



<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper published in *IEEE Transaction on Pattern Analysis and Machine Intelligence*. This paper has been peer-reviewed but does not include the final publisher proof-corrections or journal pagination.

Citation for the original published paper (version of record):

Liao, Q., Sun, D., Andreasson, H. [Year unknown!]

Point Set Registration for 3D Range Scans Using Fuzzy Cluster-based Metric and Efficient Global Optimization

IEEE Transaction on Pattern Analysis and Machine Intelligence

<https://doi.org/10.1109/TPAMI.2020.2978477>

Access to the published version may require subscription.

N.B. When citing this work, cite the original published paper.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:oru:diva-80214>

Point Set Registration for 3D Range Scans Using Fuzzy Cluster-based Metric and Efficient Global Optimization

Qianfang Liao, Da Sun, and Henrik Andreasson

Abstract—This study presents a new point set registration method to align 3D range scans. In our method, fuzzy clusters are utilized to represent a scan, and the registration of two given scans is realized by minimizing a fuzzy weighted sum of the distances between their fuzzy cluster centers. This fuzzy cluster-based metric has a broad basin of convergence and is robust to noise. Moreover, this metric provides analytic gradients, allowing standard gradient-based algorithms to be applied for optimization. Based on this metric, the outlier issues are addressed. In addition, for the first time in rigid point set registration, a registration quality assessment in the absence of ground truth is provided. Furthermore, given specified rotation and translation spaces, we derive the upper and lower bounds of the fuzzy cluster-based metric and develop a branch-and-bound (BnB)-based optimization scheme, which can globally minimize the metric regardless of the initialization. This optimization scheme is performed in an efficient coarse-to-fine fashion: First, fuzzy clustering is applied to describe each of the two given scans by a small number of fuzzy clusters. Then, a global search, which integrates BnB and gradient-based algorithms, is implemented to achieve a coarse alignment for the two scans. During the global search, the registration quality assessment offers a beneficial stop criterion to detect whether a good result is obtained. Afterwards, a relatively large number of points of the two scans are directly taken as the fuzzy cluster centers, and then, the coarse solution is refined to be an exact alignment using the gradient-based local convergence. Compared to existing counterparts, this optimization scheme makes a large improvement in terms of robustness and efficiency by virtue of the fuzzy cluster-based metric and the registration quality assessment. In the experiments, the registration results of several 3D range scan pairs demonstrate the accuracy and effectiveness of the proposed method, as well as its superiority to state-of-the-art registration approaches.

Index Terms—point set registration, fuzzy clusters, registration quality assessment, 3D range scans, branch-and-bound.

1 INTRODUCTION

1.1 Background

POINT set registration, also known as point cloud matching or scan registration, defines a task of finding the spatial transformation to correctly align two sets of points. It is an important technology in various applications, including computer vision, pattern recognition, medical image processing, and robotics. Point set registration can be utilized to build a coherent 3D model from a group of partial scans [1], to recognize text for optical readers [2], to fuse magnetic resonance or computed tomography scans for medical analysis [3], [4], and to realize simultaneous localization and mapping for robots [5], [6]. In this study, we focus on rigid point set registration of range scans, where the spatial transformation consists of 3D rotation and 3D translation.

Several challenges exist in the development of an effective registration method. The primary challenge is the selection of a metric to measure the similarity between two point sets. A proper metric is robust to noise and easily optimized. More importantly, it provides accurate alignments. Different types of metrics have been proposed. The most widely applied metric is the iterative closest point (ICP) metric [7], which minimizes the nearest point-to-point

Euclidean distances to realize registration. Various modifications and extensions of this metric have been developed [2], [8]–[10]. Some methods preprocess point sets via different techniques and take the properties of the processed sets as the similarity metric, such as probability-based methods [11]–[16], which describe point sets by probability distributions and take the probability-based errors as the metric, and neural network-based methods [17]–[20], which utilize learning skills to encode point clouds and match them based on the encoded information. Other existing metrics include geometric regularity in the frequency domain [21] and mutual information [4], [22]. The algorithm used to optimize the metric is another key factor for registration. The algorithm should be computationally efficient, be capable of treating the nonlinearity of the metric, and produce optimal solutions instead of being trapped in local optima. So far, diverse optimization algorithms can be found. For local convergence, there are expectation-maximization (EM) [13], [23], gradient-based methods [2], [10], [12], [14], [15], etc. For global search, there are genetic algorithms [24], particle filtering [25], simulated annealing [26], branch-and-bound (BnB) [27]–[29], etc. Moreover, several schemes are available to enhance computational efficiency and improve registration quality, such as the kd-tree and distance transform [2], [29], which speed up the nearest neighbor search for ICP; the trimming strategies [30], which reduce the negative effects of outliers; and the techniques of local feature correspondence construction for accurate alignments [31]–[34].

- The authors are with the Centre for Applied Autonomous Sensor Systems (AASS), Örebro University, Sweden (Corresponding author: Da Sun).
E-mail: Qianfang.Liao@oru.se; Da.Sun@oru.se;
Henrik.Andreasson@oru.se

One important aspect of point set registration is registration quality assessment without knowing the ground truth. However, to the best of our knowledge, basically no study in this area has been proposed. In practice, a ground truth transformation against which to compare the registration solution is often unavailable. Consequently, it is difficult for the existing methods to automatically assess whether a solution derived in the optimization process is good or not. As a result, the methods may need user intervention, such as requiring the user to constantly check the accuracy of the solution during optimization, or they may have to spend extra time and computational cost in optimization, such as choosing a large number of iterations in the calculation to guarantee that the final solution is correct. These methods are inconvenient and lead to loss of efficiency. Therefore, it is necessary to develop a registration quality assessment in the absence of ground truth.

1.2 Related Work

There are a large number of published studies about point set registration, and we only review some that are relevant to ours. Comprehensive surveys can be found in [35], [36].

ICP [7] is the most well-known registration method. Starting with an initial transformation, it employs a direct and intuitive method that iteratively minimizes the nearest points' distances between the two point sets until convergence. The extensions of ICP include revised metrics, such as the approach considering normals [37], the generalized expression [8], and the L_p error of sparse ICP for treating noise [9]; different convergence schemes, such as the EM algorithm [23] and the Levenberg-Marquardt algorithm (LM-ICP) [2]; and soft assignment of correspondence [10], [38] and outlier trimming [30] for greater robustness and higher accuracy.

The traditional ICP may converge to a local minimum and yield a misalignment if it does not have a good initialization. To address this issue, a number of global optimization methods have been developed, including stochastic techniques [24], [39] and deterministic BnB schemes [27]–[29]. Compared to stochastic methods that do not guarantee that the optimal solutions will be found, deterministic methods, which can always locate the global optima, are more attractive. Among the existing deterministic BnB schemes, Go-ICP [29] is outstanding: It reveals the upper and lower bounds of the traditional ICP metric in $SE(3)$ and combines BnB with the traditional ICP to give a globally optimal solution up to the expected accuracy. Moreover, Go-ICP builds a 3D distance transform to speed up the nearest correspondence search. In addition, unlike the BnBs in [27], [28], which either need the two point sets to be equal size or require a known correspondence, Go-ICP can be implemented without any unrealistic assumptions.

Another type of registration method has a preprocessing step that represents the point sets using different models, and then, the model-based metric is optimized to realize registration. Gaussian mixture models (GMMs) and the normal distributions transform (NDT) are popular models of this type. Myronenko and Song propose the coherent point drift [13], which takes one point set as the GMM centroids and the other point set as the elements and then aligns the two

sets in a point-to-distribution matching way using EM. This method requires the user to set the ratio of noisy outliers and may incur a high computational cost. Jian and Vemuri [12] describe both point sets by GMMs and align the two sets in a distribution-to-distribution matching way by minimizing the L_2 distances between the two GMMs. In [1], a groupwise method called JRMPC is developed to simultaneously register multiple point sets based on GMM. Magnusson et al. [14] preprocess one point set using NDT and take the other set as the elements, and they then apply the point-to-distribution matching method to realize registration using Newton's algorithm. In [15], Stoyanov et al. combine NDT and the distribution-to-distribution matching method of [12]: Both point sets are preprocessed by NDT, and the registration is achieved by minimizing the L_2 distance between the two normal distribution groups. Compared to ICPs, these methods yield a piecewise smooth cluster representation for one or both of the point sets, which can alleviate the impact of noise, increase the robustness to poor initialization, and largely reduce the memory storage and computational cost because a few parameters instead of a large number of points are included in the optimization. In addition, these cluster representations have analytic expressions of derivatives such that standard gradient-based algorithms can be applied to optimize the metric.

Apart from the probability-based clusters, neural networks have recently been utilized to preprocess the point sets. Aoki et al. [17] describe the point sets via PointNet, which is a learnable structured representation, and apply a modified L&K algorithm for optimization. Gojic et al. [18] propose 3DSmoothNet, which is a deep learning method with fully convolutional layers to match point clouds. In [19], 3DFeat-Net is used to learn both a feature detector and descriptor for point set registration under weak supervision. In [20], Elbaz et al. employ a deep neural network to encode local 3D geometric structures and construct the correspondence between the two point sets for a coarse alignment; they then apply ICP for fine-tuning. The main problems of these methods are that they require a lengthy training phase, need to manually select a relatively large number of parameters for the training, and cannot guarantee the generalization of the trained neural network to match any type of point set.

A common issue facing all the methods stated above is that they lack a registration quality assessment to automatically evaluate whether a solution is good during their optimization when no ground truth is available.

1.3 Proposed Work and the Innovations

In this paper, we propose a new point set registration method. The innovations are as follows:

i). Fuzzy clusters are leveraged to describe a point set, and the registration of two given point sets is achieved by minimizing a fuzzy weighted sum of the distances between their fuzzy cluster centers. This fuzzy cluster-based metric is developed based on the fuzzy c-means algorithm [40]. It employs a fuzzy weighted one-to-all correspondence, which offers a broad basin of convergence and a precise alignment. Compared to ICPs, this metric does not search for the nearest correspondence and has the advantages of cluster representations stated above. Compared to the existing

probability-based clusters, fuzzy clusters are more robust to noise. Unlike the neural network-based approaches, our method neither requires any training phase nor suffers from the generalization issue.

ii). With the fuzzy cluster-based metric, simple and effective outlier handling strategies are developed for better alignment. To treat noisy outliers, we design a noniterative pruning method using the robust fuzzy c-means method proposed in [41]. To address the nonoverlapping part of partially overlapping scans, we propose a trimming approach inspired by the trimmed ICP in [30].

iii). Two indexes, which describe the points' disposition and dispersion with respect to fuzzy cluster centers, are proposed to assess registration quality without requiring knowing the ground truth. To the best of our knowledge, this is the first work involving registration quality assessment in the absence of ground truth for rigid point set registration. This assessment can greatly improve the computational efficiency of global optimization and is one of our key contributions.

iv). Given specified rotation and translation spaces, we derive the upper and lower bounds of the fuzzy cluster-based metric and develop a BnB-based optimization scheme, which can locate the global minimum of the metric regardless of the initialization. This scheme is performed in an efficient coarse-to-fine fashion. First, fuzzy clustering is applied to describe each of the point sets using a small number of fuzzy clusters. Then, the global search, which integrates the BnB and gradient-based algorithms, is implemented to coarsely align the two scans. During the global search, the registration quality assessment offers a beneficial stop criterion that keeps detecting if a good solution is achieved. Afterwards, a relatively large number of points of each scan are directly taken as the fuzzy cluster centers, and then, the gradient-based local convergence is executed to refine the coarse alignment and attain an exact alignment. This optimization scheme is related to Go-ICP [29] but has the following advantages: Our method does not need to build a distance transform and possesses increased robustness to noise. Moreover, our method requires fewer steps to converge owing to the broad convergence basin of the fuzzy cluster-based metric. Furthermore, by means of the registration quality assessment, our method is more cost-effective.

2 PROBLEM STATEMENT

Given two 3D scans/point sets that have overlaps, where one is the fixed set $\mathbb{P}_F = \{\mathbf{p}_{Fi}, i = 1, \dots, N_{P_F}\}$ and the other is the moving set $\mathbb{P}_M = \{\mathbf{p}_{Mj}, j = 1, \dots, N_{P_M}\}$, where $\mathbf{p}_{Fi}, \mathbf{p}_{Mj} \in \mathbb{R}^3$ are point coordinates, the registration must find a parameter vector λ of a transformation function $\mathcal{T}(\lambda)$ such that the transformed moving set, denoted by $\mathcal{T}(\lambda, \mathbb{P}_M) = \{\mathcal{T}(\lambda, \mathbf{p}_{Mj}), j = 1, \dots, N_{P_M}\}$, can be correctly aligned with \mathbb{P}_F . We define $\lambda = (\mathbf{r}; \mathbf{t}) \in \mathbb{R}^6$, where $\mathbf{r}, \mathbf{t} \in \mathbb{R}^3$ are rotation and translation parameters in 3D space, respectively. $\mathcal{T}(\lambda, \mathbf{p}_{Mj})$ can be expressed as

$$\mathcal{T}(\lambda, \mathbf{p}_{Mj}) = \mathcal{R}(\mathbf{r}) \cdot \mathbf{p}_{Mj} + \mathbf{t} \quad (1)$$

where $\mathcal{R}(\mathbf{r}) \in SO(3)$ is the rotation matrix. We express \mathbf{r} by the axis-angle representation, where the axis is $\mathbf{r}/\|\mathbf{r}\|$ and

the angle is $\|\mathbf{r}\|$. The registration results are prone to being less accurate if the overlapping regions occupy a small part of the point sets. In practice, a 50% overlap is critical [42]. In this study, we apply the following assumption:

Assumption 1. The overlapping regions of the two point sets for registration are at least 50%.

When designing a metric for registration, a proper correspondence is an important factor to obtain accurate transformations. Standard ICPs [7], [29] and the NDT in [15] utilize binary nearest-point/-component correspondence. This one-to-one fashion makes the metric simple and direct but generally leads to a narrow convergence basin and causes the algorithm to be easily trapped in local optima. GMMs in [1], [12], [13] employ one-to-multiple correspondence with equal weights. Compared to the one-to-one fashion, this one-to-multiple correspondence has a broader convergence range and is more robust to noise since it considers neighborhood information. However, the equal weights may not correctly reflect the neighborhood relationship. Some *softassign* methods for ICP using one-to-multiple correspondence with varying weights can be found in [10], [38]. The robust point matching method in [10] uses Gaussian weights in the metric and applies an annealing process to the variance. This method can be time-consuming, especially for aligning high-resolution point sets. In [38], a linear distance-based weighting and a normals' compatibility-based weighting are utilized. However, the linear weighting may degrade the registration performance in some cases [14], and the normals of a scan are not always available and are generally sensitive to noise.

Regarding optimization, many existing registration methods, such as the ICPs in [2], [7], [23], [37], the GMMs in [1], [12], [13] and the NDTs in [14], [15], are only applicable for local convergence and will fall into local optima without a good initialization. Global optimization, which can address the local optima issue, is more appealing. Go-ICP [29] is the first deterministic global optimization for rigid point set registration of 3D ICP. However, some problems exist in Go-ICP: Its 3D distance transform construction is relatively time-consuming, and it may take many steps to converge in the global search due to the narrow convergence basin of the ICP's metric. Moreover, Go-ICP inherits the ICP's sensitivity to uncertainties and may fail when noisy outliers appear to cause false correspondence.

Another problem in Go-ICP, and other global optimization schemes as well, is that it is difficult to automatically assess the registration quality during optimization without the ground truth alignment against which to compare. Generally, the global search stops and produces the final solution when reaching a predefined criterion, such as the maximum number of iterations, the minimum size of the remaining search space, or a certain threshold. However, proper values for these stop criteria are not easy to choose. With a loose choice, the final solution may be erroneous. With a strict choice, the algorithm has to run for a relatively long extra period of time to meet the stop criterion after the optimal solution is found. If a registration quality assessment is available, those stop criteria can choose arbitrarily strict values, and then, the global search will automatically detect when a good solution is derived and stop in time to save the unnecessary costs. Therefore, developing a good registration

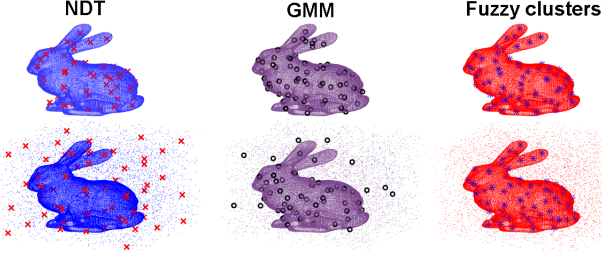


Fig. 1. The means/centers of NDT, GMM and fuzzy clusters (denoted by “x”s, “o”s and “*”s, respectively) for the full bunny model with and without noisy outliers. This model has 35947 points, and the number of components of each representation is 64. The number of noisy outliers is 20% that of the original points, and they are uniformly distributed in a cuboid centered by the original model and with length, width and height 1.5 times those of the original bounding box.

quality assessment in the absence of ground truth can be an important contribution for point set registration.

3 FUZZY CLUSTERING FOR POINT SETS

A fuzzy cluster is an extension of the classical notion of set. Unlike the binary membership of the classical set, the membership grade of an element in a fuzzy cluster is between 0 and 1, and one element can belong to multiple fuzzy clusters with different membership grades. The “soft” fuzzy membership grade provides a more natural and reasonable method to characterize elements and is more robust to uncertainties [43]–[46]. In this study, we utilize fuzzy clusters to represent each of the 3D point sets for registration.

Given a 3D point set $\mathbb{P} = \{\mathbf{p}_j, j = 1, \dots, N_P\}$, we suppose that it is described by N_C fuzzy clusters, whose centers are denoted by $\mathbb{C} = \{\mathbf{c}_i \in \mathbb{R}^3, i = 1, \dots, N_C\}$, and we employ the popular fuzzy c-means algorithm [40] to perform the fuzzy clustering. Fuzzy c-means determines \mathbb{C} by iteratively minimizing the following objective function [40]:

$$\begin{aligned} \min_{\mathbb{C}} J(\mathbb{P}, \mathbb{C}) &= \min_{\mathbb{C}} \sum_{j=1}^{N_P} J(\mathbf{p}_j, \mathbb{C}) \\ &= \min_{\mathbb{C}} \sum_{j=1}^{N_P} \sum_{i=1}^{N_C} \mu_{c_i}(\mathbf{p}_j)^m \cdot D(\mathbf{p}_j, \mathbf{c}_i)^2 \end{aligned} \quad (2)$$

where $D(\mathbf{p}_j, \mathbf{c}_i) = \|\mathbf{p}_j - \mathbf{c}_i\|$; $J(\mathbf{p}_j, \mathbb{C})$ is the distance loss of \mathbf{p}_j in $J(\mathbb{P}, \mathbb{C})$; $m > 1$ determines the level of the clusters’ fuzziness: The partition is fuzzier with a greater m , and $m = 2$ is a widely accepted choice; and $\mu_{c_i}(\mathbf{p}_j)$ is the fuzzy membership grade of \mathbf{p}_j in the fuzzy cluster centered at \mathbf{c}_i , which satisfies $0 \leq \mu_{c_i}(\mathbf{p}_j) \leq 1$ and $\sum_{i=1}^{N_C} \mu_{c_i}(\mathbf{p}_j) = 1$, and is calculated as follows [40]:

$$\mu_{c_i}(\mathbf{p}_j) = \begin{cases} 1, & \text{if } D(\mathbf{p}_j, \mathbf{c}_i) = 0 \\ 0, & \text{if } \forall_{k \neq i} D(\mathbf{p}_j, \mathbf{c}_k) = 0 \\ \frac{1}{\sum_{k=1}^{N_C} \left(\frac{D(\mathbf{p}_j, \mathbf{c}_i)}{D(\mathbf{p}_j, \mathbf{c}_k)} \right)^{\frac{2}{m-1}}}, & \text{else} \end{cases} \quad (3)$$

The details of fuzzy c-means clustering can be found in [40].

Remark 1. N_C must not exceed N_P . When choosing $N_C = N_P$, each \mathbf{p}_j is a fuzzy cluster center. The selection of N_C for registration will be elaborated in the next section.

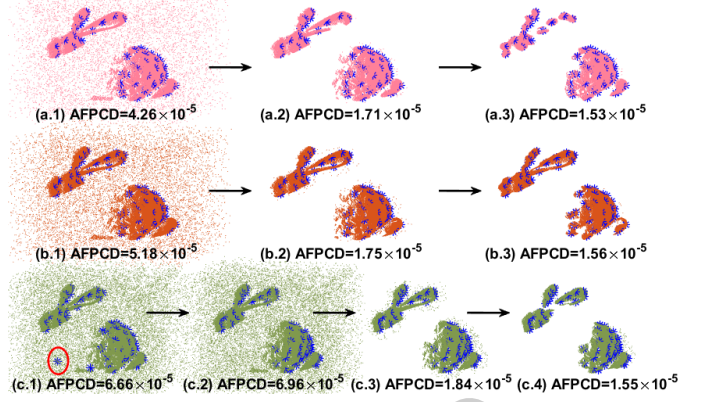


Fig. 2. Fuzzy clustering and noisy outlier pruning results for the partial bunny scan “bun090” with different degrees of noise. The first to third rows show the cases with the number of noisy outliers to be 20%, 30% and 50%, respectively, of that of the real points (the number of real points is 30379). The noisy outliers are distributed in the same way as in Fig. 1. For each case, $N_C = 60$, and each “*” denotes a fuzzy cluster center. (a.1), (b.1) and (c.1) show the clustering results using fuzzy c-means [40], and (c.2) shows the result using robust fuzzy c-means [41]. The rest shows the results of the pruning method introduced in Section 4.2, where (a.2), (b.2) and (c.3) are the pruning results of the first step and (a.3), (b.3) and (c.4) are the pruning results of the second step by removing 15% of the points with the largest distance losses. “AFPCD” is an index related to the registration quality assessment defined in Section 4.3.

Compared to the probability-based clusters, such as GMM and NDT, fuzzy clusters are more robust to noisy outliers. A comparison using the Stanford bunny model [47] is shown in Fig. 1. The noisy outliers significantly change the positions of the means of NDT and GMM, which is likely to result in misalignment (see Fig. 8 in Section 5.2). In contrast, the fuzzy cluster centers remain on or are very close to the scanned object’s surface. The reason for this fact is that in many scenarios, noisy outliers have lower densities than real points do, and fuzzy cluster centers will be placed at the “dense” area. Other examples using a partial Stanford bunny scan are shown in Figs. 2(a.1) and 2(b.1), where “AFPCD” is an index related to the registration quality assessment and will be introduced in Section 4.3.

The existence of noisy outliers may impair the registration performance. Moreover, some fuzzy cluster centers may be misplaced due to serious noise pollution, such as the example shown in Fig. 2(c.1), where one center (in the red ellipse) is placed in the noise area. For this issue, some filtering techniques can be applied to preprocess the noisy point sets, such as the structure-aware data consolidation [48] iteratively moving noisy outliers towards underlying, lower-dimensional structures. In this study, for simplicity, we use a different method to address this issue: The robust fuzzy c-means method in [41] is applied to postprocess the results of fuzzy c-means and drag the misplaced center(s) to the real point area; afterwards, a simple and noniterative pruning method is performed to remove noisy outliers. The result of robust fuzzy c-means for the case of Fig. 2(c.1) is shown in Fig. 2(c.2), where all the centers are placed in the real point area. Compared to the filtering method in [48], robust fuzzy c-means is a simpler implementation. In addition, a functioning depth camera is unlikely to give point

sets with serious noise pollution. Therefore, in most cases, fuzzy c-means can work with satisfactory performance, and the postprocessing step using robust fuzzy c-means is not necessary. In the next section, we present the details of the noniterative pruning method and introduce the rest of Fig. 2.

4 REGISTRATION BASED ON FUZZY CLUSTERS

This section presents the main contributions of our study. First, a fuzzy cluster-based metric for point set registration is introduced. Afterwards, we describe the strategies to filter out noisy outliers and trim the nonoverlapping part of partially overlapping scans. Next, two fuzzy cluster-based indexes are defined to provide a registration quality assessment. Finally, an efficient deterministic global optimization scheme for minimizing the metric is presented.

4.1 Fuzzy Cluster-based Metric

Given \mathbb{P}_F and \mathbb{P}_M , by respectively applying fuzzy clustering to them, we have N_{C_F} fuzzy cluster centers $\mathbb{C}_F = \{\mathbf{c}_{Fi}, i = 1, \dots, N_{C_F}\}$ of \mathbb{P}_F and N_{C_M} fuzzy cluster centers $\mathbb{C}_M = \{\mathbf{c}_{Mj}, j = 1, \dots, N_{C_M}\}$ of \mathbb{P}_M . When \mathbb{P}_F and \mathbb{P}_M are aligned by $\mathcal{T}(\lambda)$, there are two cases: i). One scan is fully overlapped by the other, as in the examples shown in Figs. 3(a) and 3(b); and ii). The two scans partially overlap each other, as in the example shown in Fig. 3(c). For both cases, in the overlapping region, the points as well as the fuzzy cluster centers of $\mathcal{T}(\lambda, \mathbb{P}_M)$ can be regarded as the elements of \mathbb{P}_F 's fuzzy clusters, since they describe the same spatial properties as \mathbf{p}_{Fi} 's do. We denote the fuzzy cluster centers of the overlapping part of \mathbb{P}_M as $\mathbb{C}'_M = \{\mathbf{c}'_{Mj}, j = 1, \dots, N_{C'_M}\}$. Note that when \mathbb{P}_M is fully overlapped by \mathbb{P}_F , \mathbb{C}'_M equals \mathbb{C}_M , and when \mathbb{P}_M is partially overlapped by \mathbb{P}_F , \mathbb{C}'_M is a subset of \mathbb{C}_M . Since $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ s can be regarded as the elements of \mathbb{P}_F 's fuzzy clusters, they can minimize the corresponding fuzzy c-means objective function with respect to \mathbb{C}_F . Therefore, we leverage the following objective function revised from Eq. (2) to be the metric of registration and minimize it to derive the transformation λ :

$$\begin{aligned} \min_{\lambda, \mathbb{C}'_M} J(\lambda, \mathbb{C}'_M, \mathbb{C}_F) &= \min_{\lambda, \mathbb{C}'_M} \sum_{j=1}^{N_{C'_M}} J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) \\ &= \min_{\lambda, \mathbb{C}'_M} \sum_{j=1}^{N_{C'_M}} \sum_{i=1}^{N_{C_F}} \mu_{\mathbf{c}_{Fi}}(\mathcal{T}(\lambda, \mathbf{c}'_{Mj}))^m \cdot D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})^2 \end{aligned} \quad (4)$$

where $D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) = \|\mathcal{T}(\lambda, \mathbf{c}'_{Mj}) - \mathbf{c}_{Fi}\|$, and $J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ is the distance loss of $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ in $J(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$. We will state how to pick out \mathbb{C}'_M from \mathbb{C}_M for partial overlap registration in Section 4.2.

The fuzzy cluster-based metric $J(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$ in Eq. (4) utilizes a one-to-all correspondence with dynamic fuzzy membership grades as the weights. Unlike the binary nearest one-to-one correspondence used by ICP and the NDT in [15], this one-to-all fashion has a broader convergence basin since it includes neighborhood information. Compared to the one-to-multiple fashion with equal and fixed weights used by the GMM-based methods [1], [12], [13], the dynamic

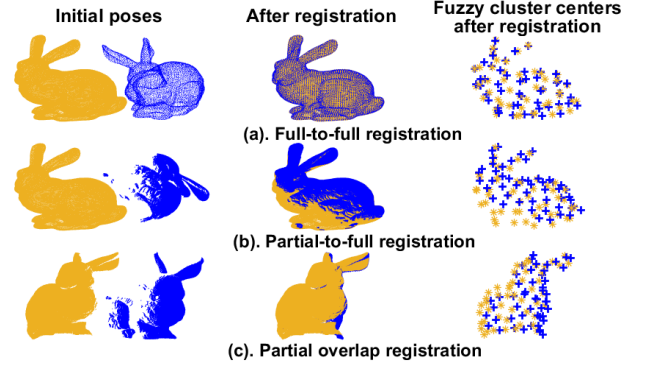


Fig. 3. Some examples of point set registration. The fixed sets and the moving sets are in yellow and blue, respectively. In the rightmost column, the yellow “*”s and the blue “+”s are their fuzzy cluster centers.

weights describe the neighborhood relationship more accurately. Moreover, our dynamic weights are more reasonable and attainable than those stated in [38].

Note that for any case of registration, we will first select the scan/point set with a larger area and then take its fuzzy cluster centers as \mathbb{C}_F to calculate the fuzzy membership grades in Eq. (4). We will state how to perform this selection in Section 4.3. This method can guarantee that more spatial information is included to calculate the neighborhood relationships, and consequently, a better alignment can be achieved. Therefore, if \mathbb{P}_M has a larger area than \mathbb{P}_F does, such as if \mathbb{P}_M is the full model of an object and \mathbb{P}_F is only a partial scan of the same object, we swap them to proceed with the optimization. Afterwards, the inverse transformation is taken as the final solution.

4.2 Outlier Handling

There are two types of outliers that we need to address. One type is the noisy outliers that are incorrectly measured points, such as the examples shown in Figs. 1 and 2, and the other is the correctly measured points in the nonoverlapping part of partially overlapping point sets [30].

Noisy outliers may degrade registration performance. Thus, we need to filter out as many as possible. In this study, a simple two-step pruning method is used to remove these unwanted elements. Given a point set \mathbb{P} with noisy outliers, in the first step, the following equation from robust fuzzy c-means [41] is employed to identify noisy outliers by defining a parameter η_i for each fuzzy cluster:

$$\eta_i^2 = \frac{\sum_{j=1}^{N_P} \mu_{\mathbf{c}_i}(\mathbf{p}_j)^m \cdot D(\mathbf{p}_j, \mathbf{c}_i)^2}{\sum_{j=1}^{N_P} \mu_{\mathbf{c}_i}(\mathbf{p}_j)^m}, i = 1, \dots, N_C \quad (5)$$

If \mathbf{p}_j satisfies $D(\mathbf{p}_j, \mathbf{c}_i) > \eta_i$ for all $i = 1, \dots, N_C$, then it is regarded as a noisy outlier and is removed. If a relatively large number of noisy outliers near the real points still exist after performing the first step due to serious noise pollution, a further pruning based on the per-point distance loss $J(\mathbf{p}_j, \mathbb{C})$ of Eq. (2) is applied as the second step: A small subset of \mathbb{P} that contains the elements with the largest distance losses $J(\mathbf{p}_j, \mathbb{C})$ s is removed. According to our experimental experience, the ratio of the elements of this subset to that of \mathbb{P} can be selected between 10% and 15%.

Fig. 2 shows some examples. For different degrees of noise pollution, the two-step pruning method can clean most of the noisy outliers.

Regarding the nonoverlapping part of partially overlapping scans, inspired by the trimmed ICP [30], a trimming strategy based on the per-element distance loss $J(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ of the metric in Eq. (4) is developed to pick out \mathcal{C}'_M from \mathcal{C}_M . Given a trimming ratio ξ , we have $N_{\mathcal{C}'_M} = N_{\mathcal{C}_M} \cdot (1 - \xi)$. Afterwards, at each computation of a new spatial transformation, \mathcal{C}'_M is chosen as the subset of \mathcal{C}_M with $N_{\mathcal{C}'_M}$ elements having the smallest per-element distance losses $J(\lambda, \mathcal{C}_M, \mathcal{C}_F)$ s, where λ is the current transformation. The convergence proof of this trimming strategy is similar to that of the trimmed ICP [30] and thus is omitted here. The selection of ξ is discussed in Section 4.4.3.

4.3 Average Fuzzy Point-to-Center Distance and Average Fuzzy Center-to-Center Distance

In this section, we introduce two useful indexes to enhance the accuracy and efficiency of our method. Given a point set \mathbb{P} and its fuzzy cluster centers \mathcal{C} , the distance loss $J(\mathbb{P}, \mathcal{C})$ in Eq. (2) is a weighted sum of the distances between \mathbf{p}_j and all centers using the fuzzy membership grades $\mu_{c_i}(\mathbf{p}_j)$ s as the weights. According to Eq. (3), $\mu_{c_i}(\mathbf{p}_j)$ is inversely proportional to the distance between \mathbf{p}_j and \mathbf{c}_i . Therefore, in $J(\mathbb{P}, \mathcal{C})$, a longer/shorter point-to-center distance has a smaller/larger weight. With this property, $J(\mathbb{P}, \mathcal{C})$ can be analogously regarded as the “center of gravity” of \mathbf{p}_j ’s point-to-center distances. This “center of gravity” indicates a character of \mathbf{p}_j ’s position with respect to \mathcal{C} . The average of the “center of gravity” values of the elements of \mathbb{P} is an index that can reflect the points’ disposition and dispersion related to the topology of this point set described by fuzzy clusters. We name this index *average fuzzy point-to-center distance*, abbreviated as “AFPCD” and denoted by $\bar{J}(\mathbb{P}, \mathcal{C})$. The AFPCD of a point set can be easily calculated from $J(\mathbb{P}, \mathcal{C})$ in Eq. (2) after fuzzy clustering is completed:

$$\bar{J}(\mathbb{P}, \mathcal{C}) = \frac{1}{N_{\mathbb{P}}} \cdot J(\mathbb{P}, \mathcal{C}) \quad (6)$$

Several examples of AFPCD using the Stanford bunny point sets are shown in Figs. 2, 4 and 5. In Fig. 2, the first column and (c.2) show that the noisy outliers severely change the values of AFPCD. After pruning by the two-step method, the AFPCDs become close to the original value shown in Fig. 4(b.2). In addition, Fig. 4(a.2) and Fig. 5 show that the four full models, which are described by different resolutions but use the same N_C , have AFPCDs close to one another. This fact demonstrates that when the resolution of a point set is reasonable and can clearly reflect the object’s spatial properties, its impact on AFPCD can be neglected.

On the other hand, in each row of Fig. 4, as N_C increases, the fuzzy cluster centers become more crowded, and the point-to-center distances decrease to give a gradually reduced AFPCD. In each column of Fig. 4 using the same N_C for different point sets, compared to the partial scans, the full model has a larger surface area of the object (bunny), and thus, its fuzzy cluster centers are distributed more sparsely, which leads to longer point-to-center distances and a larger AFPCD. By virtue of this property, AFPCD can be

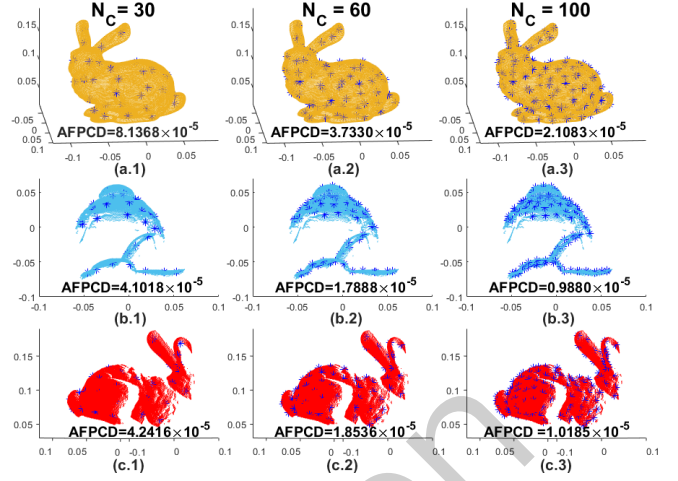


Fig. 4. Examples of AFPCD using bunny point sets. Each ‘*’ denotes a fuzzy cluster center. The first to the third rows are “bun_zipper” (full model), “bun090”, and “ear_back”, respectively.

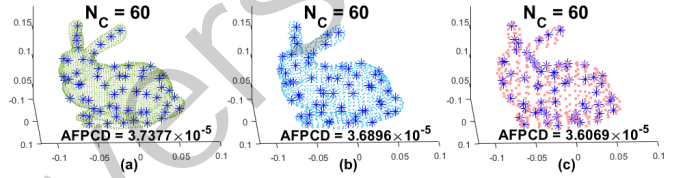


Fig. 5. The AFPCDs of three full bunny models with different resolutions. Each “*” denotes a fuzzy cluster center. (a). “bun_zipper_res2”; (b). “bun_zipper_res3”; (c). “bun_zipper_res4”.

used to compare the scanned surface area of two point sets, and we have the following strategy:

Strategy 1. Given two point sets described by equal numbers of fuzzy clusters, the point set with a larger AFPCD is chosen as \mathbb{P}_F and the other as \mathbb{P}_M for the metric in Eq. (4) to proceed with the optimization for registration.

Given two point sets \mathbb{P}_F and \mathbb{P}_M with fuzzy cluster centers \mathcal{C}_F and \mathcal{C}_M and given a transformation λ , we define another index, called the *average fuzzy center-to-center distance*, abbreviated as “AFCCD”, denoted as $\bar{J}(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$, and calculated from $J(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ of Eq. (4):

$$\bar{J}(\lambda, \mathcal{C}'_M, \mathcal{C}_F) = \frac{1}{N_{\mathcal{C}'_M}} \cdot J(\lambda, \mathcal{C}'_M, \mathcal{C}_F) \quad (7)$$

By comparing the AFCCD $\bar{J}(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ in Eq. (7) with the AFPCD $\bar{J}(\mathbb{P}, \mathcal{C})$ in Eq. (6), we know that AFCCD quantifies the disposition and dispersion of $\mathcal{T}(\lambda, \mathcal{C}'_M)$ with respect to \mathcal{C}_F . The variation of AFCCD in a registration process is as follows: At the beginning, the two point sets are misaligned, and thus, $J(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ has a relatively large value, as does AFCCD. During the optimization, AFCCD is gradually reduced as $J(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ decreases. Since the AFPCD of \mathbb{P}_F , denoted as $\bar{J}(\mathbb{P}_F, \mathcal{C}_F)$, is derived after fuzzy clustering minimizes the points’ distance losses with respect to the same centers \mathcal{C}_F , it can be taken as the goal for AFCCD to achieve the following: When the AFCCD $\bar{J}(\lambda, \mathcal{C}'_M, \mathcal{C}_F)$ equals the AFPCD $\bar{J}(\mathbb{P}_F, \mathcal{C}_F)$, the disposition and dispersion of $\mathcal{T}(\lambda, \mathcal{C}'_M)$ with respect to \mathcal{C}_F are very close to that of the points in \mathbb{P}_F with respect to \mathcal{C}_F . In this case, we

can consider that the two point sets are aligned or at least coarsely aligned.

Note that the minimum of AFCCD is generally less than the AFPCD since AFPCD is an average value: When a point p_j is near a fuzzy cluster center/relatively not close to any centers, its distance loss $J(p_j, \mathbb{C})$ is less/greater than the AFPCD $\bar{J}(\mathbb{P}, \mathbb{C})$. On the other hand, when the two scans are registered by $\mathcal{T}(\lambda)$, their fuzzy clusters of the overlapping part describe the same spatial properties of the object. Since a fuzzy cluster center is at the most representative position of its fuzzy cluster, in the overlapping region, the fuzzy cluster centers are located at the representative places of the scanned object's surface, and thus, there is a small likelihood that a $\mathcal{T}(\lambda, c'_{Mj})$ is not close to any c_{Fi} . Therefore, when the two scans are aligned, the AFCCD, which is the average of $\mathcal{T}(\lambda, c'_{Mj})$'s distance losses with respect to \mathbb{C}_F , will be generally lower than the AFPCD $\bar{J}(\mathbb{P}_F, \mathbb{C}_F)$. A more detailed analysis is given in the supplementary material. Therefore, we have the following:

Strategy 2 (registration quality assessment). We define a variable ρ_λ , which is the ratio of AFCCD to AFPCD. The two point sets \mathbb{P}_F and $\mathcal{T}(\lambda, \mathbb{P}_M)$ can be considered as aligned or coarsely aligned if they satisfy the following:

$$\rho_\lambda = \frac{\text{AFCCD}}{\text{AFPCD}} = \frac{\bar{J}(\lambda, c'_{Mj}, \mathbb{C}_F)}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F)} \leq 1 \quad (8)$$

Remark 2. Although the value of AFPCD $\bar{J}(\mathbb{P}_F, \mathbb{C}_F)$ depends on the number of fuzzy clusters N_{C_F} , the comparison between AFPCD and AFCCD is always reasonable under different choices for N_{C_F} since the two indexes are calculated based on the same fuzzy cluster centers \mathbb{C}_F . Note that **Strategy 2** may not be valid in the following cases: **i).** The overlapping ratios of the two point sets are too small; **ii).** The size difference between the two point sets is too large; **iii).** $N_{C'_M}$ is chosen to be too small; and **iv).** The trimming ratio ξ for partial overlap registration is chosen to be too large. We can ignore cases **i)** and **ii)** in this study since our method works well for the point sets satisfying **Assumption 1**. For case **iii)**, according to our experience from extensive experiments, a generally safe choice is $N_{C'_M} \geq 50$. The user can also freely choose a proper value for $N_{C'_M}$ in a certain application. For the problem of case **iv)**, we present a discussion about the selection of ξ in Section 4.4.3.

4.4 Optimization Methodology

In this section, we present a deterministic optimization scheme that integrates the BnB and gradient-based algorithms to globally minimize the fuzzy cluster-based metric for registration. This scheme is related to Go-ICP [29] but has the following advantages:

- i).** Our method does not search for the nearest correspondence. Thus, it does not need to build a distance transform.
- ii).** Our method is more robust to poor initialization in local optimization and can converge to the optimum using fewer steps in global optimization owing to the broad convergence basin of the fuzzy cluster-based metric.
- iii).** Our method can handle uncertainties better and is less sensitive to the trimming ratio's variation.
- iv).** Our method has a registration quality assessment, which yields a beneficial stop criterion for the global search and can greatly reduce the computational time and cost.

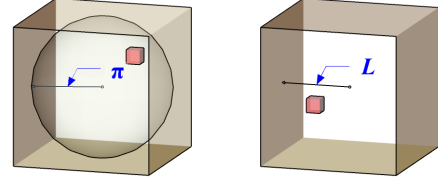


Fig. 6. Initial $SE(3)$ space for BnB, which is same as the domain parameterization of Go-ICP [29]. Left: the initial rotation cube C_r , $[-\pi, \pi]^3$. Right: the initial translation cube C_t , $[-L, L]^3$, which is supposed to include the optimal solution of translation. The small pink boxes give the examples of octant subcubes for rotation and translation.

4.4.1 Branch and Bound

The BnB algorithm in our method utilizes the same domain parameterization as that in Go-ICP [29] (shown in Fig. 6). With the axis-angle representation, the whole 3D rotation space can be compactly described by a solid ball of radius π centered at the origin. To facilitate the BnB, this solid ball is enclosed by the cube $[-\pi, \pi]^3$, as shown in the left part of Fig. 6. For translation, we suppose that the optimal solution lies in the cube $[-L, L]^3$, as shown in the right part of Fig. 6, which is easily realized by choosing a large value for L . We denote these initial rotation and translation cubes as C_r and C_t , respectively. For branch, the BnB employs the octree structure that recursively divides the subcubes into octants.

The upper and lower bounds of $J(\lambda, c'_{Mj}, \mathbb{C}_F)$ in specified rotation and translation cubes are key factors for the BnB-based optimization. The following lemma of uncertainty radii from Go-ICP [29] is used to derive the bounds.

Lemma 1. Given a 3D point p_j , a rotation cube C_r of side-length $2\sigma_r$ centered at r_0 , and a translation cube C_t of side-length $2\sigma_t$ centered at t_0 , for $\forall r \in C_r, t \in C_t$, we have the following:

$$\begin{aligned} \|\mathcal{R}(r)p_j - \mathcal{R}(r_0)p_j\| &\leq 2\sin(\min(\frac{\sqrt{3}\sigma_r}{2}, \frac{\pi}{2})) \|p_j\| \doteq \gamma_r(p_j) \\ \|(p_j + t) - (p_j + t_0)\| &\leq \sqrt{3}\sigma_t \doteq \gamma_t \end{aligned}$$

This lemma means that given a $\lambda = (r; t) \in C_r \times C_t$, $\mathcal{T}(\lambda, p_j)$ will lie in a sphere of radius $\gamma_r(p_j) + \gamma_t$ centered at $\mathcal{T}(\lambda_0, p_j)$, where $\lambda_0 = (r_0; t_0)$ [29]. Note that $\gamma_r(p_j)$ is point-dependent. For convenience, we denote $\gamma_r(c'_{Mj})$ by γ_{rj} and use a vector γ_r to denote all γ_{rj} s.

According to [29], the upper bound of the per-element distance loss can be set as $J(\lambda_0, c'_{Mj}, \mathbb{C}_F)$, while finding a suitable lower bound is much more difficult. Compared to the metric of Go-ICP, which is the sum of binary L_2 errors between the nearest points, the fuzzy cluster-based metric in Eq. (4) has a much more complex form. Hence, the task to find its lower bound is harder. We submit the fuzzy membership calculation in Eq. (3) to Eq. (4) and obtain a new expression for our fuzzy cluster-based metric as follows:

$$\begin{aligned} J(\lambda, \mathbb{C}'_M, \mathbb{C}_F) &= \sum_{j=1}^{N_{C'_M}} J(\lambda, c'_{Mj}, \mathbb{C}_F) \\ &= \sum_{j=1}^{N_{C'_M}} \left(\sum_{i=1}^{N_{C_F}} D(\lambda, c'_{Mj}, c_{Fi})^{\frac{2}{1-m}} \right)^{1-m} \quad (9) \end{aligned}$$

where the per-element distance loss $J(\lambda, c'_{Mj}, \mathbb{C}_F)$ becomes

$$J(\lambda, c'_{Mj}, \mathbb{C}_F) = \begin{cases} 0, & \text{if } \forall_{i=1, \dots, N_{C_F}} D(\lambda, c'_{Mj}, c_{Fi}) = 0 \\ \left(\sum_{i=1}^{N_{C_F}} D(\lambda, c'_{Mj}, c_{Fi})^{\frac{2}{1-m}} \right)^{1-m}, & \text{else} \end{cases} \quad (10)$$

Based on Eqs. (9) and (10), we have the following theorem, whose proof is included in the supplementary material:

Theorem 1 (bounds of the metric). For a given region $C_r \times C_t$ centered at $\lambda_0 = (r_0; t_0)$ with the uncertainty radii γ_{rj} and γ_t , the upper and lower bounds of the distance loss $J(\lambda, c'_{Mj}, \mathbb{C}_F)$, denoted by $J_{upp}(\lambda, c'_{Mj}, \mathbb{C}_F)$ and $J_{low}(\lambda, c'_{Mj}, \mathbb{C}_F)$, respectively, can be selected as follows:

$$J_{upp}(\lambda, c'_{Mj}, \mathbb{C}_F) = J(\lambda_0, c'_{Mj}, \mathbb{C}_F) \quad (11)$$

$$J_{low}(\lambda, c'_{Mj}, \mathbb{C}_F) = \begin{cases} 0, & \text{if } \forall_{i=1, \dots, N_{C_F}} D(\lambda_0, c'_{Mj}, c_{Fi}) - (\gamma_{rj} + \gamma_t) \leq 0 \\ \left(\sum_{i=1}^{N_{C_F}} \left[D(\lambda_0, c'_{Mj}, c_{Fi}) - (\gamma_{rj} + \gamma_t) \right]^{\frac{2}{1-m}} \right)^{1-m}, & \text{else} \end{cases} \quad (12)$$

As a result, the upper and lower bounds of the fuzzy cluster-based metric $J(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$, denoted by $J_{upp}(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$ and $J_{low}(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$, respectively, are obtained as follows:

$$\begin{cases} J_{upp}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} J_{upp}(\lambda, c'_{Mj}, \mathbb{C}_F) \\ J_{low}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} J_{low}(\lambda, c'_{Mj}, \mathbb{C}_F) \end{cases} \quad (13)$$

4.4.2 Gradient-based Local Convergence

The gradient of $J(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$ with respect to λ , denoted as $\nabla_{\lambda} J(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$, can be expressed as follows:

$$\nabla_{\lambda} J(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} \frac{\partial J(\lambda, c'_{Mj}, \mathbb{C}_F)}{\partial \lambda} \quad (14)$$

To obtain each $\partial J(\lambda, c'_{Mj}, \mathbb{C}_F) / \partial \lambda$, according to Eq. (10), we need to calculate $\partial D((\lambda, c'_{Mj}, c_{Fi}) / \partial \lambda \in \mathbb{R}^6$ for $i = 1, \dots, N_{C_F}$. The details are presented in the supplementary material. When having an estimate for λ , the standard gradient-based optimization algorithms, such as the BFGS quasi-Newton [49], can be applied to the metric to refine λ through a further convergence in the local area.

4.4.3 Global Optimization Scheme

Before presenting the optimization scheme, we introduce a trade-off in our method: When using a large number of fuzzy clusters to describe each point set, for example, more than 1000, the registration can achieve high accuracy but is likely to be computationally expensive and time-consuming in global search and local convergence, as well as in fuzzy clustering; when the numbers of the fuzzy clusters are small, say, less than 100, both fuzzy clustering and optimization can be completed in much less time, but the registration may not be sufficiently satisfactory since the two point sets will generally be only coarsely aligned. To resolve this problem, we design the following coarse-to-fine strategy:

Coarse registration: In this step, fuzzy clustering is applied to describe each of the two point sets by a relatively small number of fuzzy clusters (if noisy outliers exist, we

Algorithm 1 The main algorithm of global optimization for the fuzzy cluster-based metric

Input: $\mathbb{P}_F, \mathbb{P}_M, \mathbb{C}_F, \mathbb{C}_M$ and AFPCD $\bar{J}(\mathbb{P}_F, \mathbb{C}_F)$; trimming ratio ξ ; threshold ε ; side-length $2\sigma_{min_r}$ of the minimum cube; initial cubes C_r and C_t .

Output: optimal transformation $\lambda^* = (r^*; t^*)$.

- 1: Initialize a queue Q_r with only one element C_r .
- 2: $\lambda^* = (r^*; t^*) \leftarrow$ local convergence with zero as the initial estimate $\lambda^{(0)}$.
- 3: **start loop**
- 4: Quit the loop if Q_r is empty.
- 5: Quit the loop if $\rho_{\lambda^*} = \frac{J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F)}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F) \times N_{C'_M}} \leq 1$.
- 6: Pick out a cube with the lowest J_{low_r} from Q_r .
- 7: Quit the loop if $J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F) - J_{low_r} < \varepsilon$ or the side-length of this cube is smaller than $2\sigma_{min_r}$.
- 8: Divide the cube into 8 subcubes (octants).
- 9: **for** each subcube C_r **do**
- 10: Calculate J_{upp_r} for C_r and the translation \hat{t} through Algorithm 2 with r_0 , zero γ_r and λ^* .
- 11: **if** $J_{upp_r} < J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F)$ **then**
- 12: $\lambda^* \leftarrow$ local convergence with $\lambda^{(0)} = (r_0; \hat{t})$.
- 13: **end if**
- 14: Calculate J_{low_r} for C_r and the translation \check{t} through Algorithm 2 with r_0 , γ_r and λ^* .
- 15: **if** $J_{low_r} \geq J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F)$ or $\frac{J_{low_r}}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F) \times N_{C'_M}} \geq 1$ **then**
- 16: Discard C_r .
- 17: **else** Place C_r into Q_r .
- 18: **end if**
- 19: **end for**
- 20: **end loop**
- 21: $\lambda^* = (r^*; t^*) \leftarrow$ fine registration with $\lambda^{(0)} = \lambda^*$.

use the pruning method introduced in Section 4.2 to remove them). Note that N_{C_F} and N_{C_M} should be equal such that we can select the point set with a larger area according to **Strategy 1**, and the value of $N_{C'_M}$ should be greater than a lower limit to make **Strategy 2** valid. Specifically, $N_{C'_M}$ can be selected between [50, 100]. Afterwards, the BnB-based global search is performed to obtain a coarse alignment.

Fine registration: In this step, each point set is described by a relatively large number of fuzzy clusters. According to **Remark 1**, the points can be directly taken as the fuzzy cluster centers. Generally, a range scan contains a very large quantity of points. To ease the computational burden, we downsample each point set to an appropriate size. Afterwards, taking the coarse alignment as the initialization, the gradient-based local convergence is performed on the fuzzy cluster-based metric, which uses the downsampled points as the fuzzy cluster centers, to achieve a precise alignment. According to the experimental experience, a good choice is to downsample \mathbb{P}_F to approximately 1000-1500 points and downsample \mathbb{P}_M to approximately 1500-2000 points, which can provide a satisfactory performance with an acceptable time cost.

In the coarse registration, both fuzzy clustering and optimization require low costs in time and calculation since the numbers of fuzzy clusters are small. In the fine registration, the points are directly taken as the fuzzy cluster

Algorithm 2 Inner search for optimal translation

Input: \mathbb{C}_F , \mathbb{C}_M , and AFPCD $\bar{J}(\mathbb{P}_F, \mathbb{C}_F)$; trimming ratio ξ ; threshold ε ; side-length $2\sigma_{min_t}$ of the minimum cube; \mathcal{C}_t ; \mathbf{r}_0 , γ_r and best achieved λ^* .

Output: J^{**} and \mathbf{t}^{**} .

- 1: Initialize a queue Q_t with only one element \mathcal{C}_t .
- 2: $\mathbf{t}^{**} \leftarrow \mathbf{t}_0$ of \mathcal{C}_t .
- 3: **if** $J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F) > J_{upp_t}$ of \mathcal{C}_t **then**
- 4: $J^{**} \leftarrow J_{upp_t}$ of \mathcal{C}_t .
- 5: **else** $J^{**} \leftarrow J(\lambda^*, \mathbb{C}'_M, \mathbb{C}_F)$.
- 6: **end if**
- 7: **start loop**
- 8: Quit the loop if Q_t is empty.
- 9: Quit the loop if $\frac{J^{**}}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F) \times N_{C'_M}} \leq 1$.
- 10: Pick out a cube with the lowest J_{low_t} from Q_t .
- 11: Quit the loop if $J^{**} - J_{low_t} < \varepsilon$ or the side-length of this cube is smaller than $2\sigma_{min_t}$.
- 12: Divide the cube into 8 subcubes (octants).
- 13: **for** each subcube \mathcal{C}_t **do**
- 14: Calculate J_{upp_t} with \mathbf{r}_0 , \mathbf{t}_0 and γ_r .
- 15: **if** $J_{upp_t} < J^{**}$ **then**
- 16: $\mathbf{t}^{**} \leftarrow \mathbf{t}_0$, $J^{**} \leftarrow J_{upp_t}$.
- 17: **end if**
- 18: Calculate J_{low_t} with \mathbf{r}_0 , \mathbf{t}_0 , γ_r and γ_t .
- 19: **if** $J_{low_t} \geq J^{**}$ or $\frac{J_{low_t}}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F) \times N_{C'_M}} \geq 1$ **then**
- 20: Discard \mathcal{C}_t .
- 21: **else** Place \mathcal{C}_t into Q_t .
- 22: **end if**
- 23: **end for**
- 24: **end loop**

centers, which saves the time-consuming fuzzy clustering for locating substantial centers. Moreover, since the two point sets have been coarsely aligned, the gradient-based local convergence in the fine registration can obtain the exact result within a short span of time. This coarse-to-fine strategy can effectively address the trade-off issue.

Next, we present the global optimization scheme. To reduce the computational burden, the nested BnB structure presented in Go-ICP [29], which consists of an outer BnB search in rotation space and an inner BnB search in translation space, is employed in our method. For the outer search, given a rotation cube \mathbb{C}_r centered at \mathbf{r}_0 , the bounds of the fuzzy cluster-based metric are rewritten as follows:

$$\begin{cases} J_{upp_r}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \min_{t \in \mathcal{C}_t} \sum_{j=1}^{N_{C'_M}} J((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbb{C}_F) \\ J_{low_r}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \min_{t \in \mathcal{C}_t} \sum_{j=1}^{N_{C'_M}} J_{low_r_0}((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbb{C}_F) \end{cases} \quad (15)$$

where $J_{low_r_0}((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbb{C}_F)$ is

$$\begin{cases} 0, & \text{if } \forall_{i=1, \dots, N_{C_F}} D((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - \gamma_{rj} \leq 0 \\ \left(\sum_{i=1}^{N_{C_F}} \left[D((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - \gamma_{rj} \right]^{\frac{2}{1-m}} \right)^{1-m}, & \text{else} \end{cases} \quad (16)$$

For each \mathbb{C}_r , the inner BnB finds the optimal translation to derive the bounds described in Eq. (15) by starting the

division (branch) from the initial translation cube \mathcal{C}_t . Given a translation cube \mathcal{C}_t centered at \mathbf{t}_0 , the upper bound of the metric is the sum of $J_{low_r_0}((\mathbf{r}_0; \mathbf{t}), \mathbf{c}'_{Mj}, \mathbb{C}_F)$ s at \mathbf{t}_0 , and the lower bound is the sum of $J_{low}(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ s in Eq. (12):

$$\begin{cases} J_{upp_t}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} J_{low_r_0}((\mathbf{r}_0; \mathbf{t}_0), \mathbf{c}'_{Mj}, \mathbb{C}_F) \\ J_{low_t}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} J_{low}(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) \end{cases} \quad (17)$$

J_{low_r} in Eq. (15) is derived by the inner BnB, where \mathbf{r}_0 and γ_r of Eq. (17) are given from the current outer search, and J_{upp_r} in Eq. (15) is derived by the inner BnB with the same \mathbf{r}_0 but setting all γ_{rj} s to be zero in Eq. (17).

In the BnB of Go-ICP, a cube in the queue with the lowest lower bound has the highest priority to be selected, and the cube with the lower bound greater than the best achieved error is discarded. When a better solution is found by BnB, the traditional ICP is applied for a further local convergence. In addition, Go-ICP will stop when the difference between the best achieved error and the currently lowest lower bound is less than a predefined threshold ε , or the size of the cubes in the queue is sufficiently small or the queue is empty. In our method, the above lowest-first selection, cube discard strategy, and stop criteria are also employed. Similarly, the gradient-based local convergence is integrated for refinement each time a smaller distance loss is found by BnB. Furthermore, we add two new criteria as follows according to the registration quality assessment.

Strategy 3 (new criteria for global optimization).

Criterion I. A cube is removed from the queue if the metric in it has the lower bound not less than $\text{AFPCD} \times N_{C'_M}$.

Criterion II. When the best achieved λ gives $\rho_\lambda \leq 1$, which means $J(\lambda, \mathbb{C}'_M, \mathbb{C}_F) \leq \text{AFPCD} \times N_{C'_M}$, the global search stops and passes this λ to the fine registration.

The optimization scheme are described by Algorithms 1 and 2, which are extended from the outer and inner BnB search of Go-ICP. The main differences of the scheme compared to Go-ICP are the following: **i).** Our scheme optimizes the fuzzy cluster-based metric for registration; **ii).** Our scheme utilizes the two criteria of **Strategy 3**, which appear in lines 5 and 15 of Algorithm 1 and lines 9 and 19 of Algorithm 2, to improve the computational efficiency of the global optimization; and **iii).** Our scheme proceeds in a coarse-to-fine fashion. Some experiments in the next section will demonstrate the superiority of our scheme over Go-ICP.

We now discuss the selection of the trimming ratio ξ . For partial overlap registration, in the global search (coarse registration), using an overlage ξ may lead to a premature stop, where λ satisfies $\rho_\lambda \leq 1$ but is an erroneous solution. When ξ is less than its real value, AFCCD will increase since some \mathbf{c}_{Mj} s of the nonoverlapping part, which have larger center-to-center distances, are included. When ξ is too small, the AFCCD of the optimal solution may be greater than the AFPCD, and then Criterion II of **Strategy 3** will not be used, but our scheme can still converge to the optimal solution when reaching other stop criteria. In the fine registration, with a too-small ξ , the result may not be sufficiently satisfactory due to the negative effects yielded by \mathbf{c}_{Mj} s of the nonoverlapping part, while the result could be accurate by using a relatively larger ξ since it can guarantee that the included \mathbf{c}_{Mj} s are inliers. Therefore, the user can choose

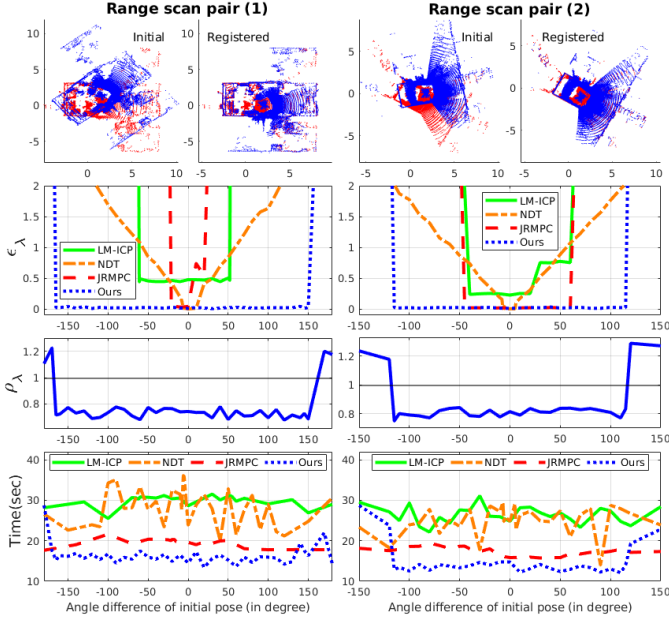


Fig. 7. Local convergence tests for LM-ICP [2], NDT [15], JRMP [1] and our method. The first row shows the two pairs of partially overlapping scans used in the tests, where an example of their initial poses is given in each left part. The second row shows the errors ϵ_λ . The third row shows the values of ρ_λ to validate the proposed registration quality assessment. The last row presents the time costs.

a relatively small ξ for the coarse registration to avoid the premature stop and then choose a slightly larger ξ in the fine registration for precise alignment. A suggestion on the basis of our experimental experience is the following:

$$\xi_{fine} = \begin{cases} 0.75 \times \xi + 0.075, & \text{if } \xi < 10\% \\ 0.5 \times \xi + 0.1, & \text{if } 10\% \leq \xi < 20\% \\ \xi, & \text{else} \end{cases} \quad (18)$$

where ξ is used in the coarse registration and ξ_{fine} is used in the fine registration. ξ can be determined by visually observing the two scans and roughly guessing their nonoverlapping ratios. If a rough guess is not available, the user can start with a small ξ and then gradually increase it for a test. Some experimental results in the next section demonstrate that our method is robust to variation in ξ .

5 EXPERIMENTS

The proposed method¹ is implemented in MATLAB R2018b on a PC with 8 GB RAM and an i5 2.30 GHz Intel CPU. For fuzzy c-means clustering, we choose $m = 2$ and set the iteration number to 100. We found that this setting works well for different point sets of registration. When using the two-step pruning method, 15% of the points with the greatest distance losses are removed in the second step. For partial overlap cases, when trimming is applied, Eq. (18) is applied. For global optimization, as in Go-ICP [29], we prenormalize the two point sets with the same scaling gains to include all the points in the domain $[-1, 1]^3$ and choose the initial rotation and translation cubes as $\mathcal{C}_r = [-\pi, \pi]^3$

1. Source code can be found at <https://gitsvn-nt.oru.se/qianfang.liao/FuzzyClusterBasedRegistration>

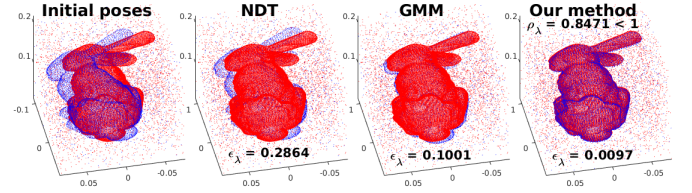


Fig. 8. Registration results of NDT [15], GMM [12] and our method for two full bunny models with noisy outliers. The fixed set (red) is “bun_zipper”, and the moving set (blue) is “bun_zipper_res2”. The noisy outliers are same as those in Fig. 1. The angle difference of the initial poses is 20° . The time costs of the optimization of NDT, GMM and our method are 0.9098 s, 1.1913 s, and 0.3730 s, respectively.

and $\mathcal{C}_t = [-0.5, 0.5]^3$. Note that the AFPCDs and AFCCDs will be different from their original values due to the scale’s change, but the registration quality assessment will still be valid since the two point sets are of the same scale. In fine registration, \mathbb{P}_F and \mathbb{P}_M are downsampled to approximately 1500 and 2000 points, respectively, to be the fuzzy cluster centers. The error of the derived $\lambda = (r; t)$ is defined as $\epsilon_\lambda = \|\lambda - \lambda_g\|$, where $\lambda_g = (r_g; t_g)$ denotes the ground truth transformation. One degree of the rotation angle difference will lead to an increment of 0.0175 in ϵ_λ .

5.1 Local Convergence Tests

In this section, we test the local convergence of the fuzzy cluster-based metric by solely using the gradient-based algorithm in the coarse-to-fine optimization and compare the results with those of three existing methods: LM-ICP [2], NDT [15] and JRMP [1]. This test is based on two pairs of partially overlapping scans recorded by Dr. Magnusson in our laboratory (the Robot Lab, AASS building, Örebro University, Sweden) and publicly available in [50]. The transformations given by [50] are taken as the ground truth. Each scan has more than 110,000 points, and we downsample it to approximately 8000 points to apply the four registration methods for efficiency. The grid size for NDT is 0.5 m [15], and the number of components for JRMP is chosen to be 300. In our method, we choose $N_{CF} = N_{CM} = 100$ for fuzzy clustering. Fig. 7 shows the registration performances of the four methods with respect to incrementally enlarged pose differences of each pair, where the initial estimates used by the four methods in all the cases are zero and no trimming is applied. In the convergence range, LM-ICP cannot adequately handle the nonoverlapping parts and yields relatively large errors in translation. Thus, its ϵ_λ s are greater than those of the other three methods. Among the four methods, our metric achieves the broadest convergence range and requires the lowest time cost in the convergence range. Moreover, the registration quality assessment yields correct results that $\rho_\lambda < 1$ for accurate alignment and $\rho_\lambda > 1$ for misalignment. Note that the AFCCDs and AFPCDs of ρ_λ are calculated based on the fuzzy clusters used in the coarse registration. The following experiments use the same method to derive ρ_λ .

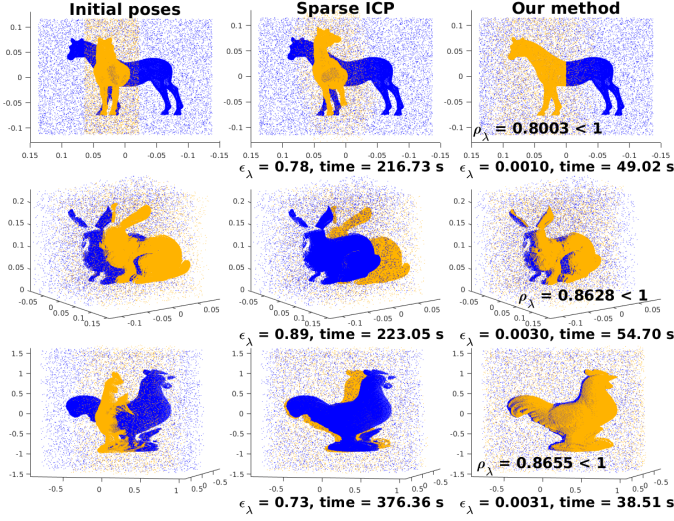


Fig. 9. Registration results of sparse ICP [9] and our method for noisy point sets. The first row shows a partial-to-full registration, and the remaining rows show two partially overlapping registrations. For each point set, the number of noisy outliers is 20% that of the real points, and the noisy outliers are distributed in the same way as that in Fig. 1. For the noisy horse scan pair, $N_{P_F} = 58182$ and $N_{P_M} = 30521$. For the noisy bunny scan pair, $N_{P_F} = 48308$ and $N_{P_M} = 48117$. For the noisy chicken scan pair, $N_{P_F} = 35422$ and $N_{P_M} = 36198$.

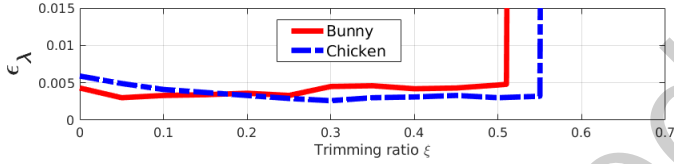


Fig. 10. The registration errors of our method for the partially overlapping bunny and chicken scans in Fig. 9 with different trimming ratios.

5.2 Robustness Tests

In this section, first, we employ two full bunny models [47] with different resolutions and with manually added noisy outliers to compare the robustness among NDT [15], GMM [12] and our method. In this test, we choose equal numbers of components, 64, for the three representations. In addition, our method solely uses coarse registration with the gradient-based algorithm for optimization. The initialization is zero for the three methods. Fig. 8 shows that our method outperforms the other two cluster representations in terms of both the accuracy and time cost of optimization. The reason is presented in Fig. 1 and described in Section 3. In addition, the ρ_λ of the final solution of our method is less than 1.

Next, we compare the robustness between sparse ICP [9] and our method based on three point set pairs (horse [51], bunny [47] and chicken [52]) with manually added noisy outliers, as shown in Fig. 9. In our method, $N_{C_F} = N_{C_M} = 80$ for fuzzy clustering. For fair comparison, our method solely uses the gradient-based algorithm in the coarse-to-fine procedure. The horse registration is a partial-to-full case, and thus, no trimming is applied. The bunny and chicken scans are partial overlap cases, and we choose $\xi = 20\%$ for them. Fig. 9 shows the performances. Com-

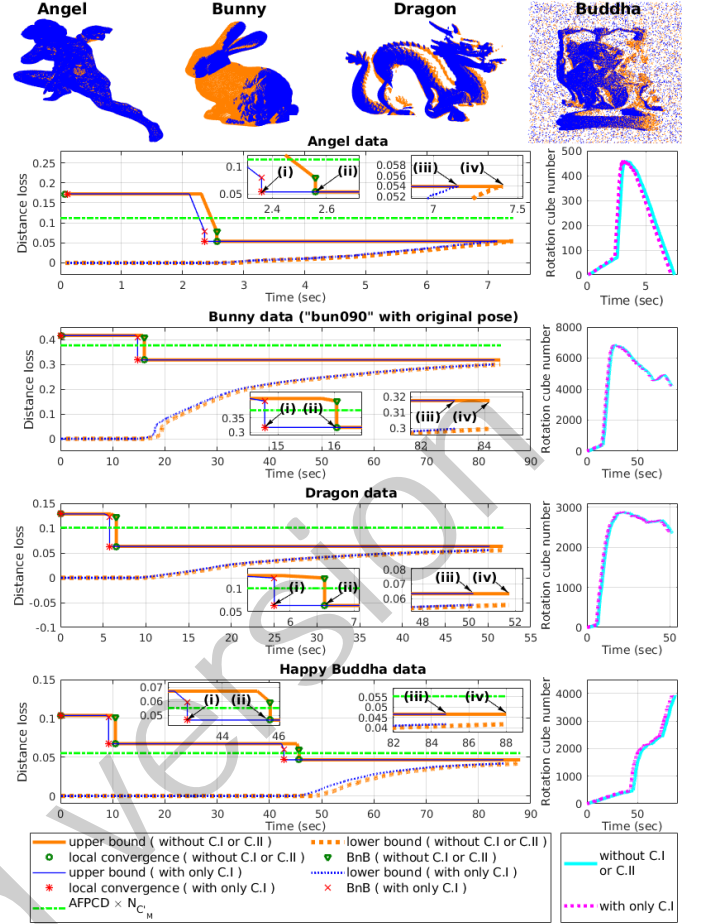


Fig. 11. The ablation study of Strategy 3. The first row shows the four scan pairs used in this study. For the rest of the figure, the right part shows the rotation cube numbers of the queue, and the left part shows the bounds and $\text{AFPCD} \times N_{C_M}$ in the global optimization (coarse registration). The upper bound denotes the distance loss of the currently best achieved λ , and the lower bound denotes the current lowest lower bound. C.I and C.II are short for Criterion I and Criterion II, respectively. (i), (ii), (iii) and (iv) mark the spots where the global optimization stops in different conditions: (i). Both C.I and C.II are applied; (ii). Only C.II is applied; (iii). Only C.I is applied; (iv). Neither C.I nor C.II is applied.

pared to sparse ICP, our method gives correct results within less time (the time cost covers fuzzy clustering, outlier pruning, coarse and fine registration). In addition, the final ρ_λ s are less than 1.

The sparse ICP's failure is due to its narrow convergence range: A good initialization is needed to derive a correct result [9], and zeros are poor initial estimates for it to align the three pairs in Fig. 9. Compared to sparse ICP, our metric has a broader local convergence range and thus is more robust to poor initialization. However, for partial overlap cases, our method needs to estimate a trimming ratio ξ . For this issue, we test our method with different ξ s to align the bunny and chicken scans. The results are shown in Fig. 10. The bunny scans can be registered when ξ is between 0 and 51%, and the chicken scans can be registered when ξ is between 0 and 55%. These results demonstrate that our metric using local optimization can handle relatively large variation in ξ : It can suppress the negative effects when some nonoverlapping elements are involved by using

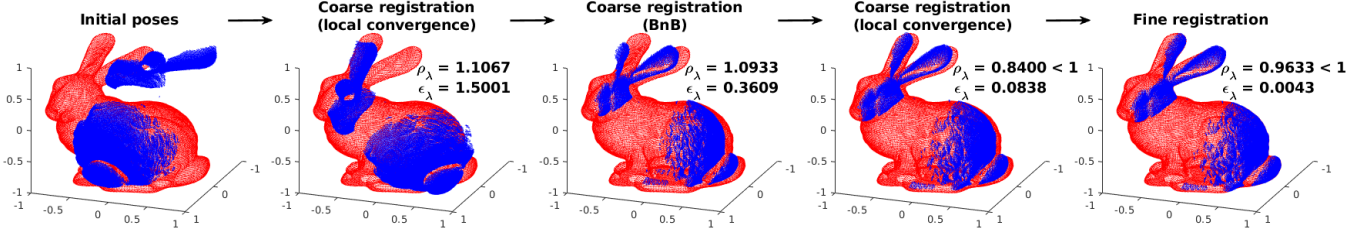


Fig. 12. The registration process of bunny data “bun_zipper” and “bun090” with original poses using the proposed method, where the three steps of the coarse registration correspond to the results updated by BnB and local convergence shown in Fig. 11

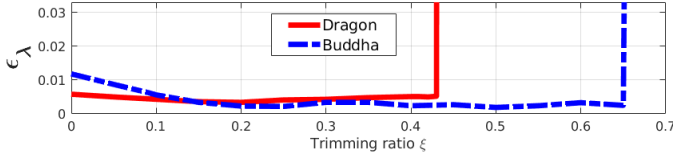


Fig. 13. The registration errors of our method for the partially overlapping dragon and Buddha scans in Fig. 11 with different trimming ratios.

a relatively small ξ and also can achieve accurate results when some inliers are trimmed by using a relatively large ξ . In the next section, we will demonstrate that our metric using global optimization is also very robust to variation in ξ . This property allows the user to roughly guess a ξ .

5.3 Global Optimization Tests

This section presents a number of tests for the global optimization scheme described in Algorithms 1 and 2.

Optimality tests. To verify that the global optimization scheme can converge to optimal solutions and achieve correct alignments, we employ the angel model from [51] and the bunny, dragon and happy Buddha data from [47] to form four different cases for the optimality tests as follows:

Full-to-full registration. We perform 100 tests of registering the angel model to its copy with random initial poses.

Partial-to-full registration. Ten pairs are formed by the bunny data, in which the fixed sets are the reconstructed full model and the moving sets are the 10 partial scans. For each pair, in addition to the original pose, we randomly select 99 initial poses for the moving set to perform 100 tests in total.

Partial overlap registration. Ten pairs of partially overlapping dragon scans are selected. For each pair, $\xi = 20\%$ and 100 tests with random initial poses are performed.

Noisy partial overlap registration. Four pairs of partially overlapping happy Buddha scans are selected. We add noisy outliers to each scan, where these outliers have the same ratio and same distribution as that in Fig. 1. For each pair, we perform 100 tests with random initial poses and $\xi = 20\%$.

In these 2500 test cases, to perform fuzzy clustering, each clean point set is downsampled to approximately 8000 points for efficiency, while no downsampling is applied to the noisy point sets. In addition, we choose $N_{CF} = N_{CM} = 50$ for the full-to-full and partial-to-full pairs and choose $N_{CF} = N_{CM} = 80$ for the rest. As expected, all the tests achieve correct alignments. Moreover, their global optimizations are all stopped by the registration quality assessment. The mean/largest ϵ_{λ} s and time costs are given in Table 1,

where the time cost covers fuzzy clustering, outlier pruning if needed, and coarse and fine registrations. Note that since each noisy Buddha scan has approximately 30,000 points, they require more time in fuzzy clustering and incur extra costs for outlier pruning when compared to the situation with the clean scans. Thus, the time costs of Buddha data are greater than those of the other three groups.

TABLE 1
Registration results of the optimality tests

Range data	ϵ_{λ} s (mean/largest)	time costs (mean/largest)
Angel	0.0027 / 0.0083	7.616 s / 17.717 s
Bunny	0.0048 / 0.0102	20.31 s / 42.34 s
Dragon	0.0041 / 0.0098	25.78 s / 72.05 s
Buddha	0.0050 / 0.0116	52.29 s / 127.28 s

Ablation study. To evaluate the improvement yielded by each criterion of **Strategy 3**, we select one test case from each of the four range data groups in the optimality tests to perform an ablation study, where the selected bunny pair is composed of the full model and the partial scan “bun090” in their original poses. Four conditions are analyzed in this ablation study: (i). Both Criteria I and II are applied; (ii). Only Criterion II is applied; (iii). Only Criterion I is applied; and (iv). Neither Criterion I nor Criterion II is applied.

TABLE 2
Time costs of the global optimization in the ablation study in Fig. 11

Range data	(i)	(ii)	(iii)	(iv)
Angel	2.3601 s	2.5593 s	7.1505 s	7.4134 s
Bunny	14.7674 s	16.0481 s	83.0716 s	84.1582 s
Dragon	5.7506 s	6.5409 s	50.1883 s	51.7468 s
Buddha	42.7620 s	45.6922 s	84.7893 s	87.9895 s

For each of the four selected test cases, we run two times of the global optimization, where one applies neither of the criteria and the other solely applies Criterion I, to obtain the results of conditions (iii) and (iv). For the stop criteria of global search, we set the threshold ϵ to be 0 and terminate the BnB when the divided subcubes reach a specified small size: The side-length is 0.02. Fig. 11 shows the results: The upper bound (the distance loss of the best achieved λ), collaboratively updated by BnB and local convergence, cascades down and remains unchanged when reaching the optimal solution, and the lower bound gradually increases to approach the upper bound. By drawing a line of $\text{AFPCD} \times N_{CM}'$ in each test case to compare against the upper bound, we can obtain the results of conditions (i)

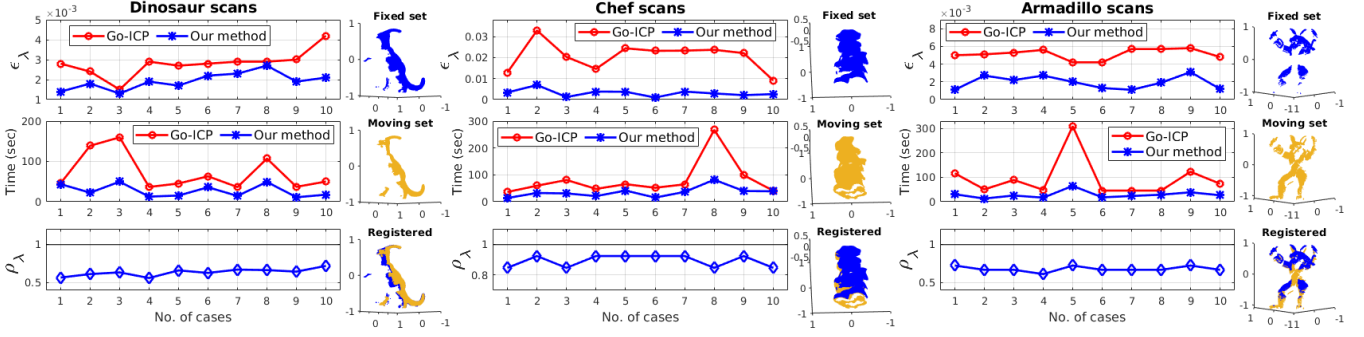


Fig. 14. The registration results of Go-ICP and our method for three partially overlapping scan pairs: dinosaur, chef and armadillo.

and (ii), in which the global optimization will stop when the upper bound becomes less than $\text{AFPCD} \times N_{C'_M}$.

Under the conditions using Criterion I, a few seconds can be saved since Criterion I can remove some unqualified cubes from the queue to reduce the time of sorting. Under the conditions without using Criterion II, the global search does not know whether a solution (upper bound) is good or not. As a result, after the upper bound reaches the optimal solution, it has to run for a relatively long extra period to raise the lowest lower bound and meet the predefined stop criteria. An important result is that among all the upper bounds in the global optimization, only the optimal solutions derived by the last local convergence are less than the associated $\text{AFPCD} \times N_{C'_M}$ to have $\rho_\lambda < 1$ (sometimes the initial estimate for the last local convergence obtained by BnB also has $\rho_\lambda < 1$, such as the angel data shown in Fig. 11). Therefore, under the conditions in which Criterion II is applied, once the upper bound reaches a good solution, the global optimization will be aware of this fact and stop in time. The time costs of the four conditions of each test case in Fig 11 are presented in Table 2, where using Criterion II reduces the time cost by up to 88.5%.

Similar to the conditions (iii) and (iv) in Fig. 11, Go-ICP has to run for an extra period to meet the predefined stop criteria after reaching the optimal solution (see Figs. 11 and 17 of [29]). The reason is as follows: Go-ICP utilizes the nearest points' distances as the metric, and it is hard to know how small this metric is to achieve a correct alignment without ground truth or user intervention. In addition, the minimum value of this metric varies with different scan resolutions and can be easily affected by outliers. Unlike Go-ICP, our method has a registration quality assessment, and it is robust to the variation in the scan's resolution and the existence of outliers. On the other hand, Go-ICP fails to align the noisy scans shown in Figs. 8, 9 and 11 since it is soon trapped in a local minimum and yields a misalignment due to the false correspondences caused by the large number of noisy outliers. As stated in [29], this problem can be resolved if Go-ICP switches to a robust metric, such as the L_p error of sparse ICP [9]. However, according to the analysis presented in Section 5.2, sparse ICP has a narrow convergence range. As a result, it may take many steps to converge in the global search. Moreover, it still does not know whether a solution is good without the ground truth and needs to spend extra costs to meet the predefined stop criteria after reaching the optimal solution. Therefore, our method is more efficient.

Moreover, the motions of the bunny point sets in this ablation study corresponding to the registration process in Fig. 11 are shown in Fig. 12. Compared to Go-ICP using five steps to align the same point sets (see Fig. 10 of [29]), our method needs fewer steps (three in the coarse registration) because of the broad convergence range of the fuzzy cluster-based metric. The motions of the other three point set pairs in Fig. 11 are presented in the supplementary material.

Tests with different trimming ratios. To test the robustness of our method to variation in the trimming ratio ξ , we apply our method using different ξ s to register the dragon and Buddha scan pairs used in the ablation study. The results are shown in Fig. 13. The dragon scans can be registered when ξ is between 0 and 43%, and the Buddha scans can be registered when ξ is between 0 and 65%. These results together with those shown in Fig. 10 demonstrate that our method is robust to variation in ξ , and the registration quality assessment can work without a premature stop even if ξ varies in a relatively large interval.

5.4 More Tests

In this section, different types of point sets are employed to test and compare the registration performances of our method and the state-of-the-art approaches. We choose $N_{C_F} = N_{C_M} = 80$ for the fuzzy clustering in our method.

First, we consider three partially overlapping point set pairs: The partial scans of dinosaur [52], chef [52] and armadillo [47], are selected to compare Go-ICP [29] and our method, where the trimming ratios are chosen to be 15%, 20%, and 25%, respectively. We randomly select 10 different initial poses for each pair and downsample each scan to approximately 5000 points to perform the two registration methods. In Go-ICP, a $300 \times 300 \times 300$ distance transform is used, and the threshold for stopping is $\varepsilon = 0.001 \times N_{P_M}$. The registration results are shown in Fig. 14. In each case, our method gives a smaller error and requires less time than Go-ICP does. In addition, the global search of our method is stopped by the registration quality assessment, and the final $\rho_\lambda < 1$. Note that the time cost of Go-ICP includes distance transform construction (approximately 36 s on our computer) and registration, and the time cost of our method covers fuzzy clustering, coarse and fine registration. For a large part of these 30 cases, the time used by our method is shorter than the time that Go-ICP spends on building distance transform only. Table 3 shows the improvements

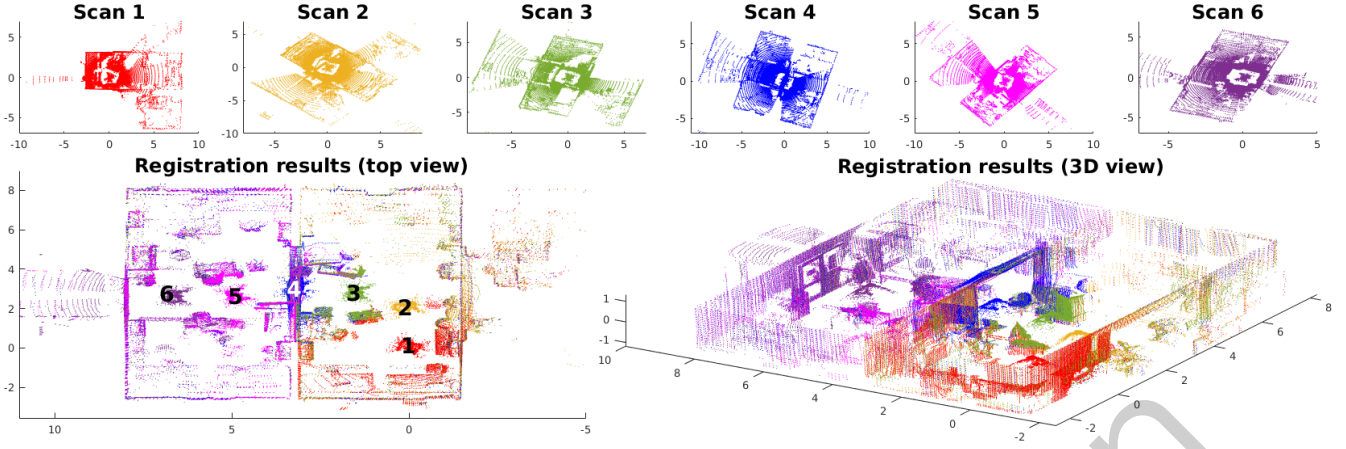


Fig. 15. The six partial scans consecutively recorded in our laboratory and their coherent 3D model built using the proposed registration method. In the bottom-left subfigure, the number i ($i = 1, \dots, 6$) indicates the position of the scanner device when taking the i th scan.

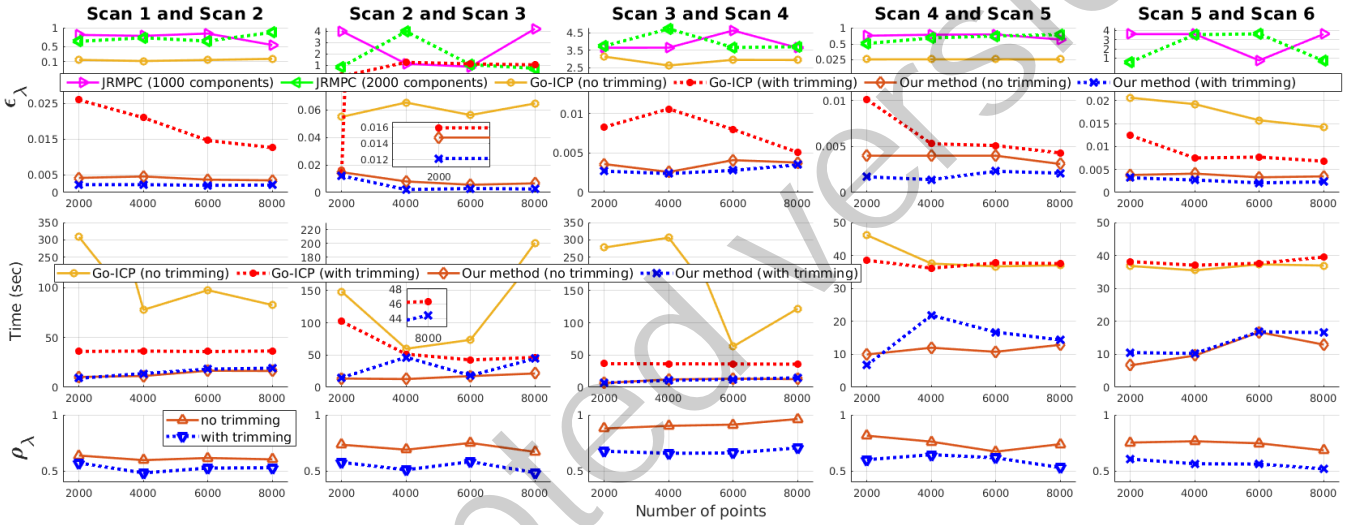


Fig. 16. The registration results of JRMPC, Go-ICP and our method for the six partial scans in Fig. 15. When each scan is downsampled to 2000, 4000, 6000 and 8000 points, the time costs of JRMPC using 1000/2000 components are 66.8 s/133.8 s, 129.1 s/271.1 s, 201.7 s/404.9 s, and 269.1 s/541.3 s, respectively, and the total time costs of our method with/without trimming to align the five pairs are 47.96 s/47.12 s, 97.22 s/57.89 s, 82.50 s/74.76 s, and 109.21 s/76.20 s.

in the registrations of the three pairs using our method when compared to the results using Go-ICP. Our method can reduce the error by up to 95.8% and reduce the time cost by up to 83.9%.

TABLE 3

The mean/largest reductions in ϵ_λ s and time costs using our method compared to results using Go-ICP in Fig. 14

	Dinosaur	Chef	Armadillo
ϵ_λ reduced	29.6% / 50.0%	83.5% / 95.8%	62.6% / 80.7%
time reduced	58.3% / 83.9%	51.0% / 70.4%	64.4% / 79.0%

Afterwards, we compare JRMPC [1], Go-ICP [29] and our method at building a 3D coherent model from six partial scans of our laboratory [50]. The six scans, shown in Fig. 15, are recorded consecutively. Each scan has more than 110,000 points, and we downsample it to 2000, 4000, 6000, and 8000 points, separately, to create four test cases. The

transformations from [50] are taken as the ground truth. In each test case, we run the groupwise method JRMPC two times using 1000 and 2000 components, separately, while we perform Go-ICP and our method to align the six scans in sequence, which means that Scans 1 and 2 are aligned first, and then, Scans 2 and 3 are aligned, ..., and the last pair is Scans 5 and 6. Finally, the six scans are merged based on the transformations of the five pairs. For each pair, the two scans partially overlap each other, and we apply Go-ICP and our method two separate times to register them, where one applies trimming with $\xi = 10\%$ and the other applies no trimming. The results are shown in Fig. 16. JRMPC fails to align the six scans in all the test cases and Go-ICP fails in some of the tests, while our method gives correct alignments in all the test cases with the highest accuracy and the lowest time cost. Note that for most of the test cases, Go-ICP needs to carry out its BnB-based global search to find the solution, while our method solely uses the gradient-based algorithm for the optimization in all the

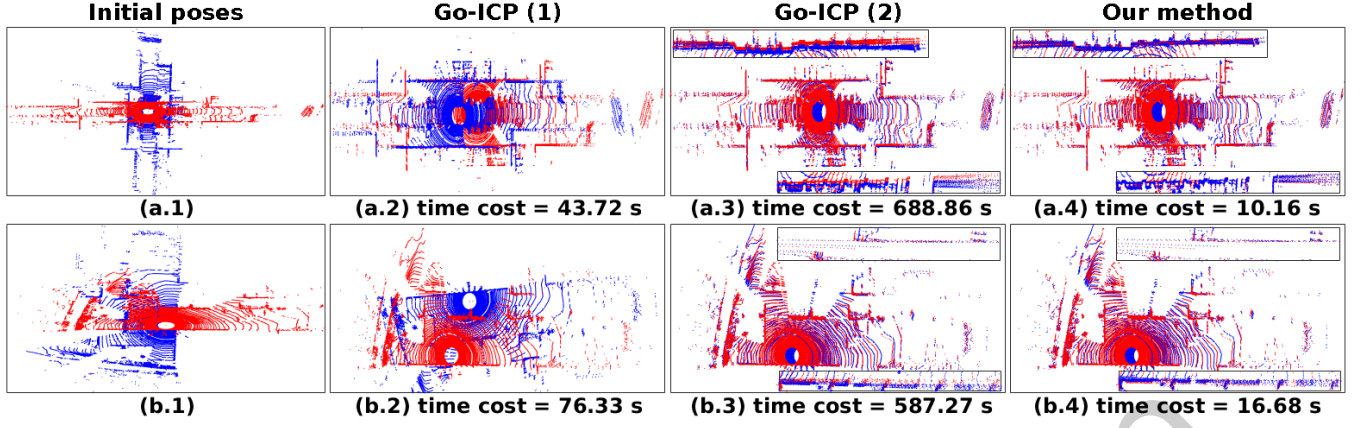


Fig. 17. Registration results of partially overlapping outdoor scans. (a.1) and (b.1) show the scans' initial poses. (a.2) and (b.2) show the results of Go-ICP using $\varepsilon = 0.001 \times N_{P_M}$. (a.3) and (b.3) show the results of Go-ICP using $\varepsilon = 0.0001 \times N_{P_M}$. (a.4) and (b.4) are the results of our method. Based on the fuzzy clusters and the metric of our method, we have ρ_λ of each λ in (a.2)-(a.4) and (b.2)-(b.4) derived by Go-ICP or our method: (a.2). $\rho_\lambda = 2.1107 > 1$; (a.3). $\rho_\lambda = 0.6173 < 1$; (a.4). $\rho_\lambda = 0.7367 < 1$; (b.2). $\rho_\lambda = 2.2529 > 1$; (b.3). $\rho_\lambda = 0.6687 < 1$; (b.4). $\rho_\lambda = 0.6364 < 1$.

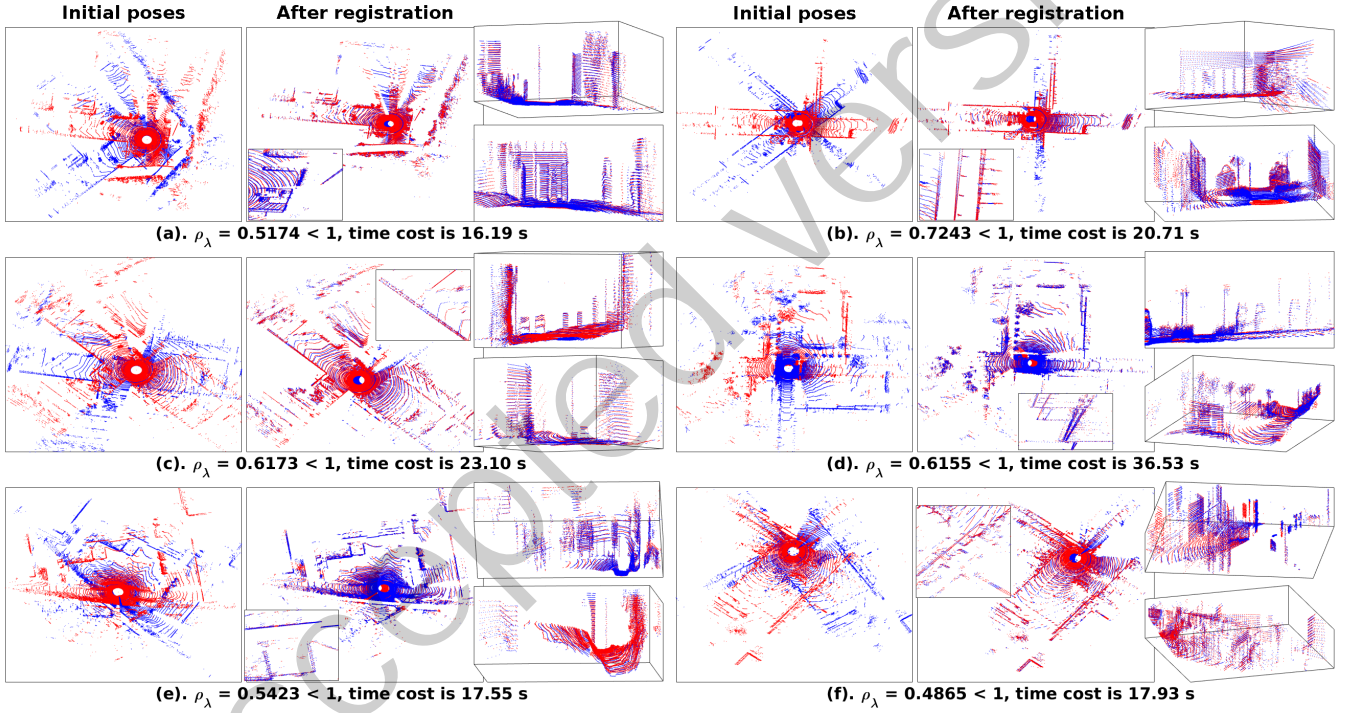


Fig. 18. The registration results of six partially overlapping outdoor scan pairs using the proposed method.

tests. The reason is because in each coarse registration of our method, the registration quality assessment detects that the solution derived by the first local convergence is already satisfactory and then stops the search before using BnB. All of the final ρ_λ s of our method are less than 1. Based on the transformations derived by our method, the six scans are merged together to build a coherent 3D model, as shown in Fig. 15.

The reasons for JRMPC's failure are that its narrow convergence range cannot handle the pose differences of these scans, and it does not have global optimization. Although our method does not simultaneously register multiple scans, it can provide a very satisfactory result in building a coherent 3D model by aligning the scans in sequence and

can treat the pose differences of these scans because of the broad convergence range of the fuzzy cluster-based metric. Moreover, our method has a global optimization scheme that can align the point sets with any initial poses. In addition, unlike JRMPC whose computation time increases linearly with the number of points, the time cost of our method is less affected by the number of points. For Go-ICP, the differences in its performance between trimming and no trimming are very large due to its sensitivity to outliers. Unlike those in Fig. 14, the points of these six scans are not evenly distributed, and the same object may be described by the points with different densities in different scans, which can introduce uncertainties into the registration. In addition, the scanner device leaves a relatively large number

of points of itself in each scan, as shown in Fig. 15, which can adversely affect the performance of Go-ICP, since it uses binary point-to-point correspondence. In contrast, our method is robust to these uncertainties and yields accurate results with and without trimming ($\epsilon_\lambda < 0.005$ for most of the cases, and those with trimming give smaller ϵ_λ s than their no-trimming counterparts do, as illustrated in Fig. 16).

Next, a number of outdoor range scans, recorded by Pellenz and Lang in the University of Koblenz-Landau, Germany and publicly available in [50], are employed for tests. The ground truth transformations for these scans are unavailable. We enlarge some parts of the registered scans to show the accuracy. Two pairs of partially overlapping scans are selected, as shown in Fig. 17, and each scan is downsampled to approximately 5000 points to perform Go-ICP and our method. The two scans of each pair are recorded consecutively and with a high degree of overlap, and we apply no trimming when registering them. Fig. 17 shows that Go-ICP with $\varepsilon = 0.001 \times N_{P_M}$ yields erroneous results. Then, using a smaller threshold $\varepsilon = 0.0001 \times N_{P_M}$, it achieves reasonable results but takes a very long time (688.86 s and 587.27 s). Compared to Go-ICP, our method accomplishes the tasks much faster (10.16 s and 16.68 s) and reduces the time costs by 98.53% and 97.16%, respectively. In addition, according to the enlarged figures shown in Fig. 17, our method yields more precise alignments for the two pairs. Furthermore, based on the fuzzy clusters and the metric of our method, we calculate ρ_λ of each solution λ derived by Go-ICP or our method in Fig. 17. As shown in the caption of Fig. 17, the erroneous solutions of (a.2) and (b.2) derived by Go-ICP have $\rho_\lambda > 1$, and the remaining ones have $\rho_\lambda < 1$. These results indicate that Go-ICP does not know how small its metric is to provide an accurate result and cannot determine whether its final solution is correct without ground truth or user intervention. In contrast, our method includes a reliable registration quality assessment that can automatically evaluate whether a solution is good or not. Therefore, our method is more suitable to be applied in autonomous devices and systems.

For the final test, our method is applied to register six other pairs of partially overlapping outdoor scans, as shown in Fig. 18. We downsample each scan to approximately 5000 points to perform the registration for efficiency, and no trimming is applied. The time costs are between 16.19 and 36.53 s, and the enlarged figures in Fig. 18 show that each pair is correctly aligned. In addition, all the global searches of our method for these six cases are stopped by the registration quality assessment, and the final ρ_λ s are less than 1.

6 CONCLUSION

This paper proposes a novel point set registration method for 3D range scan alignment. The main contributions include the following: First, we use fuzzy clusters to represent each of the two given scans and register them by minimizing a fuzzy weighted sum of the distances between their fuzzy cluster centers. This fuzzy cluster-based metric has a broad basin of convergence and provides analytic gradients such that standard gradient-based algorithms can be applied for

optimization. Second, based on this metric, simple and effective strategies to treat outliers are developed. Furthermore, for the first time in rigid point set registration, a registration quality assessment in the absence of ground truth is provided, which is a key innovation of this study. Finally, an optimization scheme integrating the BnB and gradient-based algorithms is designed; it can locate the global minimum of the metric regardless of the initialization. This global optimization scheme is performed in an effective coarse-to-fine way, where registration quality assessment is applied to boost the computational efficiency. Different types of range scans are employed to demonstrate the accuracy, robustness, effectiveness and superiority of the proposed method relative to that of several state-of-the-art approaches. In future work, we plan to investigate groupwise registration based on fuzzy clusters, which simultaneously registers multiple scans using efficient global optimization.

ACKNOWLEDGMENTS

This work was supported by the Semantic Robots Research Profile, funded by the Swedish Knowledge Foundation (KKS). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

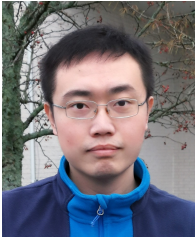
REFERENCES

- [1] Georgios Dimitrios Evangelidis and Radu Horaud. Joint alignment of multiple point sets with batch and incremental expectation-maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1397–1410, 2018.
- [2] Andrew W Fitzgibbon. Robust registration of 2d and 3d point sets. *Image and Vision Computing*, 21(13-14):1145–1153, 2003.
- [3] Guy Medan, N Shamul, and Leo Joskowicz. Sparse 3d radon space rigid registration of ct scans: Method and validation study. *IEEE Transactions on Medical Imaging*, 36(2):497–506, 2017.
- [4] Mark P Wachowiak, Renata Smolíková, Yufeng Zheng, Jacek M Zurada, Adel Said Elmaghraby, et al. An approach to multimodal biomedical image registration utilizing particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):289–301, 2004.
- [5] Andreas Nüchter, Kai Lingemann, Joachim Hertzberg, and Hartmut Surmann. 6d slam—3d mapping outdoor environments. *Journal of Field Robotics*, 24(8-9):699–722, 2007.
- [6] Todor Stoyanov, Jari Saarinen, Henrik Andreasson, and Achim J Lilienthal. Normal distributions transform occupancy map fusion: Simultaneous mapping and tracking in large scale dynamic environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4702–4708. IEEE, 2013.
- [7] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256, 1992.
- [8] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-icp. In *Robotics: science and systems*, volume 2, page 435, 2009.
- [9] Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Sparse iterative closest point. In *Proceedings of the Eleventh Eurographics/ACMSIGGRAPH Symposium on Geometry Processing*, pages 113–123. Eurographics Association, 2013.
- [10] Anand Rangarajan, Haili Chui, Eric Mjolsness, Suguna Pappu, Lila Davachi, Patricia Goldman-Rakic, and James Duncan. A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis*, 1(4):379–398, 1997.
- [11] Yanghai Tsin and Takeo Kanade. A correlation-based approach to robust point set registration. In *European Conference on Computer Vision*, pages 558–569. Springer, 2004.
- [12] Bing Jian and Baba C Vemuri. Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633–1645, 2011.

- [13] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, 2010.
- [14] Martin Magnusson, Achim Lilienthal, and Tom Duckett. Scan registration for autonomous mining vehicles using 3d-ndt. *Journal of Field Robotics*, 24(10):803–827, 2007.
- [15] Todor Stoyanov, Martin Magnusson, Henrik Andreasson, and Achim J Lilienthal. Fast and accurate scan registration through minimization of the distance between compact 3d ndt representations. *The International Journal of Robotics Research*, 31(12):1377–1393, 2012.
- [16] Luis G Sanchez Giraldo, Erion Hasanbelliu, Murali Rao, and Jose C Principe. Group-wise point-set registration based on renyi's second order entropy. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2454–2462. IEEE, 2017.
- [17] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7163–7172, 2019.
- [18] Zan Gojic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5545–5554, 2019.
- [19] Zi Jian Yew and Gim Hee Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *European Conference on Computer Vision*, pages 630–646. Springer, 2018.
- [20] Gil Elbaz, Tamar Avraham, and Anath Fischer. 3d point cloud registration for localization using a deep neural network auto-encoder. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4631–4640, 2017.
- [21] Luca Lucchese, Gianfranco Doretto, and Guido M. Cortelazzo. A frequency domain technique for range data registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(11):1468–1484, 2002.
- [22] Nicholas J Tustison, Suyash P Awate, Gang Song, Tessa S Cook, and James C Gee. Point set registration using havrda-charvat-tsalis entropy measures. *IEEE Transactions on Medical Imaging*, 30(2):451–460, 2011.
- [23] Sébastien Granger and Xavier Pennec. Multi-scale em-icp: A fast and robust approach for surface registration. In *European Conference on Computer Vision*, pages 418–432. Springer, 2002.
- [24] Luciano Silva, Olga Regina Pereira Bellon, and Kim L Boyer. Precision range image registration using a robust surface interpenetration measure and enhanced genetic algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):762–776, 2005.
- [25] Romeil Sandhu, Samuel Dambreville, and Allen Tannenbaum. Point set registration via particle filtering and stochastic dynamics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1459–1473, 2010.
- [26] Chavdar Papazov and Darius Burschka. Stochastic global optimization for robust point set registration. *Computer Vision and Image Understanding*, 115(12):1598–1609, 2011.
- [27] Hongdong Li and Richard Hartley. The 3d-3d registration problem revisited. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.
- [28] Carl Olsson, Fredrik Kahl, and Magnus Oskarsson. Branch-and-bound methods for euclidean registration problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):783–794, 2009.
- [29] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: a globally optimal solution to 3d icp point-set registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2241–2254, 2016.
- [30] Dmitry Chetverikov, Dmitry Stepanov, and Pavel Krsek. Robust euclidean alignment of 3d point sets: the trimmed iterative closest point algorithm. *Image and Vision Computing*, 23(3):299–309, 2005.
- [31] Will Chang and Matthias Zwicker. Automatic registration for articulated shapes. In *Computer Graphics Forum*, volume 27, pages 1459–1468. Wiley Online Library, 2008.
- [32] Ran Gal and Daniel Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Transactions on Graphics (TOG)*, 25(1):130–150, 2006.
- [33] Alain Pitiot, Hervé Delingette, and Paul M Thompson. Learning shape correspondence for nd curves. *International Journal of Computer Vision*, 71(1):71–88, 2007.
- [34] Henrik Andreasson and Achim Lilienthal. Vision aided 3d laser scanner based registration. In *3rd European Conference on Mobile Robots, ECMR'07, Freiburg, Germany, September 19-21, 2007*, pages 192–197, 2007.
- [35] Gary KL Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C Langbein, Yonghuai Liu, David Marshall, Ralph R Martin, Xian-Fang Sun, and Paul L Rosin. Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217, 2013.
- [36] Baraka Maiseli, Yanfeng Gu, and Huijun Gao. Recent developments and trends in point set registration methods. *Journal of Visual Communication and Image Representation*, 46:95–106, 2017.
- [37] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992.
- [38] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings 3rd International Conference on 3-D Digital Imaging and Modeling*, pages 145–152. IEEE, 2001.
- [39] Craig Robertson and Robert B Fisher. Parallel evolutionary registration of range data. *Computer Vision and Image Understanding*, 87(1-3):39–50, 2002.
- [40] James C Bezdek, Robert Ehrlich, and William Full. Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203, 1984.
- [41] Jiang-She Zhang and Yiu-Wing Leung. Robust clustering by pruning outliers. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 33(6):983–998, 2003.
- [42] Joaquim Salvi, Carles Matabosch, David Fofi, and Josep Forest. A review of recent range image registration methods with accuracy evaluation. *Image and Vision computing*, 25(5):578–596, 2007.
- [43] Qian-Fang Liao and Da Sun. Interaction measures for control configuration selection based on interval type-2 takagi-sugeno fuzzy model. *IEEE Transactions on Fuzzy Systems*, 26(5):2510–2523, 2018.
- [44] Qianfang Liao and Da Sun. Sparse and decoupling control strategies based on takagi-sugeno fuzzy models. *IEEE Transactions on Cybernetics*, 2019.
- [45] Da Sun, Qianfang Liao, Todor Stoyanov, Andrey Kiselev, and Amy Loutfi. Bilateral telerobotic system using type-2 fuzzy neural network based moving horizon estimation force observer for enhancement of environmental force compliance and human perception. *Automatica*, 106:358–373, 2019.
- [46] Da Sun, Andrey Kiselev, Qianfang Liao, Todor Stoyanov, and Amy Loutfi. A new mixed-reality-based teleoperation system for telepresence and maneuverability enhancement. *IEEE Transactions on Human-Machine Systems*, 50(1):55–67, 2020.
- [47] Stanford Computer Graphics Laboratory. The stanford 3d scanning repository. <http://graphics.stanford.edu/data/3Dscanrep/>.
- [48] Shihao Wu, Peter Bertholet, Hui Huang, Daniel Cohen-Or, Minglun Gong, and Matthias Zwicker. Structure-aware data consolidation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2529–2537, 2018.
- [49] Roger Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2013.
- [50] Robotic 3D Scan Repository. <http://kos.informatik.uni-osnabrueck.de/3Dscans/>.
- [51] Greg Turk and Brendan Mullins. Large geometric models archive at georgia tech. https://www.cc.gatech.edu/projects/large_models/.
- [52] Ajmal S Mian, Mohammed Bennamoun, and Robyn Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1584–1601, 2006.



Qianfang Liao received a B.Eng. degree in automation from Wuhan University, Wuhan, China, in 2006, an M.Eng. degree in automation from Shanghai Jiao Tong University, Shanghai, China, in 2009, and a Ph.D. from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2015. Currently, she is a researcher with Örebro University, Örebro, Sweden. Her research interests include fuzzy modeling, control theory, and computer vision.



Da Sun received B.Eng. and Ph.D. degrees in mechatronics from the University of Wollongong, Wollongong, NSW, Australia, in 2012 and 2016, respectively. From 2016 to 2017, he was a Research Fellow with the National University of Singapore, Singapore. He is currently a researcher with Örebro University, Örebro, Sweden. His research interests include learning, vision and control of telerobotics.



Henrik Andreasson received a Master's degree in mechatronics from the Royal Institute of Technology, Stockholm, Sweden, in 2001 and a Ph.D. in computer science from Örebro University, Örebro, Sweden, in 2008. He is currently an associate professor with the Center for Applied Autonomous Sensor Systems (AASS), Örebro University, Örebro, Sweden. His research interests include mobile robotics, computer vision, and machine learning.

Supplementary material

1. Proof for Theorem 1

The validity of the per-element upper bound in Eq. (11) is apparent: $J(\lambda_0, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ must be larger than the minimal $J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ in $\mathcal{C}_r \times \mathcal{C}_t$. For the per-element lower bound in Eq. (12), a detailed explanation is needed. Since $m > 1$ in Eqs. (9) and (10), we know that $J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ will become smaller when all $D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})$ s, $i = 1, \dots, N_{CF}$, decrease. On the other hand, according to **Lemma 1**, if the uncertainty sphere of $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ does not enclose any \mathbf{c}_{Fi} , the minimal value of $D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})$ will be achieved when $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ is placed at the point on the sphere's surface which is nearest to \mathbf{c}_{Fi} , and this minimal value is $D(\lambda_0, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - (\gamma_{rj} + \gamma_t)$, as illustrated in Fig. S1. Therefore, for the distance between $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ and each \mathbf{c}_{Fi} , we have:

$$D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) \geq D(\lambda_0, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - (\gamma_{rj} + \gamma_t) \quad (\text{S.1.1})$$

Subsequently, for the per-element distance loss $J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)$, we have:

$$\begin{aligned} J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) &= \left(\sum_{i=1}^{N_{CF}} D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})^{\frac{2}{1-m}} \right)^{1-m} \geq \left(\sum_{i=1}^{N_{CF}} \left(D(\lambda_0, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - (\gamma_{rj} + \gamma_t) \right)^{\frac{2}{1-m}} \right)^{1-m} \\ &= J_{low}(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) \end{aligned} \quad (\text{S.1.2})$$

If any \mathbf{c}_{Fi} is within or on the surface of the uncertainty sphere, which means $D(\lambda_0, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) - (\gamma_{rj} + \gamma_t) \leq 0$, then the minimal value of $D(\lambda, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})$ is 0. In this case, $J_{low}(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) = 0$. In Eq. (13), the upper and lower bounds of the fuzzy cluster-based metric $J(\lambda, \mathbb{C}_M, \mathbb{C}_F)$ are corollaries of Eq. (11) and (12), respectively.

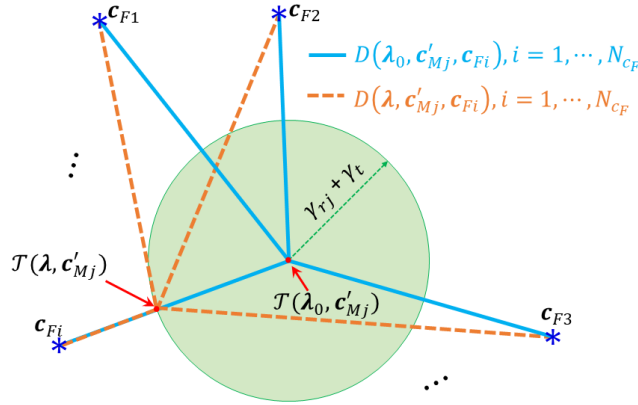


Figure S1. The distances between a $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ and all \mathbf{c}_{Fi} s in an uncertainty sphere (the green ball) of radius $(\gamma_{rj} + \gamma_t)$ centered at $\mathcal{T}(\lambda_0, \mathbf{c}'_{Mj})$. Each * denotes a \mathbf{c}_{Fi} , and no \mathbf{c}_{Fi} is enclosed in this sphere.

2. Analysis of AFCCD and AFPCD when two point sets are registered

We present an intuitive analysis based on mathematical functions to show that the two point sets \mathbb{P}_F and $\mathcal{T}(\lambda, \mathbb{P}_M)$, which have overlaps, can be considered as aligned or coarsely aligned if they satisfy:

$$\rho_\lambda = \frac{\text{AFCCD}}{\text{AFPCD}} = \frac{\bar{J}(\lambda, \mathbb{C}'_M, \mathbb{C}_F)}{\bar{J}(\mathbb{P}_F, \mathbb{C}_F)} \leq 1$$

where λ is the transformation; \mathbb{C}_F and \mathbb{C}_M are the fuzzy cluster centers of \mathbb{P}_F and \mathbb{P}_M , respectively; and $\mathcal{T}(\lambda, \mathbb{C}'_M)$ denotes the fuzzy cluster centers of the transformed moving set $\mathcal{T}(\lambda, \mathbb{P}_M)$ in the overlapping region.

For the fixed set $\mathbb{P}_F = \{\mathbf{p}_{Fj}, j = 1, 2, \dots, N_{PF}\}$, its fuzzy cluster centers $\mathbb{C}_F = \{\mathbf{c}_{Fi}, i = 1, 2, \dots, N_{CF}\}$ is derived by minimizing the following fuzzy c-means objective function:

$$\begin{aligned} \min_{\mathbb{C}_F} J(\mathbb{P}_F, \mathbb{C}_F) &= \min_{\mathbb{C}} \sum_{j=1}^{N_P} \sum_{i=1}^{N_C} \mu_{\mathbf{c}_{Fi}}(\mathbf{p}_{Fj})^m \cdot \|\mathbf{p}_{Fj} - \mathbf{c}_{Fi}\|^2 \\ &= \min_{\mathbb{C}_F} \sum_{j=1}^{N_{PF}} \frac{1}{\left(\frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{F1}\|^{\frac{2}{m-1}}} + \frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{F2}\|^{\frac{2}{m-1}}} + \dots + \frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{FN_{CF}}\|^{\frac{2}{m-1}}} \right)^{m-1}} \end{aligned} \quad (\text{S.2.1})$$

where $m > 1$. In our study, we choose $m = 2$, which is a widely used choice. Then Eq. (S.2.1) becomes:

$$\begin{aligned} \min_{\mathbb{C}_F} J(\mathbb{P}_F, \mathbb{C}_F) &= \min_{\mathbb{C}_F} \sum_{j=1}^{N_{PF}} J(\mathbf{p}_{Fj}, \mathbb{C}_F) \\ &= \min_{\mathbb{C}_F} \sum_{j=1}^{N_{PF}} \frac{1}{\frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{F1}\|^2} + \frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{F2}\|^2} + \dots + \frac{1}{\|\mathbf{p}_{Fj} - \mathbf{c}_{FN_{CF}}\|^2}} \end{aligned} \quad (\text{S.2.2})$$

where $J(\mathbf{p}_{Fj}, \mathbb{C}_F) = 1 / \left(1 / \|\mathbf{p}_{Fj} - \mathbf{c}_{F1}\|^2 + 1 / \|\mathbf{p}_{Fj} - \mathbf{c}_{F2}\|^2 + \dots + 1 / \|\mathbf{p}_{Fj} - \mathbf{c}_{FN_{CF}}\|^2 \right)$ is the per-point distance loss in $J(\mathbb{P}_F, \mathbb{C}_F)$. According to Eq. (S.2.2), $J(\mathbb{P}_F, \mathbb{C}_F)$ will be minimized if all $\|\mathbf{p}_{Fj} - \mathbf{c}_{Fi}\|$ s ($i = 1, \dots, N_{CF}, j = 1, \dots, N_{PF}$) are as small as possible. As a result, each fuzzy cluster center \mathbf{c}_{Fi} should be close to as many as possible points, and thus a \mathbf{c}_{Fi} is located at the most representative position in its local area. Similar phenomenon can be found with \mathbb{P}_M and \mathbb{C}_M . The AFPCD of \mathbb{P}_F , which is the average per-point distance loss with respect to \mathbb{C}_F , is calculated as:

$$\text{AFPCD} = \bar{J}(\mathbb{P}_F, \mathbb{C}_F) = \frac{J(\mathbb{P}_F, \mathbb{C}_F)}{N_{PF}} = \frac{\sum_{j=1}^{N_{PF}} J(\mathbf{p}_{Fj}, \mathbb{C}_F)}{N_{PF}} \quad (\text{S.2.3})$$

AFPCD in Eq. (S.2.3) describes the points' dispersion and disposition with respect to \mathbb{C}_F . When a point \mathbf{p}_{Fj} is near a fuzzy cluster center \mathbf{c}_{Fi} , its distance loss $J(\mathbf{p}_{Fj}, \mathbb{C}_F)$ is smaller than AFPCD; and

when \mathbf{p}_{Fj} is relatively not close to any fuzzy cluster centers, its distance loss $J(\mathbf{p}_{Fj}, \mathbb{C}_F)$ is larger than AFPCD.

Suppose \mathbb{P}_F and \mathbb{P}_M are aligned by $\mathcal{T}(\lambda)$. In the overlapping region, the points of \mathbb{P}_F and $\mathcal{T}(\lambda, \mathbb{P}_M)$ describe the same spatial properties of the object, and thus they have the same geometric properties. We denote the points of \mathbb{P}_F in the overlapping region as $\mathbb{P}'_F = \{\mathbf{p}'_{Fj}, j = 1, 2, \dots, N_{P'_F}\}$, and \mathbb{P}'_F can also be used to represent the points of $\mathcal{T}(\lambda, \mathbb{P}_M)$ in this region. According to the analysis of fuzzy clustering stated above, in the overlapping region, \mathbf{c}_{Fi} s and $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ s are located at the representative positions, which are close to as many as possible points of \mathbb{P}'_F . Consequently, we can know that the following $J(\mathbb{P}'_F, \mathbb{C}_F)$ and $J(\mathbb{P}', \mathbb{C}'_M)$ are minimized values or close to the minimized values:

$$J(\mathbb{P}'_F, \mathbb{C}_F) = \sum_{j=1}^{N_{P'_F}} J(\mathbf{p}'_{Fj}, \mathbb{C}_F) = \sum_{j=1}^{N_{P'_F}} \frac{1}{\frac{1}{\|\mathbf{p}'_{Fj} - \mathbf{c}_{F1}\|^2} + \frac{1}{\|\mathbf{p}'_{Fj} - \mathbf{c}_{F2}\|^2} + \dots + \frac{1}{\|\mathbf{p}'_{Fj} - \mathbf{c}_{FN_{C_F}}\|^2}} \quad (\text{S.2.4})$$

$$\begin{aligned} J(\mathbb{P}'_F, \mathcal{T}(\lambda, \mathbb{C}'_M)) &= \sum_{j=1}^{N_{P'_F}} J(\mathbf{p}'_{Fj}, \mathcal{T}(\lambda, \mathbb{C}'_M)) \\ &= \sum_{j=1}^{N_{P'_F}} \frac{1}{\frac{1}{\|\mathbf{p}'_{Fj} - \mathcal{T}(\lambda, \mathbf{c}'_{M1})\|^2} + \frac{1}{\|\mathbf{p}'_{Fj} - \mathcal{T}(\lambda, \mathbf{c}'_{M2})\|^2} + \dots + \frac{1}{\|\mathbf{p}'_{Fj} - \mathcal{T}(\lambda, \mathbf{c}'_{MN_{C'_M}})\|^2}} \end{aligned} \quad (\text{S.2.5})$$

Based on the above discussion and equations, we can infer that a transformed center $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ is likely to be close to a certain \mathbf{c}_{Fi} in the overlapping region. Consequently, the distance loss of a $\mathcal{T}(\lambda, \mathbf{c}'_{Mj})$ with respect to \mathbb{C}_F , expressed as follows, is prone to being smaller than the AFPCD.

$$J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F) = \frac{1}{\frac{1}{\|\mathcal{T}(\lambda, \mathbf{c}'_{Mj}) - \mathbf{c}_{F1}\|^2} + \frac{1}{\|\mathcal{T}(\lambda, \mathbf{c}'_{Mj}) - \mathbf{c}_{F2}\|^2} + \dots + \frac{1}{\|\mathcal{T}(\lambda, \mathbf{c}'_{Mj}) - \mathbf{c}_{FN_{C_F}}\|^2}} \quad (\text{S.2.6})$$

As a result, the AFCCD $\bar{J}(\lambda, \mathbb{C}'_M, \mathbb{C}_F)$, which is the average per-element distance loss of $\mathcal{T}(\lambda, \mathbb{C}'_M)$ with respect to \mathbb{C}_F , is smaller than the AFPCD $\bar{J}(\mathbb{P}_F, \mathbb{C}_F)$:

$$\text{AFCCD} = \bar{J}(\lambda, \mathbb{C}'_M, \mathbb{C}_F) = \frac{\sum_{j=1}^{N_{C'_M}} J(\lambda, \mathbf{c}'_{Mj}, \mathbb{C}_F)}{N_{C'_M}} \leq \frac{\sum_{j=1}^{N_{P_F}} J(\mathbf{p}_{Fj}, \mathbb{C}_F)}{N_{P_F}} = \bar{J}(\mathbb{P}_F, \mathbb{C}_F) = \text{AFPCD}$$

In the **Remark 2** of the manuscript, we state that the registration quality assessment may be invalid in the following four cases:

- i). The overlapping ratios of the two point sets are too small;
- ii). The size difference between the two point sets is too large;
- iii). $N_{C'_M}$ is chosen to be too small;

iv). The trimming ratio ξ for partial overlap registration is chosen to be too large.

We can ignore cases **i)** and **ii)** in our study since the proposed method works well for the point sets satisfying **Assumption 1**. For case **iii)**, according to our experimental experience, a generally safe choice is $N_{\mathcal{C}'_M} \geq 50$. For the problem of case **iv)**, we have presented a discussion about the selection of ξ at the end of Section 4.4.3.

3. The gradient of the fuzzy cluster-based metric

The fuzzy cluster-based metric with $m = 2$ for registration is expressed as follows:

$$J(\boldsymbol{\lambda}, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} \sum_{i=1}^{N_{C_F}} (\mu_{c_{Fi}}(\mathcal{T}(\boldsymbol{\lambda}, \mathbf{c}'_{Mj})))^2 \cdot (D(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}))^2 = \sum_{j=1}^{N_{C'_M}} J(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbb{C}_F) \quad (\text{S.3.1})$$

where

$$J(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbb{C}_F) = \begin{cases} 0, & \text{if } \forall_{i=1, \dots, N_{C_F}} D(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}) = 0 \\ \frac{1}{\sum_{i=1}^{N_{C_F}} \frac{1}{(D(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi}))^2}}, & \text{else} \end{cases} \quad (\text{S.3.2})$$

The gradient of $J(\boldsymbol{\lambda}, \mathbb{C}'_M, \mathbb{C}_F)$ with respect to $\boldsymbol{\lambda}$ is:

$$\nabla_{\boldsymbol{\lambda}} J(\boldsymbol{\lambda}, \mathbb{C}'_M, \mathbb{C}_F) = \sum_{j=1}^{N_{C'_M}} \frac{\partial J(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbb{C}_F)}{\partial \boldsymbol{\lambda}} \in \mathbb{R}^6 \quad (\text{S.3.3})$$

For simplicity, we denote $J(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbb{C}_F)$ as J_j , and $D(\boldsymbol{\lambda}, \mathbf{c}'_{Mj}, \mathbf{c}_{Fi})$ as D_{ji} . When D_{ji} s, $i = 1, \dots, N_{C_F}$, are not 0, we have

$$\frac{\partial J_j}{\partial \boldsymbol{\lambda}} = \frac{\partial}{\partial \boldsymbol{\lambda}} \left(\sum_{i=1}^{N_{C_F}} \frac{1}{D_{ji}^2} \right)^{-1} = \left(\sum_{i=1}^{N_{C_F}} \frac{1}{D_{ji}^2} \right)^{-2} \cdot \left(\sum_{i=1}^{N_{C_F}} (D_{ji}^2)^{-2} \cdot \frac{\partial (D_{ji}^2)}{\partial \boldsymbol{\lambda}} \right) = J_j^2 \cdot \left(\sum_{i=1}^{N_{C_F}} (D_{ji}^2)^{-2} \cdot \frac{\partial (D_{ji}^2)}{\partial \boldsymbol{\lambda}} \right) \quad (\text{S.3.4})$$

where $\partial(D_{ji}^2)/\partial \boldsymbol{\lambda} \in \mathbb{R}^6$ is calculated by:

$$\frac{\partial (D_{ji}^2)}{\partial \boldsymbol{\lambda}} = \frac{\partial (\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t} - \mathbf{c}_{Fj})^T \cdot (\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t} - \mathbf{c}_{Fj})}{\partial \boldsymbol{\lambda}} = 2 \cdot \frac{\partial (\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t})^T}{\partial \boldsymbol{\lambda}} \cdot (\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t} - \mathbf{c}_{Fj}) \quad (\text{S.3.5})$$

where $\mathbf{r} = [r_x \ r_y \ r_z]^T$ and $\mathbf{t} = [t_x \ t_y \ t_z]^T$. Note that \mathbf{r} uses axis-angle representation, where the axis is $\mathbf{r}/\|\mathbf{r}\|$ and the angle θ is $\|\mathbf{r}\|$. The relationship between \mathbf{r} and the rotation matrix $\mathcal{R}(\mathbf{r})$ is:

$$\mathcal{R}(\mathbf{r}) = \mathbf{I}_3 + \frac{M_r \cdot \sin \theta}{\theta} + \frac{(M_r)^2 \cdot (1 - \cos \theta)}{\theta^2}$$

where $M_r = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}$. For example, if the axis is $[1, 0, 0]^T$ and the rotation angle is $\theta = \pi$,

then $\mathbf{r} = [\pi, 0, 0]^T$. By substituting \mathbf{r} to M_r and $\mathcal{R}(\mathbf{r})$, we have the following result:

$$M_r = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\pi \\ 0 & \pi & 0 \end{bmatrix}, \text{ and } \mathcal{R}(\mathbf{r}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

We denote $\mathbf{c}'_{Mj} = [\mathbf{c}'_{Mj}(x) \ \mathbf{c}'_{Mj}(y) \ \mathbf{c}'_{Mj}(z)]^T$, and for simplicity, we represent $\cos \theta$ and $\sin \theta$ by $c\theta$ and $s\theta$, respectively. From the relationship between \mathbf{r} and $\mathcal{R}(\mathbf{r})$, we have:

$$\begin{aligned}
& \mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t} \\
&= \begin{bmatrix} c\theta + \frac{r_x r_x}{\theta^2} (1 - c\theta) & -\frac{r_z}{\theta} s\theta + \frac{r_x r_y}{\theta^2} (1 - c\theta) & \frac{r_y}{\theta} s\theta + \frac{r_x r_z}{\theta^2} (1 - c\theta) \\ \frac{r_z}{\theta} s\theta + \frac{r_y r_x}{\theta^2} (1 - c\theta) & c\theta + \frac{r_y r_y}{\theta^2} (1 - c\theta) & -\frac{r_x}{\theta} s\theta + \frac{r_y r_z}{\theta^2} (1 - c\theta) \\ -\frac{r_y}{\theta} s\theta + \frac{r_z r_x}{\theta^2} (1 - c\theta) & \frac{r_x}{\theta} s\theta + \frac{r_z r_y}{\theta^2} (1 - c\theta) & c\theta + \frac{r_z r_z}{\theta^2} (1 - c\theta) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{c}'_{Mj}(x) \\ \mathbf{c}'_{Mj}(y) \\ \mathbf{c}'_{Mj}(z) \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \\
&= \begin{bmatrix} \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x \\ \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y \\ \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_z \end{bmatrix} \tag{S.3.6}
\end{aligned}$$

where

$$\begin{aligned}
\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x &= \mathbf{c}'_{Mj}(x) \cdot c\theta + (\mathbf{c}'_{Mj}(z) \cdot r_y - \mathbf{c}'_{Mj}(y) \cdot r_z) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(x) \cdot r_x r_x + \mathbf{c}'_{Mj}(y) \cdot r_x r_y + \mathbf{c}'_{Mj}(z) \cdot r_x r_z) \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + t_x
\end{aligned}$$

$$\begin{aligned}
\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y &= \mathbf{c}'_{Mj}(y) \cdot c\theta + (\mathbf{c}'_{Mj}(x) \cdot r_z - \mathbf{c}'_{Mj}(z) \cdot r_x) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(x) \cdot r_y r_x + \mathbf{c}'_{Mj}(y) \cdot r_y r_y + \mathbf{c}'_{Mj}(z) \cdot r_y r_z) \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + t_y
\end{aligned}$$

$$\begin{aligned}
\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_z &= \mathbf{c}'_{Mj}(z) \cdot c\theta + (\mathbf{c}'_{Mj}(y) \cdot r_x - \mathbf{c}'_{Mj}(x) \cdot r_y) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(x) \cdot r_z r_x + \mathbf{c}'_{Mj}(y) \cdot r_z r_y + \mathbf{c}'_{Mj}(z) \cdot r_z r_z) \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + t_z
\end{aligned}$$

Therefore, $\partial(\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t})/\partial\lambda \in \mathbb{R}^{3 \times 6}$, the 18 elements of $\partial(\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t})/\partial\lambda$ can be derived using the following equations:

$$\begin{aligned}
\frac{\partial\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial r_x} &= r_x \cdot \left(-\mathbf{c}'_{Mj}(x) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(z) \cdot r_y - \mathbf{c}'_{Mj}(y) \cdot r_z) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \\
& (2 \cdot \mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_x \cdot r_x \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \tag{S.3.7}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial r_y} &= r_y \cdot \left(-\mathbf{c}'_{Mj}(x) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(z) \cdot r_y - \mathbf{c}'_{Mj}(y) \cdot r_z) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \mathbf{c}'_{Mj}(z) \cdot \frac{s\theta}{\theta} + \\
& \mathbf{c}'_{Mj}(y) \cdot r_x \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_y \cdot r_x \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \tag{S.3.8}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial\{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial r_z} &= r_z \cdot \left(-\mathbf{c}'_{Mj}(x) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(z) \cdot r_y - \mathbf{c}'_{Mj}(y) \cdot r_z) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) - \mathbf{c}'_{Mj}(y) \cdot \frac{s\theta}{\theta} + \\
& \mathbf{c}'_{Mj}(z) \cdot r_x \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_z \cdot r_x \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \tag{S.3.9}
\end{aligned}$$

$$\frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial t_x} = 1, \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial t_y} = 0, \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_x}{\partial t_z} = 0 \quad (\text{S.3.10})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{M_j} + \mathbf{t}\}_y}{\partial r_x} &= r_x \cdot \left(-\mathbf{c}'_{M_j}(y) \cdot \frac{s\theta}{\theta} + \left(\mathbf{c}'_{M_j}(x) \cdot r_z - \mathbf{c}'_{M_j}(z) \cdot r_x \right) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) - \mathbf{c}'_{M_j}(z) \cdot \frac{s\theta}{\theta} + \\ &\mathbf{c}'_{M_j}(x) \cdot r_y \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_x \cdot r_y \cdot \left(\mathbf{c}'_{M_j}(x) \cdot r_x + \mathbf{c}'_{M_j}(y) \cdot r_y + \mathbf{c}'_{M_j}(z) \cdot r_z \right) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.11})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + t\}_y}{\partial r_y} = & r_y \cdot \left(-\mathbf{c}'_{Mj}(y) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(x) \cdot r_z - \mathbf{c}'_{Mj}(z) \cdot r_x) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \\ & (\mathbf{c}'_{Mj}(x) \cdot r_x + 2 \cdot \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_y \cdot r_y \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \\ & \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.12})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y}{\partial r_z} = & r_z \cdot \left(-\mathbf{c}'_{Mj}(y) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(x) \cdot r_z - \mathbf{c}'_{Mj}(z) \cdot r_x) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \mathbf{c}'_{Mj}(x) \cdot \frac{s\theta}{\theta} + \\ & \mathbf{c}'_{Mj}(z) \cdot r_y \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_z \cdot r_y \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.13})$$

$$\frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y}{\partial t_x} = 0, \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y}{\partial t_y} = 1, \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_y}{\partial t_z} = 0 \quad (\text{S.3.14})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(\mathbf{r}) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}}{\partial r_x} \Big|_z &= r_x \cdot \left(-\mathbf{c}'_{Mj}(z) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(y) \cdot r_x - \mathbf{c}'_{Mj}(x) \cdot r_y) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \mathbf{c}'_{Mj}(y) \cdot \frac{s\theta}{\theta} + \\ &\mathbf{c}'_{Mj}(x) \cdot r_z \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_x \cdot r_z \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.15})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(r) \cdot \mathbf{c}'_{Mj} + \mathbf{t}\}_z}{\partial r_y} &= r_y \cdot \left(-\mathbf{c}'_{Mj}(z) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(y) \cdot r_x - \mathbf{c}'_{Mj}(x) \cdot r_y) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) - \mathbf{c}'_{Mj}(x) \cdot \frac{s\theta}{\theta} + \\ &\mathbf{c}'_{Mj}(y) \cdot r_z \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_y \cdot r_z \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.16})$$

$$\begin{aligned} \frac{\partial \{\mathcal{R}(r) \cdot \mathbf{c}'_{Mj} + t\}_z}{\partial r_z} = & r_z \cdot \left(-\mathbf{c}'_{Mj}(z) \cdot \frac{s\theta}{\theta} + (\mathbf{c}'_{Mj}(y) \cdot r_x - \mathbf{c}'_{Mj}(x) \cdot r_y) \cdot \left(\frac{c\theta}{\theta^2} - \frac{1}{\theta^2} \cdot \frac{s\theta}{\theta} \right) \right) + \\ & (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + 2 \cdot \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(\frac{1}{\theta^2} - \frac{c\theta}{\theta^2} \right) + r_z \cdot r_z \cdot (\mathbf{c}'_{Mj}(x) \cdot r_x + \mathbf{c}'_{Mj}(y) \cdot r_y + \\ & \mathbf{c}'_{Mj}(z) \cdot r_z) \cdot \left(-\frac{2}{\theta^4} + \frac{1}{\theta^2} \frac{s\theta}{\theta} + \frac{2c\theta}{\theta^4} \right) \end{aligned} \quad (\text{S.3.17})$$

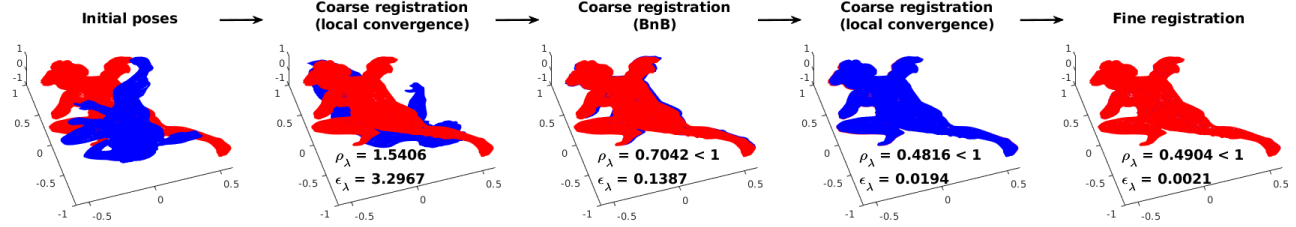
$$\frac{\partial\{\mathcal{R}(\mathbf{r})\cdot\mathbf{c}'_{Mj}+\mathbf{t}\}_z}{\partial t_x}=0, \frac{\partial\{\mathcal{R}(\mathbf{r})\cdot\mathbf{c}'_{Mj}+\mathbf{t}\}_z}{\partial t_y}=0, \frac{\partial\{\mathcal{R}(\mathbf{r})\cdot\mathbf{c}'_{Mj}+\mathbf{t}\}_z}{\partial t_z}=1$$

(S.3.18)

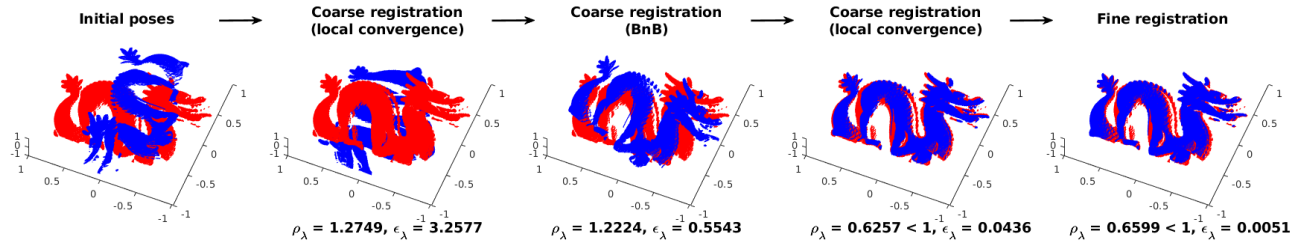
Submitting equations (S.3.7)-(S.3.18) to (S.3.4), we can have the value of $\partial J_j/\partial \boldsymbol{\lambda}$, and then sum J_j s of all the \mathbf{c}'_{Mj} s, we can have $\nabla_{\boldsymbol{\lambda}} J(\boldsymbol{\lambda}, \mathbb{C}'_M, \mathbb{C}_F)$.

4. The registration processes of the angel, dragon and happy Buddha point sets in the ablation study shown in Fig. 11 of Section 5.3

Angel:



Dragon:



Happy Buddha:

