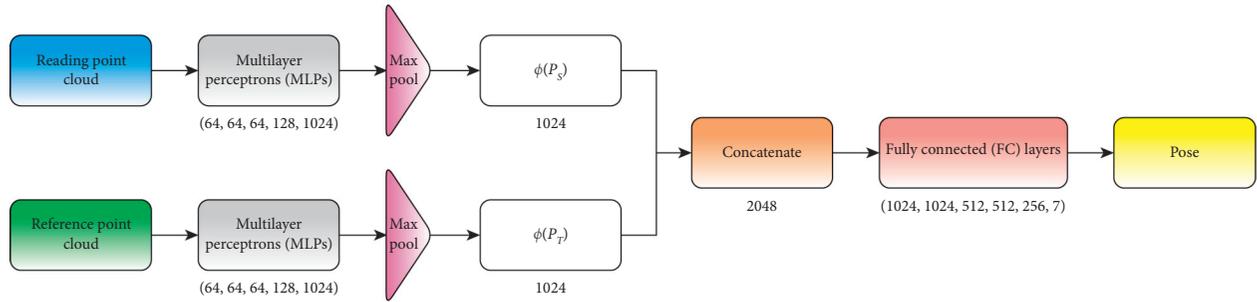


FIGURE 19: An architecture of PCRNet.

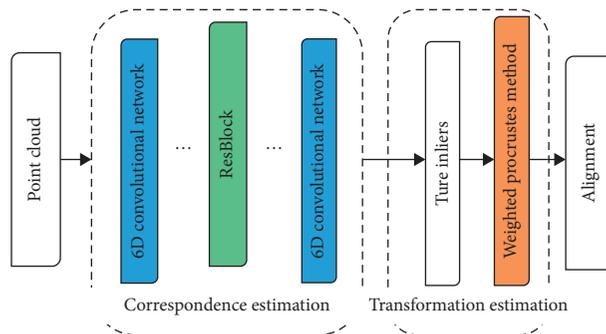


FIGURE 20: An architecture for DGR.

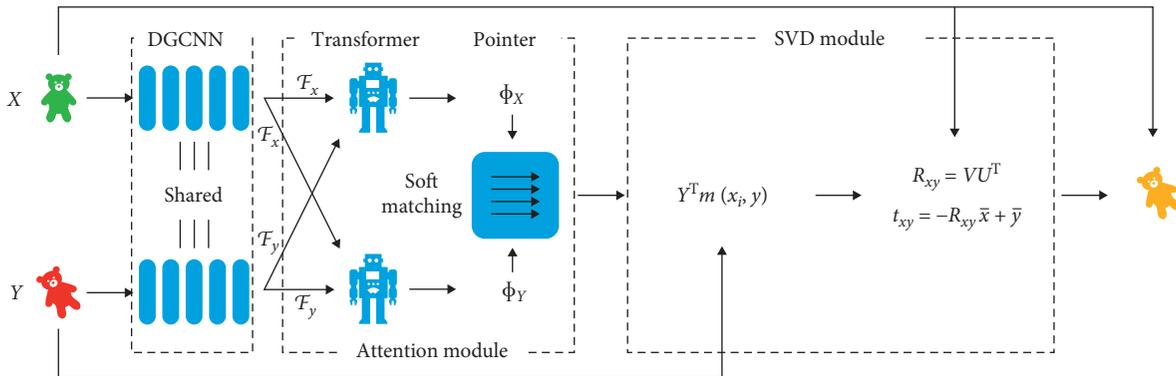


FIGURE 21: Network architecture of DCP.

proposed in 2020. Thanks to the existence of various datasets, many learning-based methods can be put together and compared with each other in terms of the same indicators, such as time consumption, accuracy, and robustness to noise. However, in Table 3, the performance based on evaluation metric is our main concern because time consumption is highly dependent on the computational devices which are very diverse. Furthermore, the robustness to noise is also difficult to measure uniformly, since different noise generation distributions may affect the performance of algorithm unless the algorithm is tested for several times with different noise distributions, as in the case of Peng Li [39] applied various standard deviations of Gaussian noise to point cloud.

Moreover, Network type refers to the predominant methods used to solve correspondence estimation and transformation estimation problem, though there are

secondary networks applying for prediction of specific parameters, such as RPM-Net optional uses a secondary network to adjust parameters during iteration. Need refinement indicates whether this method needs other solutions to produce a higher precise result, probably because the accuracy is not enough. Eventually, Tested dataset and Evaluation metric are an important part of characteristics for learning-based methods because a uniform and standardized indicator is provided for performance.

3.4. Methods with Probability Density Function. Point cloud registration based on a probability density function (PDF), such that using a statistical model for registration, is a well-studied problem [179, 180]. The key idea in this method is to represent data by a specific probability density function, such as Gaussian mixture model (GMM) and normal

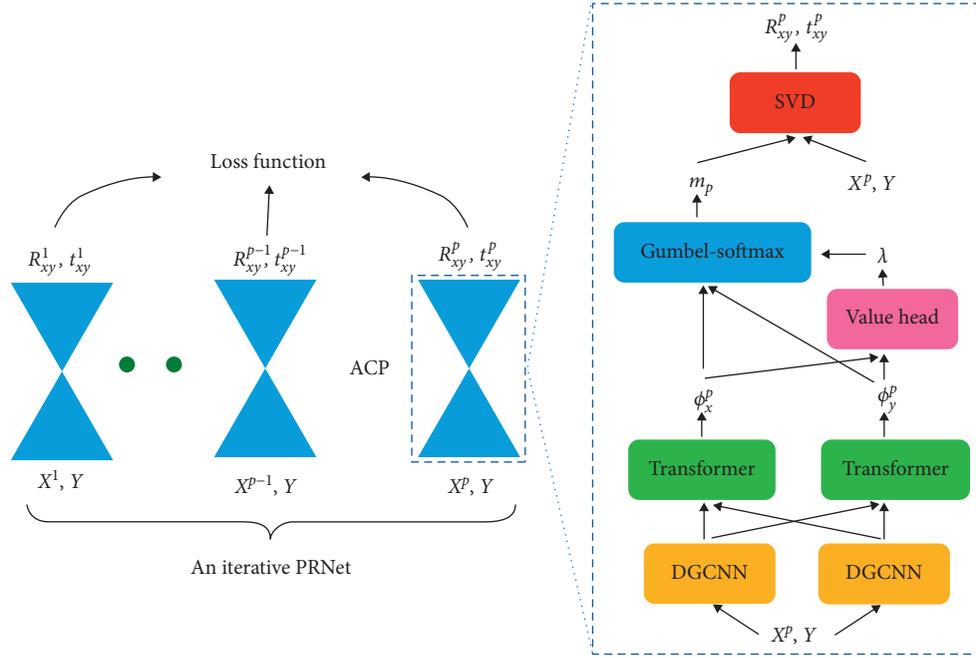


FIGURE 22: An architecture for PRNet, and $R_{x,y}^p, t_{x,y}^p$ denote the p -th rigid registration predicted by PRNet for point cloud X and Y .

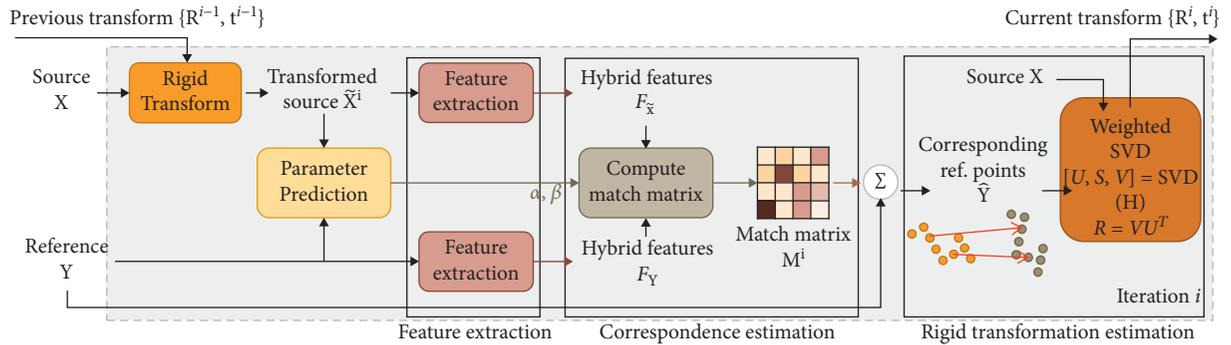


FIGURE 23: Overview of RPM-Net that illustrates the i th iteration, and the output of module, compute match matrix, refers to the correspondences. Source data X can be considered as reading data in this article.

distribution (ND). Then, the registration task is reformulated to the problem of aligning two corresponding distribution, followed by an objective function measuring and minimizing a statistical discrepancy between them [179]. Meanwhile, due to the representation of PDF, the point cloud can be regarded as a distribution rather than many individual points, so that it avoids the estimation of correspondences and has a good performance of robustness to noise, but in general slower than ICP-based methods [181]. In this section, methods jointly with probability density function for point cloud registration are introduced.

3.4.1. Gaussian Mixture. The point cloud registration task can be reinterpreted as a mixture density estimation and likelihood maximum problem. Therefore, a general framework, namely, mixture point matching (MPM), was introduced by Chui et al. [25]. In particular, let us take a GMM under Euclidean transformation [180] as an example, given two point clouds that were defined by X and Y individually.

Let $\Omega_X = \{x_i, \sigma_{ix}^2, \phi_i^X | 1 \leq i < n\}$ and $\Omega_Y = \{y_j, \sigma_{jy}^2, \phi_j^Y | 1 \leq j < m\}$ be GMMs constructed from X and Y , with means single point x_i and y_j , variances σ_{ix}^2 and σ_{jy}^2 , mixture weights ϕ_i^X and ϕ_j^Y , and the number of items n and m , respectively. Then, the L_2 distance transformed Ω_X and Ω_Y is given by

$$L_2(R, t) = \int_{\mathbb{R}^3} (p(\mathbf{p}|T(\Omega_X, R, t)) - p(\mathbf{p}|\Omega_Y))^2 d\mathbf{p}, \quad (7)$$

where $T(\Omega, R, t)$ denotes the function that rigidly transforms Ω with R and t , and $p(\mathbf{p}|\Omega)$ is the probability of observing a point \mathbf{p} under a mixture model $\Omega = \{\mu_i, \sigma_i^2, \phi_i\}$, defined by

$$p(\mathbf{p}|\Omega) = \sum_{i=1} \phi_i \Gamma(\mathbf{p}|\mu_i, \sigma_i^2), \quad (8)$$

where $\Gamma(\mathbf{p}|\mu_i, \sigma_i^2)$ is the probability of Gaussian, $[p(\mathbf{p}|T(\Omega_X, R, t))]^2$ is invariant under rigid transformation, and $[p(\mathbf{p}|\Omega_Y)]^2$ is independent of (R, t) . Then, the closed-form expression for an objective function is

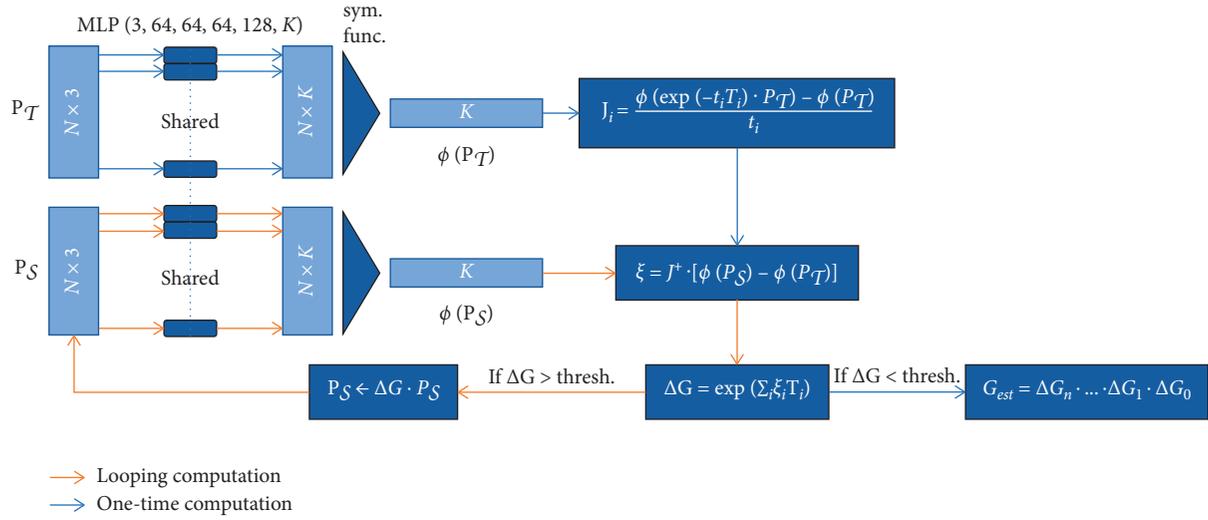


FIGURE 24: Model of PointNetLK, and the final estimate G_{est} aligns P_S to point cloud P_T .

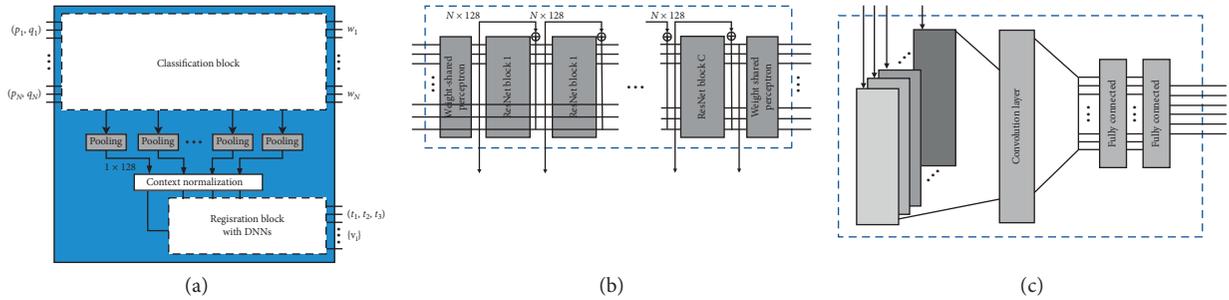


FIGURE 25: 3D registration network. (a) Overview of 3DRegNet consists of classification and the registration blocks. (b) Classification block. (c) Registration block with DNNs.

$$f(R, t) = - \int_{\mathbb{R}^3} (p(\mathbf{p}|T(\Omega_X, R, t)) - p(\mathbf{p}|\Omega_Y)) d\mathbf{p}, \quad (9)$$

which can be minimized to produce (R, t) through kernel density estimation (KDE), [182] expectation-maximization [183], or support vector machine (SVM) [184].

3.4.2. Coherent Point Drift. Coherent point drift (CPD) was introduced by Myronenko et al. [185] and further improved in [186]. Given two point clouds, similar to [25], CPD fits a GMM to the reference point cloud, and the Gaussian centroids are initialized from the points of reading data. The core of the CPD method is to coherently move GMM centroids as a group to preserve the topological structure of the point cloud. In CPD-based registration, a weighted GMM probability density function is defined as

$$\begin{cases} p(x) = w \frac{1}{N} + (1-w) \sum_{m=1}^M \frac{1}{M} p(x|m), \\ p(x|m) = \frac{1}{(2\pi\sigma^2)^{D/2}} \exp\left(-\frac{\|x - y_m\|_2}{2\sigma^2}\right), \end{cases} \quad (10)$$

where M is the number of points $Y\{y_1, y_2, \dots, y_m\}$ in reading point cloud, N denotes the number of points in reference data, and $w(0 \leq w \leq 1)$ is a weight parameter that reflects distribution of the amount of noise and outliers in point cloud. It is generated by reference data comprising $X = \{x_1, x_2, \dots, x_n\}$ and capable of explicitly modeling outliers. The GMM centroid location can be adjusted by a set of transformation θ and σ^2 that can be estimated by minimizing the negative log-likelihood function in equation (11), under data assumption of independent and identically distributed:

$$E(\theta, \sigma^2) = - \sum_{n=1}^M \log \sum_{m=1}^{M+1} p(m) p(x_n|m). \quad (11)$$

It is usually solved by the EM algorithm to find θ and σ^2 [183]. The EM algorithm proceeds by alternating P^{new} and P^{old} during expectation and maximum steps until convergence.

Many modifications of the original CPD algorithm were introduced. For instance, to accelerate the speed of the CPD algorithm, an accelerated CPD algorithm was presented in [187] that incorporated squared iterative expectation-maximization to converge to the global extreme

TABLE 3: Comparison among these learning-based methods.

Year	Method	Network type (for correspondence estimation and transformation estimation)	Need refinement	Tested dataset	Evaluation metric	Performance	Tested method
2020	Deep global registration (DGR)	Fully convolutional geometric features (FCGFs)/weighted Procrustes method	No	3DMatch	Translation error rotation error	Better than 3DRegNet, DCP, PRNet, and PointNetLK	—
2020	Robust point matching network (RPM-Net)	PointNet + FPFH/weighted SVD	No	ModelNet40	Mean isotropic Rotation error Mean isotropic Translation error	Better than PointNetLK and DCP-v2	—
2020	3D registration network (3DRegNet)	ResNet + FPFH/DNNs	No	ICL-NUIM, SUN3D	Translation error Rotation error	—	Only 3DRegNet
2019	Point cloud registration network (PCRNet)	PointNet/fully connected layers	No	ModelNet40	Axis-angle of rotation error L2-norm of translation error Area under the curve	Better than PointNetLK	Iterative PCRNet
2019	Partial registration network (PRNet)	DGCNN and transformer/an Actor-critic method	Yes	ModelNet40	Root mean squared error Mean absolute error Coefficient of determination	Better than PointNetLK and DCP-v2	—
2019	Deep closest point (DCP)	Dynamic graph convolutional networks (DGCNNs)/singular value decomposition (SVD) layer	Yes	ModelNet40	Root mean squared error Mean absolute error	Better than PointNetLK	DCP-v2 (DCP with attention module)
2019	PointNetLK	PointNet/LK algorithm	No	ModelNet40	Translation error Rotation error	—	—

“—” in Performance implies that this method compares with traditional solutions rather than the learning-based techniques. The Tested method refers explicitly to the solution tested.

and keep the same monotonically as the EM algorithm [188], and a Dual-Tree Improved Fast Gauss Transform (DT-IFGT) method was used to further accelerate CPD algorithm. To avoid setting the value of w in equation (10), Peng et al. [189] presented a refined CPD algorithm using a hybrid optimization, and experiments showed that it was better than the traditional CPD algorithm in terms of robustness and accuracy. Jun et al. [190] modified the original CPD algorithm with the voxel-grid filter before processing to improve the speed of registration while maintaining the same accuracy.

3.4.3. Normal Distribution Transform. The normal distribution transform (NDT) algorithm [191, 192] regards input data as probability density function by the normal distribution of every cell (i.e., voxel box as discussed above) in the point cloud. It calculates the transformation between pairwise data as a minimization problem [193]. The probability density function is defined as

$$p(x) = \frac{1}{c} \left(-\frac{(x-q)^T C^{-1} (x-q)}{2} \right), \quad (12)$$

where q and C are the mean vector and covariance matrix of every cell that contains point x , and c is a normalizing constant that can be set to one. q and C are calculated as

$$q = \frac{1}{n} \sum_{k=1}^n x_k, \quad (13)$$

$$C = \frac{1}{n-1} \sum_{k=1}^n (x_k - q)(x_k - q)^T,$$

where $\{x_k | k \in n\}$ is the point contained in the cell and n is the number of point in a cell. To formulate an optimization problem as a minimization problem, a score function of a pose \mathbf{p} and transformation function $T(\mathbf{p}, x)$ is defined as

$$s(\mathbf{p}) = - \sum_{k=1}^n p(T(\mathbf{p}, x_k)), \quad (14)$$

followed by $\mathbf{H}\Delta\mathbf{p} = -\mathbf{g}$, where \mathbf{H} and \mathbf{g} are the Hessian and gradient of s . The increment $\Delta\mathbf{p}$ is solved by Newton's algorithm iteratively or Levenberg–Marquardt algorithm [191, 194, 195] and can be added to current estimate in each iteration.

NDT-based registration has many advantages, including more robust and faster than transitional ICP algorithm when processing large-scale data, supporting the multiresolution representation of point cloud [196], and having no explicit correspondences [191]. However, it is hard to achieve high accuracy compared with ICP [197]. Additionally, a key challenge in the NDT algorithm is how to choose an appropriate cell size for the point cloud because the accuracy and speed depend on the cell size. In particular, when cell size is small, it requires more computing quantity and memory, while if the cell size is big, the resolution of cells is high so that it results in less accurate registration [193]. Nevertheless, due to the attractive contribution of the NDT algorithm, an NDT-based solution is still a good choice in a fine-to-coarse strategy at the coarse step [198, 199, 200]. Therefore, many modifications related to the NDT algorithm were proposed by researchers. For example, Yong et al. [201] introduced an improved normal distribution transform algorithm in a precise registration step. In this method, PDF is replaced by a mixed probability density function. Multilayered NDT [202], key-layered NDT [203], and variable size voxel NDT [204] were proposed to determine an appropriate size of cells, for converging to the optimal result and achieving high-quality registration. To avoid the discontinuity of the cost function, i.e., there are points existing at the cell boundaries, Arun et al. [205] introduced a segmented region growing NDT that generates the Gaussian clusters after removing the ground plane for accurate registration.

3.5. Others. Along with the development of the ICP-based, feature-based, learning-based solutions, and probability density function, the problem of point cloud registration also can be solved through other methods. Specifically, as we all know that the transformation between two point clouds can be easily found out by a set of correspondences. Therefore, Aiger et al. [206] focused on looking for the high confidence correspondences directly that only consist of four congruent points. Furthermore, auxiliary data generated from auxiliary approaches, such as image-assisted and Global Navigation Satellite System (GNSS), can also be used to simultaneously provide critical information for the task of point cloud registration.

3.5.1. Fast Global Registration. Fast global registration (FGR) [31] provided a fast strategy for point cloud registration where no initialization is required. Specifically, FGR operates on candidate matches that cover the surfaces of objects, and no correspondence updates or closest-point queries are performed. The special feature of this method is that a joint registration can be produced directly through a single optimization of a robust objective defined densely over the surface. However, the existing approaches to solve

point cloud registration usually produce candidates or more correspondences between two point clouds and then compute and update a global result. Moreover, in the fast global registration, the correspondences are established at once in optimization and will not be estimated again in the following steps, as shown in Figure 26. Therefore, the expensive nearest-neighbor lookup is avoided to keep at a low computational cost. As a result, the line process for each correspondence and a linear system for pose estimation in the iteration step are efficient. FGR evaluates on multiple datasets such as UWA benchmark [207] and Stanford Bunny [208], compared with ICP with point-to-point and point-to-plane, and ICP variants such as the Go-ICP. Experiments showed that FGR outperforms in the presence of noise.

3.5.2. 4-Points Congruent Sets Algorithm. 4-points congruent sets (4PCS) [206] provided an initial transformation for reading data without starting position assumption. Generally, a rigid registration transformation between two point clouds can be uniquely defined by a pair of triplets in which one from reference data and one from reading data. However, in this method, it looks for special 4-points bases, i.e., 4-coplanar congruent points from each point cloud, by searching in small potentially sets, as illustrated in Figure 27. The best rigid transformation is solved in the largest common point set (LCP) problem. The algorithm achieves close performance when the overlap rate of the pairwise point cloud is low, and there are outliers. In order to adapt to different applications, many researchers introduced more significant work related to classic 4PCS solution [209–211]. For example, in this work [212], in contrast to the quadratic time complexity of traditional 4PCS, it presented a super-4PCS that achieved the performance with linear running time. Pascal et al. [213] proposed a keypoint-based 4-points congruent sets (K-4PCS) solution for coarse registration of point cloud through extracting 3D key points as the input that can be a discriminative representation of the reading data. Semantic-keypoint-based 4PCS proposed by Xuming [214] handles massive point cloud and improves K-4PCS in terms of accuracy and computational efficiency through semantic features for urban building scenes.

3.5.3. Methods with Auxiliary Data. In various registration tasks, the problem of finding the transformation between pairwise or multiview can be transformed into estimating the spatial position of acquisitions in a common coordinate. Specifically, when retrieving point cloud data, some techniques worked with a depth camera can provide its location and orientation in real-time, such as the Global Navigation Satellite System (GNSS), inertial measurement unit (IMU), and image-based algorithms. Specifically, GNSS refers to a constellation of satellites that produce positioning and timing data to GNSS receivers. These data can facilitate the determination of location. IMU is an electronic device to measure the orientation, velocity, and gravitational forces of the body. Image-based algorithms have been widely used in simultaneous localization and mapping (SLAM) technique that aims to localize an autonomous mobile robot in an

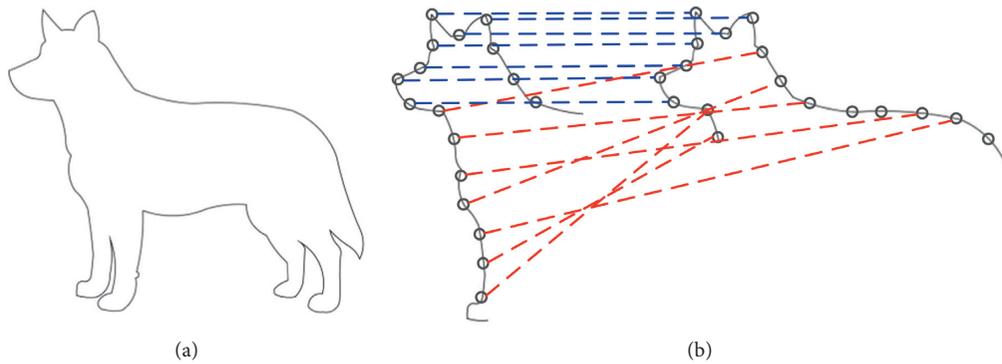


FIGURE 26: Fast global registration: correspondences are generated before optimization. (a) Latent shape. (b) Potential correspondences that consists of genuine and erroneous ones.

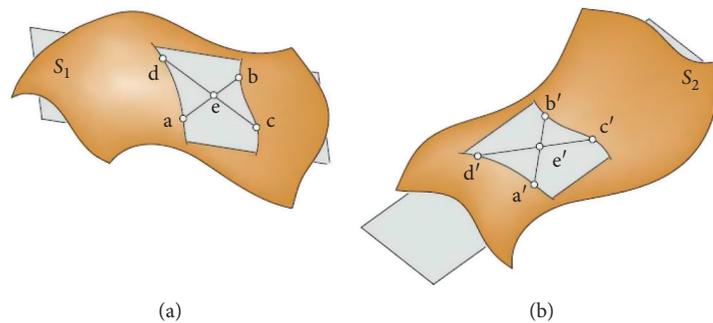


FIGURE 27: A pair of 4-points sets in the 4PCS algorithm: S_2 is another surface that you can image, and it is constructed by point cloud. S_2 matches S_1 , and the 4-points coplanar base lies in the overlap region represented by $a, b, c,$ and d .

unknown environment while building a consistent and increment map around the robot [215].

In applications related to mobile mapping systems (MMSs) in which a rapid registration process is required and too large data are collected at once, a GNSS-IMU assistance system for forest mapping that can provide rough registration simultaneously is presented [216, 217]. However, the GNSS signal usually loses under some complex areas such as building occlusion and under the canopy. Marek et al. [218] proposed the use of graph-SLAM [219] to generate a local map of forest such that a stereo camera focuses on collecting data and a GPS is used to calculate an estimate of the global tree positions, wherein a 3D map is first generated by vision odometry and then improved by robust graph-SLAM. For 3D city models with color information, Bernard et al. [220] estimated the relative camera parameters from aerial image sequences as a coarse transformation for ICP algorithm refinement. Xuming et al. [221] introduced an image-driven registration for urban scenes, where a scanning network was constructed by exploiting the image information in order to guide the coarse pairwise registration task.

3.6. Applications. The point cloud registration is a critical requisition for computer vision and robotics. However, the registration problem in computer vision usually is converted to the image matching problem (i.e., image registration) which is studied and researched in many fields, such as the

medical domain [222, 223], handwritten digit recognition [224], and image stitching [225, 226].

For the point cloud registration, the robotic application is a hot topic. Pomerleau [34] provided four types of applications that consist of several good use cases covering different kinds of platforms, environments, and tasks. The first type of applications is on search and rescue tasks that need a real-time registration, in which 3D maps are required and can be used to provide the users with situation awareness and support critical decisions about risky platform motions. In this field, 3D map reconstruction is a basic and popular area of research, but it is merely a single application without more use cases. The second type of applications is on automation of inspection. Some tasks need to execute inspection or maintenance operations in environments that are difficult to access by humans due to dimensional, temperature, or air quality constraints. The third type of applications is on the development of an autonomous surface vessel in order to support environmental monitoring of freshwater bodies, in which a 3D laser was installed to complement the analysis of the ecosystem with geological information. The fourth type of applications is on an autonomous-driving or self-driving car. In addition, recently the deep learning techniques based on the point cloud are widely used [227] which take the raw point cloud as input directly.

3.7. Summary. In this section, five categories of solutions and methods for solving point cloud registration are introduced. Specifically, in the ICP-based methods, its

steps, variants, and implementation details are described, and then the popular and open-source libraries or software is given. To use the ICP-based algorithm better, we need to know that ICP-based algorithms can obtain a good alignment result if providing an initial position for a point cloud to be aligned. The key is how to generate a good initial position for your applications. Therefore, for applications that need point cloud registration, a popular strategy was proposed [228, 229], which consists of two steps: (1) computing a global transformation as an initial position for the point cloud and (2) calculating a local transformation that can guarantee the result of registration. These two steps are called coarse-to-fine strategies. In this case, feature-based, learning-based, and probability-based methods are usually used to produce a global transformation, and then further improvements are made by the ICP-based algorithms. The point-to-point and point-to-plane ICP methods are widely used in real-world situations [39, 90]. To use the ICP algorithms better, many scholars considered the evaluation of ICP and its variants. For example, Peng et al. [39] took overlap ratio, angle, and distance as the influence factors of ICP and evaluated the performance on the same four datasets. In addition, Martin et al. [197] and Mouna et al. [230] compared the ICP and NDT in terms of accuracy and speed. Lachat [231] studied different software and tools to solve point cloud registration in terrestrial and airborne laser scanning. To overcome the bias on experiments, Pomerleau [232] provided datasets that cover a diverse range of challenging environments for registration algorithms.

Feature-based and learning-based methods are studied increasingly in recent years, because in general the point cloud registration problem was straightforwardly divided into several modules: feature extractor, matching, outlier rejection, and motion estimation. For the feature extractor module, in addition to the PFH and SHOT, what we describe in this section is the local-based feature. Various methods to extract the feature and description of the point cloud are also studied and summarized [140]. Feature-based methods usually extract pairwise or higher-order relationships in the histogram in a handcrafted way. However, many of these handcrafted descriptors that highly rely on distinctive geometric properties of point cloud data merely work well on real-world point clouds [233] that are noisy and low density. Learning-based methods involve regressing the transformation parameters by searching for the difference between features which generally are global-based. The state-of-art methods presented recent years are introduced, and a brief comparison of their performance and modules is provided. They are less sensitive and more robust on real-world data. To obtain a good initial transformation, the learning-based and feature-based methods can be helpful. For better using these techniques, we should pay attention to the following points: (1) a feature detection algorithm should be considered, which attains the trade-off of accuracy and efficiency. (2) A learning framework should be established, particularly for the applications containing data collection and training. In addition, probability-based methods are also

used for initial pose estimation, such as NDT and CPD that are often utilized for nonrigid registration.

4. Conclusion and Outlook

Point cloud registration is a crucial step for many applications that require to build a 3D map environment and recover the 3D surface of an object or scene. However, it is impossible to retrieve all surface data at once or directly obtain the relationship between two point clouds in each coordinate system. Therefore, to place a strong emphasis on the comprehensive review of pairwise point cloud registration with various state-of-the-art algorithms, this paper begins by the basics and principles of point cloud registration, and it then goes on to divide the registration process into five main classifications: ICP-based algorithm, feature-based, machine learning-based, probability-based solution, and others. With these registration categories, we provide insight that every method has its applicable fields and disadvantages. For two completely identical point clouds with an arbitrary position, a valid transformation could be computed by traditional algorithms, such as ICP-based algorithms and feature-based solutions. For two acquired point clouds, machine-learning methods have been developed to achieve more robustness and accuracy than traditional algorithms based on experiments. Therefore, learning-based methods have attracted more and more attention for providing a new perspective and have achieved great success for solving point cloud registration. However, in real-world data, the task of registration will face uncertain situations, such as the variant of overlap, fluctuation of noise, and the huge amount of computation due to massive input data. Therefore, developing a general framework to solve the point cloud registration in the real world is still a challenging task. There is a brief guideline for those who want to use point cloud registration as a tool: when it has the apparent characteristics of the captured objects, feature-based is a good choice for your applications that need point cloud registration. However, when it is not sure what the retrieved objects are, machine-based and probability-based solutions can always provide a reliable transformation as a rough result. In this case, an ICP-based algorithm should be applied to achieve a refined registration.

With the advancement of 3D imaging technology, the resolution of point cloud data and the number of point sets will increase dramatically. For instance, a point cloud accurately representing the details on a workpiece acquired from the structured light camera (SLC) now has more than tens of millions of points. One frame data approximately describing the subway station taken by a LiDAR have at least five million points. Therefore, a technique that probably is integrated with hardware and software optimization to swiftly process and operate these unstructured point cloud data is required in the future. Additionally, as a quantitative measurement, a ubiquitous dataset containing ground-truth transformation and different environments, such as indoor and outdoor scenes, is beneficial for evaluating learning-based methods and previous strategies. As more and more public datasets are expected to be established in the future, we

believe that a comparison between traditional algorithms and machine-learning methods is essential for us to choose an appropriate registration solution for a specific application.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] Q. Wang and M.-K. Kim, "Applications of 3d point cloud data in the construction industry: a fifteen-year review from 2004 to 2018," *Advanced Engineering Informatics*, vol. 39, pp. 306–319, 2019.
- [2] R. W. Wolcott and R. M. Eustice, "Robust lidar localization using multiresolution Gaussian mixture maps for autonomous driving," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 292–319, 2017.
- [3] X. Yue, B. Wu, S. A. Seshia, K. Keutzer, and A. L. Sangiovanni-Vincentelli, "A lidar point cloud generator: from a virtual world to autonomous driving," in *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pp. 458–464, Yokohama, Japan, 2018.
- [4] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-net: towards learning based lidar localization for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6389–6398, Long Beach, CA, USA, 2019.
- [5] R. Wang, Y. Xu, M. A. Sotelo et al., "A robust registration method for autonomous driving pose estimation in urban dynamic environment using lidar," *Electronics*, vol. 8, no. 1, p. 43, 2019.
- [6] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-time 3d reconstruction in dynamic scenes using point-based fusion," in *Proceedings of the 2013 International Conference on 3D Vision-3DV*, vol. 2013, Washington, DC, 2013.
- [7] M. Donlic, T. Petkovic, and T. Pribanic, "On tablet 3d structured light reconstruction and registration," in *Proceedings of the The IEEE International Conference on Computer Vision (ICCV) Workshops*, Venice, Italy, October 2017.
- [8] B. Fang and C. Wang, "A 3d reconstruction method based on the combination of the icp and artificial potential field algorithm," *Lecture Notes in Electrical Engineering*, Springer, Berlin, Germany, pp. 483–493, 2017.
- [9] P. Wu, W. Li, and M. Yan, "3d scene reconstruction based on improved ICP algorithm," *Microprocessors and Microsystems*, Article ID 103064, 2020.
- [10] S. Izadi, D. Kim, O. Hilliges et al., "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of the 24th annual ACM symposium on user interface software and technology*, pp. 559–568, New Orleans, LA, USA, 2011.
- [11] E. Mendes, P. Koch, and S. Lacroix, "Icp-based pose-graph slam," in *Proceedings of the 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 195–200, Hokkaido, Japan, 2016.
- [12] H. Cho, E. K. Kim, and S. Kim, "Indoor slam application using geometric and ICP matching methods based on line features," *Robotics and Autonomous Systems*, vol. 100, pp. 206–224, 2018.
- [13] P. Kim, J. Chen, and Y. K. Cho, "Slam-driven robotic mapping and registration of 3d point clouds," *Automation in Construction*, vol. 89, pp. 38–48, 2018.
- [14] L. Han, L. Xu, D. Bobkov, E. Steinbach, and L. Fang, "Real-time global registration for globally consistent rgb-d slam," *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 498–508, 2019.
- [15] Q. Qian, *Multi-camera registration for Vr: a flexible, feature-based approach*, Master's thesis, Technische Universiteit Delft, Delft, Netherlands, 2018.
- [16] D. Borrmann, A. Nuechter, and T. Wiemann, "Large-scale 3d point cloud processing for mixed and augmented reality," in *Proceedings of the 2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, vol. 35, Munich, Germany, 2018.
- [17] D. Ni, A. Song, X. Xu, H. Li, C. Zhu, and H. Zeng, "3d-point-cloud registration and real-world dynamic modelling-based virtual environment building method for teleoperation," *Robotica*, vol. 35, no. 10, pp. 1958–1974, 2017.
- [18] S. Xu, J. Zhu, Z. Jiang, Z. Lin, J. Lu, and Z. Li, "Multi-view registration of unordered range scans by fast correspondence propagation of multi-scale descriptors," *PloS One*, vol. 13, no. 9, Article ID e0203139, 2018.
- [19] H. Zhu, B. Guo, K. Zou et al., "A review of point set registration: from pairwise registration to groupwise registration," *Sensors*, vol. 19, no. 5, p. 1191, 2019.
- [20] L. Cheng, S. Chen, X. Liu et al., "Registration of laser scanning point clouds: a review," *Sensors*, vol. 18, no. 5, p. 1641, 2018.
- [21] G. K. L. Tam, Z. Zhi-Quan, Y. Yu-Kun Lai et al., "Registration of 3d point clouds and meshes: a survey from rigid to nonrigid," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 7, pp. 1199–1217, 2013.
- [22] B. Yang, Z. Dong, F. Liang, and Y. Liu, "Automatic registration of large-scale urban scene point clouds based on semantic feature points," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 113, pp. 43–58, 2016.
- [23] Y. Wang and J. M. Solomon, "Prnet: self-supervised learning for partial-to-partial registration," *Advances in Neural Information Processing Systems*, pp. 8814–8826, 2019.
- [24] C. Choy, W. Dong, and V. Koltun, "Deep global registration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2514–2523, 2020.
- [25] H. Chui and A. Rangarajan, "A feature registration framework using mixture models," in *Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pp. 190–197, IEEE, San Francisco, CA, USA, 2000.
- [26] Y. He, B. Liang, J. Yang, S. Li, and J. He, "An iterative closest points algorithm for registration of 3d laser scanner point clouds with geometric features," *Sensors*, vol. 17, no. 8, p. 1862, 2017.
- [27] B. Radu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, pp. 3212–3217, IEEE, Kobe, Japan, 2009.
- [28] B. Radu, N. Blodow, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3384–3391, IEEE, Breazeal, Cynthia, 2008.

- [29] J. Yang, Z. Cao, and Q. Zhang, "A fast and robust local descriptor for 3d point cloud registration," *Information Sciences*, vol. 346-347, pp. 163-179, 2016.
- [30] A. Gruen and D. Akca, "Least squares 3d surface and curve matching," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 59, no. 3, pp. 151-174, 2005.
- [31] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration, computer vision - ECCV 2016," in *Proceedings of the European Conference on Computer Vision*, pp. 766-782, Springer, Glasgow, UK, 2016.
- [32] K. Alhamzi, M. Elmogy, and S. Barakat, "3d object recognition based on local and global features using point cloud library," *International Journal of Advancements in Computing Technology*, vol. 7, no. 3, p. 43, 2015.
- [33] L. F. Robert, N. Gros, Y. Noutary, L. Malleus, F. Precioso, and D. Lingrand, "Keypoint-based point-pair-feature for scalable automatic global registration of large rgb-d scans," US Patent 10,217,277, 2019.
- [34] F. Pomerleau, F. Colas, and R. Siegwart, "A review of point cloud registration algorithms for mobile robotics," *Foundations and Trends in Robotics*, vol. 4, no. 1, pp. 1-104, 2015.
- [35] G. K. L. Tam, Z. Zhi-Quan Cheng, Y. Yu-Kun Lai et al., "Registration of 3d point clouds and meshes: a survey from rigid to nonrigid," *Visualization and Computer Graphics*, vol. 18, 2013.
- [36] H. Li and R. Hartley, "The 3d-3d registration problem revisited," in *Proceedings of the 2007 IEEE 11th International Conference on Computer Vision*, pp. 1-8, IEEE, Rio De Janeiro, Brazil, 2007.
- [37] B. Bellekens, S. Vincent, R. Berkvens, and M. Weyn, "A survey of rigid 3d pointcloud registration algorithms," in *Proceedings of the AMBIENT 2014: The Fourth International Conference on Ambient Computing, Applications*, pp. 8-13, Services and Technologies, Rome, Italy, 2014.
- [38] Z. Dong, F. Liang, B. Yang et al., "Registration of large-scale terrestrial laser scanner point clouds: a review and benchmark," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 163, pp. 327-342, 2020.
- [39] Li Peng, R. Wang, Y. Wang, and W. Tao, "Evaluation of the icp algorithm in 3d point cloud registration," *IEEE Access*, vol. 8, pp. 68030-68048, 2020.
- [40] X. Jiang, J. Ma, G. Xiao, Z. Shao, and X. Guo, "A review of multimodal image matching: methods and applications," *Information Fusion*, 2021.
- [41] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, "Image matching from handcrafted to deep features: a survey," *International Journal of Computer Vision*, vol. 129, no. 1, pp. 23-79, 2021.
- [42] A. Zabatani, V. Surazhsky, E. Sperling et al., "Intel® real-sense™ sr300 coded light depth camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [43] O. Ozyesil, V. Voroninski, B. Ronen, and A. Singer, "A survey of structure from motion," *Acta Numerica*, vol. 26, pp. 305-364, 2017.
- [44] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362-1376, 2009.
- [45] R. Chen, S. Han, J. Xu, and H. Su, "Point-based multi-view stereo network," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1538-1547, Seoul, Korea, 2019.
- [46] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards internet-scale multi-view stereo," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1434-1441, IEEE, Seoul, Korea, 2010.
- [47] A. Grunnet-Jepsen, J. N. Sweetser, W. Paul, A. Takagi, and J. Woodfill, *Projectors for Intel® real-sense™ Depth Cameras D4xx. Intel Support*, Intel Corporation, Santa Clara, CA, USA, 2018.
- [48] A. Goshtasby and W. A. Gruver, "Design of a single-lens stereo camera system," *Pattern Recognition*, vol. 26, no. 6, pp. 923-937, 1993.
- [49] A. Pezzuolo, D. Giora, L. Sartori, and S. Guercini, "Automated 3d reconstruction of rural buildings from structure-from-motion (sfm) photogrammetry approach," in *Proceedings of the International Scientific Conference*, pp. 23-25, Vladivostok, Russia, 2018.
- [50] S. Zhu, R. Zhang, L. Zhou et al., "Very large-scale global sfm by distributed motion averaging," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4568-4577, Salt Lake City, UT, USA, 2018.
- [51] V. Usenko, J. Engel, Jörg Stückler, and D. Cremers, "Reconstructing street-scenes in real-time from a driving car," in *Proceedings of the 2015 International Conference on 3D Vision*, pp. 607-614, IEEE, Lyon, France, 2015.
- [52] P. Gronat, M. Havlena, J. Sivic, and P. Tomas, *Building Streetview Datasets for Place Recognition and City reconstruction*, Czech Technical University, Prague, Czech Republic, 2011.
- [53] Y. Xu, V. John, S. Mita, H. Tehrani, K. Ishimaru, and S. Nishino, "3d point cloud map based vehicle localization using stereo camera," in *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium*, pp. 487-492, IEEE, Dearborn, MI, USA, 2017.
- [54] Z. Gong, W. Xue, Z. Liu et al., "Design of a reconfigurable multi-sensor testbed for autonomous vehicles and ground robots," in *Proceedings of the 2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, IEEE, Hokkaido, Japan, 2019.
- [55] E. Ackerman, "Lidar that will make self-driving cars affordable [news]," *IEEE Spectrum*, vol. 53, no. 10, p. 14, 2016.
- [56] Michael Van den Bergh and Luc Van Gool, "Combining rgb and tof cameras for real-time 3d hand gesture interaction," in *Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV)*, pp. 66-72, IEEE, Washington, DC, 2011.
- [57] T. Bell, B. Li, and S. Zhang, "Structured light techniques and applications," *Electrical and Electronics Engineering*, pp. 1-24, 1999.
- [58] S. Zhang, "Recent progresses on real-time 3d shape measurement using digital fringe projection techniques," *Optics and Lasers in Engineering*, vol. 48, no. 2, pp. 149-158, 2010.
- [59] S. Zhang, "High-speed 3d shape measurement with structured light methods: a review," *Optics and Lasers in Engineering*, vol. 106, pp. 119-131, 2018.
- [60] D. M. McClatchy, E. J. Rizzo, J. Meganck et al., "Calibration and analysis of a multimodal micro-ct and structured light imaging system for the evaluation of excised breast tissue," *Physics in Medicine & Biology*, vol. 62, no. 23, pp. 8983-9000, 2017.
- [61] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: the kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231-1237, 2013.
- [62] A. Behl, D. Paschalidou, D. Simon, and A. Geiger, "Point-flownet: learning representations for rigid motion estimation from point clouds," in *Proceedings of the IEEE/CVF*

- Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019.
- [63] B. J. Brown and S. Rusinkiewicz, "Global non-rigid alignment of 3-d scans," in *Proceedings of the ACM SIGGRAPH 2007 papers*, New York, NY, USA, 2007.
- [64] H. Li, R. W. Sumner, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," *Computer Graphics Forum*, vol. 275, pp. 1421–1430, 2008.
- [65] Federica Bogo, J. Romero, G. Pons-Moll, and J. Michael, "Dynamic faust: registering human bodies in motion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.
- [66] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: reconstruction and tracking of non-rigid scenes in real-time," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 343–352, Boston, MA, USA, 2015.
- [67] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, 2003.
- [68] S. Chang, C. Ahn, M. Lee, and S. Oh, "Graph-matching-based correspondence search for nonrigid point cloud registration," *Computer Vision and Image Understanding*, vol. 192, Article ID 102899, 2020.
- [69] R. A. Newcombe, S. Izadi, O. Hilliges et al., "Kinectfusion: real-time dense surface mapping and tracking," in *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pp. 127–136, IEEE, Beijing, China, 2011.
- [70] A. Ullrich and M. Pfennigbauer, "Noisy lidar point clouds: impact on information extraction in high-precision lidar surveying," in *Proceedings of the Laser Radar Technology and Applications*, vol. 28, 2018, Article ID 10636.
- [71] M. Berger, A. Tagliasacchi, S. Lee et al., "State of the art in surface reconstruction from point clouds," in *Proceedings of the Eurographics 2014-State of the Art Reports*, vol. 1, no. 1, pp. 161–185, Strasbourg, France, 2014.
- [72] P. Schaer, S. Jan, S. Landtwin, and K. Legat, "Accuracy estimation for laser point cloud including scanning geometry," in *Proceedings of the Mobile Mapping Symposium 2007*, Padova, Italy, 2007.
- [73] J. Park, H. Kim, Y. W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3d-tof cameras," in *Proceedings of the 2011 International Conference on Computer Vision*, pp. 1623–1630, IEEE, Washington, DC, 2011.
- [74] Y. Wang and H.-Y. Feng, "Effects of scanning orientation on outlier formation in 3d laser scanning of reflective surfaces," *Optics and Lasers in Engineering*, vol. 81, pp. 35–45, 2016.
- [75] H. Chen and J. Shen, "Denoising of point cloud data for computer-aided design, engineering, and manufacturing," *Engineering with Computers*, vol. 34, no. 3, pp. 523–541, 2018.
- [76] S. Li, J. Wang, Z. Liang, and L. Su, "Tree point clouds registration using an improved icp algorithm based on kd-tree," in *Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 4545–4548, IEEE, Honolulu, HI, USA, 2016.
- [77] L. Di Angelo and L. Giaccari, "An efficient algorithm for the nearest neighbourhood search for point clouds," *International Journal of Computer Science Issues (IJCSI)*, vol. 8, no. 5, p. 1, 2011.
- [78] K. Zhou, Q. Hou, R. Wang, and B. Guo, "Real-time kd-tree construction on graphics hardware," *ACM Transactions on Graphics*, vol. 27, no. 5, pp. 1–11, 2008.
- [79] M. Connor and P. Kumar, "Fast construction of k-nearest neighbor graphs for point clouds," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 4, pp. 599–608, 2010.
- [80] A. Nuchter, K. Lingemann, and J. Hertzberg, "Cached kd tree search for icp algorithms," in *Proceedings of the Sixth International Conference on 3-D Digital Imaging and Modeling*, pp. 419–426, IEEE, Washington, DC, 2007.
- [81] J. Li, B. M. Chen, and G. H. Lee, "So-net: self-organizing network for point cloud analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9397–9406, Salt Lake City, UT, USA, 2018.
- [82] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling*, pp. 145–152, IEEE, Quebec, Canada, 2001.
- [83] D. Holz, A. E. Ichim, F. Tombari, R. B. Rusu, and S. Behnke, "Registration with the point cloud library: a modular framework for aligning in 3-d," *IEEE Robotics & Automation Magazine*, vol. 22, no. 4, pp. 110–124, 2015.
- [84] C. Benoit and S. Prima, "An efficient em-icp algorithm for non-linear registration of large 3d point sets," *Computer Vision and Image Understanding*, vol. 191, p. 102854, 2020.
- [85] T. Tamaki, M. Abe, B. Raytchev, and K. Kaneda, "Softassign and em-icp on gpu," in *Proceedings of the 2010 First International Conference on Networking and Computing*, pp. 179–183, IEEE, 2010.
- [86] S. Rusinkiewicz, "Efficient variants of the icp algorithm," in *Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling*, pp. 145–152, Quebec, Canada, 2011.
- [87] X.-F. Han, J. S. Jin, M.-J. Wang, W. Jiang, L. Gao, and L. Xiao, "A review of algorithms for filtering the 3d point cloud," *Signal Processing: Image Communication*, vol. 57, pp. 103–112, 2017.
- [88] M. Yilmaz and M. Uysal, "Comparison of data reduction algorithms for LiDAR-derived digital terrain model generalisation," *Area*, vol. 48, no. 4, pp. 521–532, 2016.
- [89] S. Chen, T. Dong, F. Chen, Anthony Vetro, and J. Kovačević, "Fast resampling of three-dimensional point clouds via graphs," *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 666–681, 2017.
- [90] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, February 2013.
- [91] R. Benjema and F. Schmitt, "Fast global registration of 3d sampled surfaces using a multi-z-buffer technique," *Image and Vision Computing*, vol. 17, no. 2, pp. 113–123, 1999.
- [92] N. Ko and K. Ikeuchi, "Robust simultaneous registration of multiple range images," in *Digitally Archiving Cultural Objects*, pp. 71–88, Springer, New York, NY, USA, 2008.
- [93] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.
- [94] D. Meagher, "Geometric modeling using octree encoding," *Computer Graphics and Image Processing*, vol. 19, no. 2, pp. 129–147, 1982.
- [95] W. Guan, W. Li, and Y. Ren, "Point cloud registration based on improved icp algorithm," in *Proceedings of the 2018 Chinese Control and Decision Conference (CCDC)*, pp. 1461–1465, Shenyang, China, 2018.
- [96] J. Meng, J. Li, and X. Gao, "An accelerated icp registration algorithm for 3d point cloud data," in *Proceedings of the 9th International Symposium on Advanced Optical*

- Manufacturing and Testing Technologies: Optical Test, Measurement Technology, and Equipment*, vol. 10839, 2019, Article ID 1083904.
- [97] S.-Y. Park and M. Subbarao, "An accurate and fast point-to-plane registration technique," *Pattern Recognition Letters*, vol. 24, no. 16, pp. 2967–2976, 2003.
- [98] G. Blais and M. D. Levine, "Registering multiview range data to create 3d computer objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 820–824, 1995.
- [99] K. Pulli, "Multiview registration for large data sets," in *Proceedings of the Second International Conference on 3-D Digital Imaging and Modeling*, pp. 160–168, IEEE, Ottawa, Canada, 1999.
- [100] S.-Y. Park, S.-I. Choi, J. Kim, and J. S. Chae, "Real-time 3d registration using gpu," *Machine Vision and Applications*, vol. 22, no. 5, pp. 837–850, 2011.
- [101] P. Yin, F. Gu, D. Li, Y. He, L. Yang, and J. Han, "Gpu-based heuristic escape for outdoor large scale registration," in *Proceedings of the 2016 IEEE International Conference on Real-Time Computing and Robotics (RCAR)*, pp. 260–265, IEEE, Angkor Wat, Cambodia, 2016.
- [102] S. Sun, W. Song, Y. Tian, and S. Fong, "An icp-based point clouds registration method for indoor environment modeling," in *Advanced Multimedia and Ubiquitous Engineering*, pp. 339–344, Springer, New York, NY, USA, 2019.
- [103] T. Zinßer, J. Schmidt, and N. Heinrich, "A refined icp algorithm for robust 3-d correspondence estimation," in *Proceedings 2003 International Conference on Image Processing*, vol. 2, pp. 2–695, IEEE, Barcelona, Spain, 2003.
- [104] T. Masuda, K. Sakaue, and N. Yokoya, "Registration and integration of multiple range images for 3-d model construction," in *Proceedings of the 13th International Conference On Pattern Recognition*, vol. 1, pp. 879–883, IEEE, Vienna, Austria, 1996.
- [105] H. Q. Luong, M. Vlamincik, G. Werner, and W. Philips, "Consistent icp for the registration of sparse and inhomogeneous point clouds," in *Proceedings of the 2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*, pp. 262–267, IEEE, Ha-Long, Vietnam, 2016.
- [106] S. Giancola, J. Schneider, W. Peter, and B. S. Ghanem, "Integration of absolute orientation measurements in the kinectfusion reconstruction pipeline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Salt Lake City, UT, USA, June 2018.
- [107] J. H. Back, S. Kim, and Yo-S. Ho, "High-precision 3d coarse registration using ransac and randomly-picked rejections," in *Proceedings of the International Conference on Multimedia Modeling*, pp. 254–266, Springer, Bangkok, Thailand, 2018.
- [108] A. Gressin, C. Mallet, and N. David, "Improving 3d lidar point cloud registration using optimal neighborhood knowledge," in *Proceedings of ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 5, no. 2, Melbourne, Australia, 2012.
- [109] A. Ahmad, C. Léger, D. Mohamed, and L. Roger, "3d registration using a new implementation of the icp algorithm based on a comprehensive lookup matrix: application to medical imaging," *Pattern Recognition Letters*, vol. 28, no. 12, pp. 1523–1533, 2007.
- [110] F. Monnier, B. Vallet, N. Paparoditis, J.-P. Papelard, and N. David, "Registration of terrestrial mobile laser data on 2d or 3d geographic database by use of a non-rigid icp approach," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 5, 2013.
- [111] L. D. Wen, T. J. J. Tang, T. Drummond, and W. H. Li, "Robust egomotion estimation using icp in inverse depth coordinates," in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation*, p. 1671, IEEE, St Paul, MN, USA, 2012.
- [112] K. Naus and L. Marchel, "Use of a weighted icp algorithm to precisely determine usv movement parameters," *Applied Sciences*, vol. 9, no. 17, p. 3530, 2019.
- [113] J. Liu, M. E. Rettmann, D. R. Holmes, H. Duan, and R. A. Robb, "A piecewise patch-to-model matching method for image-guided cardiac catheter ablation," *Computerized Medical Imaging and Graphics*, vol. 35, no. 4, pp. 324–332, 2011.
- [114] E. Héry, P. Xu, and P. Bonnifait, "Lidar based relative pose and covariance estimation for communicating vehicles exchanging a polygonal model of their shape," in *Proceedings of the 10th Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, Madrid, Spain, 2018.
- [115] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor Fusion Iv: Control Paradigms And Data Structures*, vol. volume 1611, pp. 586–606, International Society for Optics and Photonics, 1992.
- [116] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, 1987.
- [117] D. W. Eggert, A. Lorusso, and R. B. Fisher, "Estimating 3-d rigid body transformations: a comparison of four major algorithms," *Machine Vision and Applications*, vol. 9, no. 5-6, pp. 272–290, 1997.
- [118] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145–155, 1992.
- [119] R. Bergevin, M. Soucy, H. Gagnon, and D. Laurendeau, "Towards a general multi-view registration technique," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 5, pp. 540–547, 1996.
- [120] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society of America A*, vol. 5, no. 7, pp. 1127–1135, 1988.
- [121] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America A*, vol. 4, no. 4, pp. 629–642, 1987.
- [122] A. W. Fitzgibbon, "Robust registration of 2d and 3d point sets," *Image and Vision Computing*, vol. 21, no. 13-14, pp. 1145–1153, 2003.
- [123] P. J. Neugebauer, "Geometrical cloning of 3d objects via simultaneous registration of multiple range images," in *Proceedings of 1997 International Conference on Shape Modeling and Applications*, pp. 130–139, IEEE, Aizu-Wakamatsu, Japan, 1997.
- [124] A. Censi, "An icp variant using a point-to-line metric," in *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, pp. 19–25, IEEE, Pasadena, CA, USA, 2008.
- [125] H. Men, B. Gebre, and K. Pochiraju, "Color point cloud registration with 4d icp algorithm," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*, pp. 1511–1516, IEEE, Shanghai, China, 2011.
- [126] W. Forstner and K. Khoshelham, "Efficient and accurate registration of point clouds with plane to plane correspondences," in *Proceedings of the IEEE International*

- Conference on Computer Vision Workshops*, pp. 2165–2173, Pasadena, CA, USA, 2017.
- [127] A. Segal, D. Haehnel, and S. Thrun, “Generalized-icp,” in *Robotics: science and systems* vol. volume 2p. 435, Seattle, WA, USA, 2009.
- [128] Q. Y. Zhou, J. Park, and V. Koltun, “Open3d: a modern library for 3d data processing,” 2018, <http://arxiv.org/abs/1801.09847>.
- [129] B. Radu and S. Cousins, “3d is here: point cloud library (pcl),” in *Proceedings of the 2011 IEEE International Conference On Robotics And Automation*, IEEE, Shanghai, China, 2011.
- [130] F. Pomerleau, S. Magnenat, F. Colas, M. Liu, and S. Roland, “Tracking a depth camera: parameter exploration for fast icp,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3824–3829, IEEE Press, Shanghai, China, 2011.
- [131] J. Yang, H. Li, and Y. Jia, “Go-icp: solving 3d registration efficiently and globally optimally,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1457–1464, Shanghai, China, 2013.
- [132] C. Yuan, X. Yu, and Z. Luo, “3d point cloud matching based on principal component analysis and iterative closest point algorithm,” in *Proceedings of the 2016 International Conference on Audio, Language and Image Processing (ICALIP)*, pp. 404–408, IEEE, Shanghai, China, 2016.
- [133] D. Avidar, D. Malah, and M. Barzohar, “Local-to-global point cloud registration using a dictionary of viewpoint descriptors,” in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 891–899, IEEE, Shanghai, China, 2017.
- [134] K.-H. Bae, *Automated registration of unorganised point clouds from terrestrial laser scanners*, PhD Thesis, Curtin University, Perth, Australia, 2006.
- [135] A. E. Johnson and S. B. Kang, “Registration and integration of textured 3d data,” *Image and Vision Computing*, vol. 17, no. 2, pp. 135–147, 1999.
- [136] D. Akca, “Matching of 3d surfaces and their intensities,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 2, pp. 112–121, 2007.
- [137] B. Yang, Y. Zang, Z. Dong, and R. Huang, “An automated method to register airborne and terrestrial laser scanning point clouds,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 109, pp. 62–76, 2015.
- [138] L. A. Alexandre, “3d descriptors for object and category recognition: a comparative evaluation,” in *Proceedings of the Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, Vilamoura, Portugal, 2012.
- [139] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, “A comprehensive performance evaluation of 3d local feature descriptors,” *International Journal of Computer Vision*, vol. 116, no. 1, pp. 66–89, 2016.
- [140] X.-F. Hana, S. Jesse, J. Xie, M.-J. Wang, and W. Jiang, “A comprehensive review of 3d point cloud descriptors,” 2018, <http://arxiv.org/abs/1802.02297>.
- [141] Y. Ran and X. Xu, “Point cloud registration method based on sift and geometry feature,” *Optik*, vol. 203, Article ID 163902, 2020.
- [142] X.-F. Han, S.-J. Sun, X.-Y. Song, and G.-Q. Xiao, “3d point cloud descriptors in hand-crafted and deep learning age: state-of-the-art,” 2018, <http://arxiv.org/abs/1404.3978>.
- [143] I. Hadji and G. N. De, “Local-to-global signature descriptor for 3d object recognition,” in *Proceedings of the Asian Conference on Computer Vision*, pp. 570–584, Springer, 2014.
- [144] B. Radu, N. Blodow, and M. Beetz, “Learning informative point classes for the acquisition of object model maps,” in *Proceedings of the 2008 10th International Conference on Control, Automation, Robotics and Vision*, pp. 643–650, IEEE, Hanoi, China, 2008.
- [145] F. Tombari, S. Salti, and L. Di Stefano, “Unique signatures of histograms for local surface description,” in *Proceedings of the European Conference On Computer Vision*, pp. 356–369, Springer, Crete, Greece, 2010.
- [146] R. Raguram, J.-M. Frahm, and M. Pollefeys, “A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus,” *Lecture Notes in Computer Science*, Springer, in *Proceedings of the European Conference on Computer Vision*, pp. 500–513, 2008.
- [147] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, “Robust point matching via vector field consensus,” *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1706–1721, 2014.
- [148] M. A. Fischler and R. C. Bolles, “Random sample consensus,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [149] S. Choi, T. Kim, and W. Yu, “Performance evaluation of ransac family,” *Journal of Computer Vision*, vol. 24, no. 3, pp. 271–300, 1997.
- [150] J. Ma, J. Zhao, J. Jiang, H. Zhou, and X. Guo, “Locality preserving matching,” *International Journal of Computer Vision*, vol. 127, no. 5, pp. 512–531, 2019.
- [151] J. Ma, Ji Zhao, J. Tian, Z. Tu, and L. Alan, “Robust estimation of nonrigid transformation for point set registration,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2147–2154, Portland, OR, USA, 2013.
- [152] J. Ma, W. Qiu, J. Zhao, Y. Ma, A. L. Yuille, and Z. Tu, “Robust $L_{2/1}$ estimation of transformation for non-rigid registration,” *IEEE Transactions on Signal Processing*, vol. 63, no. 5, pp. 1115–1129, 2015.
- [153] D. W. Scott, “Parametric statistical modeling by minimum integrated square error,” *Technometrics*, vol. 43, no. 3, pp. 274–285, 2001.
- [154] J. Ma, J. Wu, Ji Zhao, J. Jiang, H. Zhou, and Z. Quan, “Nonrigid point set registration with robust transformation learning under manifold regularization,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 12, pp. 3584–3597, 2018.
- [155] J. Ma, Ji Zhao, and L. Alan, “Non-rigid point set registration by preserving global and local structures,” *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 53–64, 2015.
- [156] C. R. Qi, H. Su, K. Mo, and J. Leonidas, “Pointnet: deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660, Honolulu, HI, USA, 2017.
- [157] Yi Fang, J. Xie, G. Dai et al., “3d deep shape descriptor,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, June 2015.
- [158] C. R. Qi, Li Yi, H. Su, and J. Leonidas, “Pointnet++: deep hierarchical feature learning on point sets in a metric space,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 5099–5108, Long Beach, CA, USA, 2017.
- [159] V. Sarode, X. Li, H. Goforth et al., “Pcnet: point cloud registration network using pointnet encoding,” 2019, <http://arxiv.org/abs/1908.07906>.

- [160] Y. Aoki, H. Goforth, R. Arun Srivatsan, and S. Lucey, "Pointnetlk: robust and efficient point cloud registration using pointnet," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, June 2019.
- [161] Z. Wu, S. Song, A. Khosla et al., "3d shapenets: a deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912–1920, Portland, OR, USA, 2015.
- [162] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.
- [163] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.
- [164] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: learning local geometric descriptors from rgb-d reconstructions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1802–1811, Las Vegas, NV, USA, 2017.
- [165] Y. Wang and J. M. Solomon, "Deep closest point: learning representations for point cloud registration," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3523–3532, Las Vegas, NV, USA, 2019.
- [166] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics*, vol. 38, no. 5, pp. 1–12, 2019.
- [167] C. Tang and P. Tan, "Ba-net: dense bundle adjustment network," 2018, <http://arxiv.org/abs/1806.04807>.
- [168] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," 2016, <http://arxiv.org/abs/1611.01144>.
- [169] Z. Yew and G. H. Lee, "Rpm-net: robust point matching using learned features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11824–11833, Las Vegas, NV, USA, 2020.
- [170] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness, "New algorithms for 2D and 3D point matching," *Pattern Recognition*, vol. 31, no. 8, pp. 1019–1031, 1998.
- [171] R. Sinkhorn, "A relationship between arbitrary positive matrices and doubly stochastic matrices," *The Annals of Mathematical Statistics*, vol. 35, no. 2, pp. 876–879, 1964.
- [172] E. Aarts and K. Jan, *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, Hoboken, NJ, USA, 1988.
- [173] Y. Aoki, H. Goforth, R. Arun Srivatsan, and S. Lucey, "Pointnetlk: robust & efficient point cloud registration using pointnet," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7163–7172, Seattle, WA, USA, 2019.
- [174] D. Bruce and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, Vancouver, Canada, 1981.
- [175] G. Dias Pais, S. Ramalingam, J. C. Nascimento, R. Chellappa, and P. Miraldo, "3dregnet: a deep neural network for 3d point registration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7193–7203, Seattle, WA, USA, 2020.
- [176] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.
- [177] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5556–5565, Las Vegas, NV, USA, 2015.
- [178] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 487–495, Montreal, Canada, 2014.
- [179] B. Jian and B. C. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1633–1645, 2010.
- [180] D. Campbell and L. Petersson, "Gogma: globally-optimal Gaussian mixture alignment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5685–5694, Las Vegas, NV, USA, 2016.
- [181] B. Eckart, K. Kim, and J. Kautz, "Fast and accurate point cloud registration using trees of gaussian mixtures," 2018, <http://arxiv.org/abs/1807.02587>.
- [182] H. Xiong, S. Szedmak, and J. Piater, "A study of point cloud registration with probability product kernel functions," in *Proceedings of the 2013 International Conference on 3D Vision-3DV 2013*, pp. 207–214, IEEE, Santiago, Chile, 2013.
- [183] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.
- [184] D. Campbell and L. Petersson, "An adaptive data representation for robust point-set registration and merging," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, December 2015.
- [185] A. Myronenko, X. Song, and M. A. Carreira-Perpinán, "Non-rigid point set registration: coherent point drift," *Advances In Neural Information Processing Systems*, 2007.
- [186] A. Myronenko and X. Xubo Song, "Point set registration: coherent point drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.
- [187] M. Lu, J. Zhao, Y. Guo, and Y. Ma, "Accelerated coherent point drift for automatic three-dimensional point cloud registration," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 2, pp. 162–166, 2015.
- [188] R. Varadhan and C. Roland, "Simple and globally convergent methods for accelerating the convergence of any em algorithm," *Scandinavian Journal of Statistics*, vol. 35, no. 2, pp. 335–353, 2008.
- [189] P. Wang, P. Wang, Z. Qu, Y. Gao, and Z. Shen, "A refined coherent point drift (cpd) algorithm for point set registration," *Science China Information Sciences*, vol. 54, no. 12, pp. 2639–2646, 2011.
- [190] J. Lu, W. Wang, Z. Fan, S. Bi, and C. Guo, "Point cloud registration based on cpd algorithm," in *Proceedings of the 2018 37th Chinese Control Conference (CCC)*, pp. 8235–8240, IEEE, Wuhan, China, 2018.
- [191] P. Biber and W. Straßer, "The normal distributions transform: a new approach to laser scan matching," in *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, vol. 3, pp. 2743–2748, IEEE, Wuhan, China, 2003.
- [192] M. Martin, *The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection*, Örebro universitet, PhD thesis, 2009.

- [193] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3d-ndt," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.
- [194] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, New Delhi, India, 1996.
- [195] H. Hong, "Probabilistic normal distributions transform representation for accurate 3d point cloud registration," in *Proceedings of the 2017 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3333–3338, IEEE, 2017.
- [196] J. Stückler and S. Behnke, "Multi-resolution surfel maps for efficient dense 3d modeling and tracking," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 137–147, 2014.
- [197] M. Magnusson, A. Nuchter, C. Lorken, J. Achim, and J. Hertzberg, "Evaluation of 3d registration reliability and speed—a comparison of icp and ndt," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, pp. 3907–3912, IEEE, 2009.
- [198] X. Shi, J. Peng, J. Li, P. Yan, and H. Gong, "The iterative closest point registration algorithm based on the normal distribution transformation," *Procedia Computer Science*, vol. 147, pp. 181–190, 2019.
- [199] M. Attia, Y. Slama, P. Laurent, H. Cao, and F. Haddad, "3d point cloud coarse registration based on convex hull refined by icp and ndt," in *Proceedings of the 2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, 2018.
- [200] S. Xue, G. Li, Q. Lv, X. Meng, and X. Tu, "Point cloud registration method for pipeline workpieces based on ndt and improved icp algorithms," *IOP Conference Series: Materials Science and Engineering*, vol. 677, Article ID 022131, 2019.
- [201] Y. Liu, D. Kong, D. Zhao, X. Gong, and G. Han, "A point cloud registration algorithm based on feature extraction and matching," *Mathematical Problems in Engineering*, vol. 2018, Article ID 7352691, 9 pages, 2018.
- [202] C. Ulas and H. Temeltas, "A 3d scan matching method based on multi-layered normal distribution transform," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 11602–11607, 2011.
- [203] H. Hong and B. H. Lee, "Key-layered normal distributions transform for point cloud registration," *Electronics Letters*, vol. 51, no. 24, pp. 1986–1988, 2015.
- [204] J. Lu, W. Liu, D. Dong, and Q. Shao, "Point cloud registration algorithm based on ndt with variable size voxel," in *Proceedings of the 2015 34th Chinese Control Conference (CCC)*, pp. 3707–3712, IEEE, Hangzhou, China, 2015.
- [205] A. Das and S. L. Waslander, "Scan registration using segmented region growing ndt," *The International Journal of Robotics Research*, vol. 33, no. 13, pp. 1645–1663, 2014.
- [206] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," *ACM*, 2008.
- [207] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1584–1601, 2006.
- [208] G. Turk and M. Levoy, "Zippered polygon meshes from range images," in *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, pp. 311–318, Orlando, FL, USA, 1994.
- [209] M. Mohamad, D. Rappaport, and M. Greenspan, "Generalized 4-points congruent sets for 3d registration," in *Proceedings of the 2014 2nd international conference on 3D vision*, vol. 1, pp. 83–90, IEEE, Orlando, FL, USA, 2014.
- [210] M. Mohamad, M. T. Ahmed, D. Rappaport, and M. Greenspan, "Super generalized 4pcs for 3d registration," in *Proceedings of the 2015 International Conference on 3D Vision*, pp. 598–606, IEEE, Orlando, FL, USA, 2015.
- [211] J. Huang, T.-Ho Kwok, and C. Zhou, "V4pcs: volumetric 4pcs algorithm for global registration," *Journal of Mechanical Design*, vol. 139, no. 11, 2017.
- [212] N. Mellado and D. Aiger, "Super 4pcs fast global pointcloud registration via smart indexing," *Computer Graphics Forum*, vol. 33, no. 5, pp. 205–215, 2014.
- [213] P. W. Theiler, J. D. Wegner, and K. Schindler, "Keypoint-based 4-points congruent sets—automated marker-less registration of laser scans," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 96, pp. 149–163, 2014.
- [214] X. Ge, "Automatic markerless registration of point clouds with semantic-keypoint-based 4-points congruent sets," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 130, pp. 344–357, 2017.
- [215] S. Saeedi, M. Trentini, M. Seto, and H. Li, "Multiple-robot simultaneous localization and mapping: a review," *Journal of Field Robotics*, vol. 33, no. 1, pp. 3–46, 2016.
- [216] J. Čerňava, M. Mokroš, J. Tuček, M. Antal, and Z. Slatkovská, "Processing chain for estimation of tree diameter from gnss-imu-based mobile laser scanning data," *Remote Sensing*, vol. 11, no. 6, p. 615, 2019.
- [217] P. Rönholm, X. Liang, A. Kukko, A. Jaakkola, and J. Hyyppä, "Quality analysis and correction of mobile backpack laser scanning data," *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 3, no. 1, 2016.
- [218] M. Pierzchała, P. Giguère, and R. Astrup, "Mapping forests using an unmanned ground vehicle with 3d lidar and graphslam," *Computers and Electronics in Agriculture*, vol. 145, pp. 217–225, 2018.
- [219] S. Thrun and M. Montemerlo, "The graph slam algorithm with applications to large-scale mapping of urban structures," *The International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 403–429, 2006.
- [220] B. O. Abayowa, A. Yilmaz, and R. C. Hardie, "Automatic registration of optical aerial imagery to a lidar point cloud for generation of city models," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 106, pp. 68–81, 2015.
- [221] X. Ge, H. Hu, and B. Wu, "Image-guided registration of unordered terrestrial laser scanning point clouds for urban scenes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 9264–9276, 2019.
- [222] E. Ferrante and N. Paragios, "Slice-to-volume medical image registration: a survey," *Medical Image Analysis*, vol. 39, pp. 101–123, 2017.
- [223] H. Grant, U. Kruger, and P. Yan, "Deep learning in medical image registration: a survey," *Machine Vision and Applications*, vol. 31, no. 1, pp. 1–18, 2020.
- [224] C. Hubert, "Active graph based semi-supervised learning using image matching: application to handwritten digit recognition," *Pattern Recognition Letters*, vol. 73, pp. 76–82, 2016.
- [225] R. Szeliski, "Image alignment and stitching: a tutorial," *Foundations and Trends® in Computer Graphics and Vision*, vol. 2, no. 1, pp. 1–104, 2006.
- [226] I. Rocco, R. Arandjelović, and J. Sivic, "End-to-end weakly-supervised semantic alignment," in *Proceedings of the IEEE*

- Conference on Computer Vision and Pattern Recognition*, pp. 6917–6925, Salt Lake City, UT, USA, 2018.
- [227] W. Liu, J. Sun, W. Li, T. Hu, and P. Wang, “Deep learning on point clouds and its application: a survey,” *Sensors*, vol. 19, no. 19, p. 4188, 2019.
- [228] Y. Pan, B. Yang, F. Liang, and Z. Dong, “Iterative global similarity points: a robust coarse-to-fine integration solution for pairwise 3d point cloud registration,” in *Proceedings of the 2018 International Conference on 3D Vision (3DV)*, pp. 180–189, IEEE, Verona, Italy, 2018.
- [229] X. Huang, J. Zhang, Q. Wu, L. Fan, and C. Yuan, “A coarse-to-fine algorithm for matching and registration in 3d cross-source point clouds,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2965–2977, 2017.
- [230] M. Attia, Y. Slama, and M. Amine Kamoun, “On performance evaluation of registration algorithms for 3d point clouds,” in *Proceedings of the 2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV)*, pp. 45–50, IEEE, Beni Mellal, Morocco, 2016.
- [231] E. Lachat, T. Landes, and P. Grussenmeyer, “Comparison of point cloud registration algorithms for better result assessment—towards an open-source solution,” *Towards Photogrammetry*, vol. 42, pp. 551–558, 2020.
- [232] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, “Challenging data sets for point cloud registration algorithms,” *The International Journal of Robotics Research*, vol. 31, no. 14, pp. 1705–1711, 2012.
- [233] R. Hänsch, T. Weber, and O. Hellwich, “Comparison of 3d interest point detectors and descriptors for point cloud fusion,” *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3, no. 3, pp. 57–64, 2014.