

GraphCPD: Coherent Point Drift for Point Cloud Registration via Graph Signal Processing

Yingcheng Lai¹, Xingjian Wang¹, and Li Chai^{1*}

College of Control Science and Engineering, Zhejiang University, Hangzhou, China
{laiyingcheng, xingjianwang, chaili}@zju.edu.cn

Abstract. Probabilistic point cloud registration has attracted increasing attention due to its robustness to noise, outliers and occlusions. However, existing methods often suffer from high computational cost and neglect the role of informative priors. In this paper, we propose a new probabilistic registration method based on graph signal processing (GSP), called graph coherent point drift (GraphCPD). Specifically, we design a high-pass graph filter to extract high-frequency components, which are theoretically proven to be invariant under rigid transformations. These components are combined with point coordinates and normals to form a high-dimensional graph signal. We construct a local graph based on the graph signal and use the graph Laplacian model for registration. Compared with the classical Gaussian mixture models (GMMs), graph Laplacian models provide more discriminative geometric representations and enhance the model’s ability to capture graph structure. Furthermore, we exploit the invariance of high-frequency components to define prior probabilities, significantly reducing the corresponding search space and improving the speed of registration. Experimental results demonstrate that our method improves runtime efficiency over most existing probabilistic methods, while maintaining competitive registration accuracy, especially on large-scale point clouds. The source code is available at <https://github.com/YCLai1997/GraphCPD>.

Keywords: Point Cloud Registration · Graph Signal Processing · Coherent Point Drift

1 Introduction

Point cloud registration is a fundamental task in computer vision and 3D data processing, aiming to align two sets of point clouds and estimate the spatial transformation between them. It plays a crucial role in various applications such as SLAM [36], object pose estimation [16], dense 3D reconstruction [34] and interactive tracking of articulated [24]. However, registration remains a difficult problem due to challenges such as noise, outliers, occlusions, partial overlap and varying point densities. The Iterative Closest Point (ICP) algorithm is the

* Corresponding author.

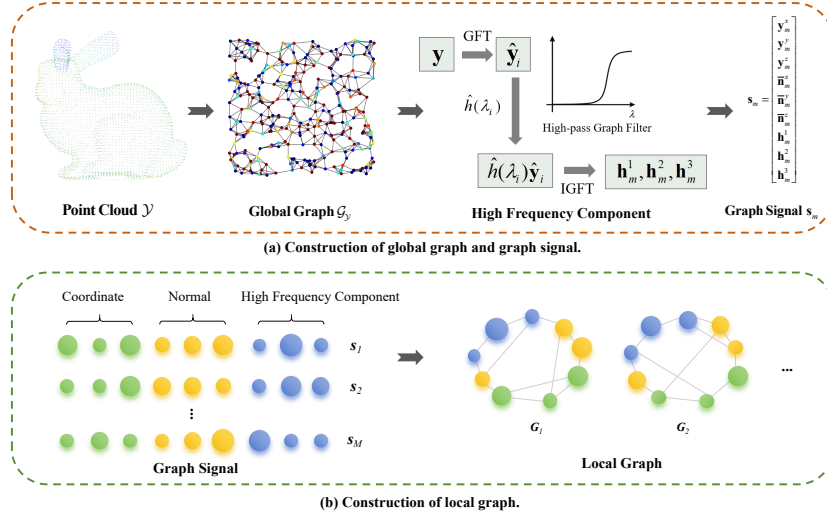


Fig. 1: An illustration of the GraphCPD method. (a) Global graph construction with multi-dimensional components as graph signals. GFT denotes graph Fourier transform and IGFT denotes inverse graph Fourier transform. (b) Local graph construction, where graph signals are smoothed on the respective graph Laplacian matrix.

most widely used approach, achieving alignment by iteratively finding nearest-neighbor correspondences between points [4, 32]. Despite its popularity, the ICP algorithm is sensitive to noise and outliers, and may be trapped in local optima [26].

Probabilistic methods model point cloud registration as a task of estimating probability density [9, 15, 33]. They typically assume that the observed data points follow a normal distribution around the model points, thereby transforming the point-to-point correspondence problem into the estimation of mixture model parameters. These approaches exhibit strong robustness by incorporating an outlier component and spherical isotropic covariance into the GMM [8, 13, 26]. However, such methods are often computationally expensive and significantly slower than ICP-based approaches, particularly when dealing with large-scale or high-resolution point clouds.

In this paper, we propose a novel probabilistic registration method integrated with GSP providing practical robustness and computational efficiency. We employ high-pass graph filtering to extract high-frequency components, effectively suppressing low-frequency signals and emphasising local geometric variations. These high-frequency components are theoretically proven to be invariant under rigid transformations, making them particularly suitable for point cloud registration. We construct a high-dimensional graph signal for each point by concatenating the extracted components with its geometric attributes (coordinates and normals). Then, a local graph is constructed for each point based on the graph signal, with edge weights encoding the similarity between different attributes.

Based on this, the isotropic covariance matrices commonly used in GMM are replaced with point-specific Laplacian matrices, providing a more geometrically meaningful representation. Furthermore, the cost function is extended from relying solely on coordinates to high-dimensional graph signals, thereby leveraging richer information during the registration process. The maximum likelihood estimation is optimized through the Expectation-Maximization (EM) algorithm [2]. Finally, we leverage the extracted high-frequency components to design prior probability, which reduces the corresponding search space and improves the efficiency of the registration process.

The contributions of this paper are three-fold. **(i)** We introduce a representation dimensionality expansion from points to graph signals for each point in the point cloud. High-frequency components are extracted using a high-pass graph filter, and their invariance under rigid transformations is established (Theorem 1). These components are then concatenated with the coordinates and normals to construct high-dimensional graph signals. Integrating these graph signals into the registration process introduces more geometric detail and enhances registration accuracy. **(ii)** We propose a novel graph-based component-wise density function. For each point, we construct a local graph Laplacian matrix based on its graph signals, whose pseudo-inverse replaces the traditional isotropic covariance matrix in the GMMs. This substitution not only preserves the classical objective of minimizing point-wise discrepancies but also requires their variation trends in the joint representation of multiple attributes within the local graph structure to remain consistent, thereby effectively incorporating graph-structural characteristics. **(iii)** An effective prior probability construction method is proposed. Unlike traditional probabilistic methods that ignore prior information, our approach leverages the extracted high-frequency components to construct prior probability. This allows each point to consider only a limited subset of candidate correspondences, which significantly accelerates the registration process, especially when dealing with large-scale point clouds. To the best of our knowledge, no existing probability-based method has yet considered the issue of prior probability.

2 Related Work

Since this work focuses on fine registration, this section mainly reviews probabilistic registration methods based on GMMs and registration methods using GSP tools. Probabilistic methods assume that the distribution of a point set can be modeled as a mixture of Gaussians, and estimate soft correspondences between two point sets in a probabilistic manner [7, 9, 11, 13, 19, 20, 23, 26, 29]. Among classical probabilistic registration frameworks, Coherent Point Drift (CPD) is one of the most representative approaches [26]. It treats the target point set as the centroids of a GMM and maximizes the likelihood of the source points through the EM algorithm. Although CPD is highly robust, its major drawback is the high computational cost, since it requires pairwise probability computation for all points in each iteration. To address this limitation, FilterReg employs

Gaussian filtering to approximate the soft correspondence computation in the E step [9]. This strategy substantially accelerates the registration process while preserving high accuracy and robustness. However, FilterReg relies on complex filter construction and approximation, which increases implementation and integration costs. In addition, the use of Gaussian filters for probability estimation may introduce numerical approximation errors. ECMPR extends probabilistic registration by allowing general anisotropic covariance and updating the full covariance matrix in the M step [13, 22]. Although it provides greater flexibility in covariance modeling and more straightforward handling of outliers, its increased algorithmic complexity and slower convergence limit its applicability to large-scale point clouds. LSG-CPD improves registration accuracy by adaptively imposing different levels of point-to-point penalties according to local surface flatness, and uses GPU computation to improve efficiency [18]. However, it still relies on exhaustive pairwise traversal, resulting in limited computational efficiency for large-scale point clouds. Recent learning-based methods have also been developed for point cloud registration, mainly through feature learning, correspondence learning, or matching generation [3, 14, 28, 35]. These methods are mostly discussed in the context of coarse or global registration, while learning-based methods remain relatively rare in fine registration.

With the continuous development of GSP theory, its advantages in handling non-Euclidean structured data have become clearer, making it naturally applicable to the analysis and processing of irregular data such as point clouds. [1] propose a novel technique based on Laplacian eigenmap coupling, which performs coarse registration by explicitly modeling local structures. GraphReg leverages GSP to extract local geometric features and incorporates them into a rigid-body dynamics framework for mass and force modeling, thereby enabling high-precision and computationally efficient registration [38]. However, when establishing correspondences based on the invariance of graph structures, GraphReg adopts a one-to-one hard matching strategy, which is susceptible to outliers or local perturbations, potentially affecting the stability and robustness of the registration process.

3 Method

3.1 Formulation of Probabilistic Model

We denote the target set by $\mathcal{Y} = \{\mathbf{y}_m\}_{1 \leq m \leq M}$, and the source set by $\mathcal{X} = \{\mathbf{x}_n\}_{1 \leq n \leq N}$. Each point in the source set undergoes a rigid transformation. The 3D coordinates of the transformed points are parameterised by $\Theta = \{\mathbf{R}, \mathbf{t}\}$, where $\mathbf{R}$ and $\mathbf{t}$ denote the rotation matrix and translation vector, respectively. To estimate the registration parameters, it is necessary to establish the correspondences between the target points and the source points. In classical GMM-based point cloud registration methods, each target data point $\mathbf{y}_m$ is assigned to a Gaussian component centered at the transformed source point. The covariance matrix Σ_m in GMM-based method is typically chosen to be isotropic.

Our method lifts each point in the point cloud from 3D spatial coordinates to a high-dimensional graph signal. This graph signal consists of spatial coordinates, surface normals and high-frequency components extracted via high-pass graph filtering (Fig. 1(a)). Then, we construct a local graph for each point, where the edge weights encode the similarity between the attributes of each dimension of the graph signal. These graph signals exhibit smoothness over their respective local graphs, as illustrated in Fig. 1(b). When the target point $\mathbf{y}_m$ is correctly aligned with the source point $\mathbf{x}_n$, the transformed source signal should preserve smoothness on the graph $\mathbf{G}_m$ constructed from $\mathbf{y}_m$. Motivated by this mutual smoothness property, we define the following component-wise density function:

$$p(\mathbf{s}_n | m) = c_m \exp \left(-\frac{1}{2\sigma^2} \|\mathbf{s}_m - \mathcal{T}(\mathbf{s}_n)\|_{\tilde{\mathbf{L}}_m}^2 \right). \quad (1)$$

Here, $\mathcal{T}(\mathbf{s}_n) = \text{diag}(\mathbf{R}, \mathbf{R}, \mathbf{I})\mathbf{s}_n + [\mathbf{t}; \mathbf{0}; \mathbf{0}]$ represents the transformed source signal. The $\mathbf{s}_n$ and $\mathbf{s}_m$ denote the graph signals corresponding to points $\mathbf{x}_n$ and $\mathbf{y}_m$, respectively. The regularized Laplacian matrix is defined as $\tilde{\mathbf{L}}_m = \varepsilon \mathbf{I} + \mathbf{L}_m$, where $\mathbf{L}_m$ denotes the graph Laplacian matrix of the local graph $\mathbf{G}_m$ corresponding to node m , and ε is a small positive constant for numerical stability. For simplicity, we use $\mathbf{L}_m$ to denote the regularized Laplacian $\tilde{\mathbf{L}}_m$ hereafter. The scaling parameter σ is automatically initialized following the strategy used in CPD [26]. Replacing the conventional isotropic covariance with the graph Laplacian matrix $\mathbf{L}_m$ not only retains the classical objective of minimizing the signal discrepancy between transformed source and target points, but also introduces a graph smoothness prior on the transformed signals and enforces consistency in their variation trends over the graph structure. This substitution carries stronger geometric significance, enhancing both the robustness and interpretability of the registration process. Detailed theoretical justification is provided in the supplementary material.

3.2 Graph Signal Representation of Point Clouds

We propose a high-dimensional graph signal representation for each point as illustrated in Fig. 1(a). For a given point $\mathbf{y}_m \in \mathcal{Y}$ as an example, the corresponding graph signal is defined as $\mathbf{s}_m \in \mathbb{R}^9$, which combines three types of information: spatial coordinates $(\mathbf{y}_m^x, \mathbf{y}_m^y, \mathbf{y}_m^z)$, scaled surface normals $(\bar{\mathbf{n}}_m^x, \bar{\mathbf{n}}_m^y, \bar{\mathbf{n}}_m^z)$ and high-frequency components $(\mathbf{h}_m^1, \mathbf{h}_m^2, \mathbf{h}_m^3)$. Formally, the graph signal $\mathbf{s}_m$ is defined as

$$\begin{aligned} \mathbf{s}_m &= (\mathbf{y}_m^x, \mathbf{y}_m^y, \mathbf{y}_m^z, \bar{\mathbf{n}}_m^x, \bar{\mathbf{n}}_m^y, \bar{\mathbf{n}}_m^z, \mathbf{h}_m^1, \mathbf{h}_m^2, \mathbf{h}_m^3)^T \\ &= (\mathbf{y}_m; \bar{\mathbf{n}}_m; \mathbf{h}_m), \end{aligned} \quad (2)$$

where $\bar{\mathbf{n}}_m \in \mathbb{R}^3$ is the scaled surface normal defined as $\bar{\mathbf{n}}_m = \frac{\max(\mathbf{y}) - \min(\mathbf{y})}{\max(\mathbf{n}) - \min(\mathbf{n})} \mathbf{n}_m$ ensuring scale consistency between coordinate and normal domains. Here, $\max()$ and $\min()$ compute the maximum and minimum absolute values across all matrix entries.

Next, we describe the construction of high-frequency components. As shown in Fig. 1(a), we first construct a weighted K nearest neighbor global graph

$\mathcal{G}_{\mathcal{Y}}$ over the point cloud $\mathcal{Y}$ for graph high-pass filtering. Specifically, Eq. (3) defines the edge weight $\mathcal{W}_{ij}$ between two neighboring points $\mathbf{y}_i$ and $\mathbf{y}_j$ by jointly considering their spatial distance and surface normal difference [6]:

$$\mathcal{W}_{ij} = \exp \left(-\frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{\sigma_p^2} - \frac{\|\mathbf{n}_i - \mathbf{n}_j\|^2}{\sigma_n^2} \right), \quad (3)$$

where the parameters σ_p and σ_n control the sensitivity to spatial distance and normal deviation, respectively.

In GSP, high-pass graph filters play a crucial role as they enhance the high-frequency components of signals defined on graph structures, thereby highlighting local variations and discontinuities such as edges, anomalies, and texture patterns [27, 31]. In our method, a high-pass graph filter is employed to amplify the differences between points while suppressing the common components shared among point cloud nodes. We adopt a linear shift-invariant FIR graph filter formulated as $\mathcal{H} = h_0\mathbf{I} + h_1\mathcal{S} + \dots + h_n\mathcal{S}^{n-1}$ [30], where $h_i > 0$ and $\mathcal{S}$ denotes the graph shift operator. In our implementation, the normalized Laplacian matrix $\mathcal{L}$ is employed as the graph shift operator to perform spectral filtering. In the supplementary material, we introduced the principle of amplifying high-frequency components using high-pass graph filters. To effectively capture both geometric and structural characteristics of the point cloud, we construct three types of high-frequency components based on spatial coordinates and surface normals as follows:

$$\begin{aligned} \mathbf{h}_m^1 &= \|[\mathcal{L}\mathbf{y}]_{m,:}\|, \\ \mathbf{h}_m^2 &= \|[\mathcal{L}\bar{\mathbf{n}}]_{m,:}\|, \\ \mathbf{h}_m^3 &= \|[(\mathcal{L} + \mathcal{L}^2)\mathbf{y}]_{m,:}\|, \end{aligned} \quad (4)$$

where $\mathbf{y}$ is the $M \times 3$ matrix composed of point coordinates $\mathbf{y}_m, \forall m = 1, \dots, M$, and $\bar{\mathbf{n}}$ is defined analogously for scaled normals. The notation $[\cdot]_{m,:}$ denotes the m -th row of the matrix. In Theorem 1, we provide the conclusion that the high-frequency components defined in (4) are invariant under the rigid transformation of the point cloud, and prove it in the supplementary material.

Theorem 1 (Invariance of high-frequency components under rigid transformation). *Let $\mathbf{x} \in \mathbb{R}^{M \times 3}$ be the coordinate matrix of a point cloud, and $\mathbf{y} = \mathbf{x}\mathbf{R}^T + \mathbf{T}$ be the result of applying a rigid transformation, where $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ is a rotation matrix, and $\mathbf{T} = \mathbf{1}_M \otimes \mathbf{t}^T$ applies the translation vector $\mathbf{t} \in \mathbb{R}^3$ to each point. Then, given the graph Laplacian $\mathcal{L}$, the high-frequency components for point $\mathbf{y}_m$ defined as (4) are invariant under rigid transformations.*

Due to the pronounced variation of high-frequency components between different points, these components serve as effective descriptors for achieving point cloud registration. To reduce computational complexity, rather than operating on the entire dense graph matrix, we use the values of K nearest neighbor indices of each node to perform weighted summation in practical calculations. To make Theorem 1 more clear and intuitive, we take the Stanford 3D Bunny as an example and visualize the energy distribution $\mathbf{h}_m^1 = \|[\mathcal{L}\mathbf{y}]_{m,:}\|$ of each point,

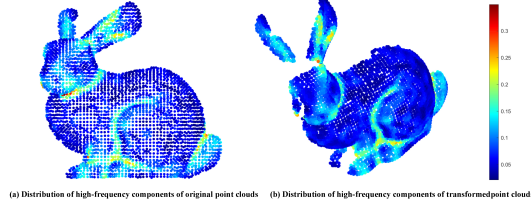


Fig. 2: Visualization of high-frequency components in point cloud registration.

as shown in Fig. 2, where the energy is normalized. As can be seen from Fig. 2, the high-frequency components of each point remain unchanged under the rigid transformation.

3.3 Local Graph Construction via Attribute Similarity

For each node $\mathbf{y}_m$, a local graph $\mathbf{G}_m$ is constructed using its graph signal $\mathbf{s}_m$ and those of its k nearest neighbors, as illustrated in Fig. 1(b). Since a single point is insufficient to characterize the underlying local geometric structure, we aggregate its k -nearest neighbors to jointly compute pairwise attribute similarities. Let $\mathcal{N}_k(m) = \{m_1, \dots, m_k\}$ denote the index set of the k -nearest neighbors of $\mathbf{y}_m$. Then, $\mathbf{Z}_m \in \mathbb{R}^{9 \times (k+1)}$ denotes the aggregated graph signals of $\mathbf{s}_m$ and its k nearest neighbors:

$$\mathbf{Z}_m = [\mathbf{s}_m, \mathbf{s}_{m_1}, \dots, \mathbf{s}_{m_k}], \quad (5)$$

where each column corresponds to a 9-dimensional graph signal, and each row represents a specific attribute, such as a coordinate axis, a normal component, or a high-frequency component.

Let $f_p \in \mathbb{R}^{k+1}$ denote the p -th row of $\mathbf{Z}_m$, corresponding to the p -th attribute across the local neighborhood, where $p = 1, \dots, 9$. We define the adjacency matrix $\mathbf{W}_m \in \mathbb{R}^{9 \times 9}$ of the local graph $\mathbf{G}_m$ using a Gaussian kernel that measures the similarity between attribute vectors:

$$\mathbf{W}_m^{ij} = \exp\left(-\frac{\|f_p - f_q\|^2}{2\sigma_f^2}\right), \forall p, q = 1, \dots, 9, \quad (6)$$

where σ_f is a scalar parameter controlling the sensitivity to attribute differences. With the adjacency matrix $\mathbf{W}_m$, we compute the normalized graph Laplacian:

$$\mathbf{L}_m = \mathbf{I} - \mathbf{D}_m^{-1/2} \mathbf{W}_m \mathbf{D}_m^{-1/2}, \quad (7)$$

where $\mathbf{D}_m$ is the diagonal degree matrix. It is important to distinguish between the global graph $\mathcal{G}_Y \in \mathbb{R}^{M \times M}$ and local graph $\mathbf{G}_m \in \mathbb{R}^{9 \times 9}$. The global graph $\mathcal{G}_Y$ is constructed over the entire point cloud $\mathcal{Y}$ and facilitates the design of a high-pass filter for extracting high-frequency components. In contrast, the local graph $\mathbf{G}_m$ is constructed individually for each point $\mathbf{y}_m$, capturing attribute-wise relationships within its 9-dimensional graph signal $\mathbf{s}_m$.

3.4 EM Optimization in Graph Signal Registration

Similar to probabilistic methods, we estimate the rigid transformation $\Theta = \{\mathbf{R}, \mathbf{t}\}$ and the registration probabilities by optimizing the following negative log-likelihood objective function:

$$L(\mathcal{T}, \sigma^2) = - \sum_{n=1}^N \log p(\mathcal{T}(\mathbf{s}_n)). \quad (8)$$

Solving this optimization problem directly is computationally infeasible. To address this, the EM algorithm is employed:

E step: Given the previous estimate $\mathcal{T}_{\text{old}}$, Bayesian inference yields the posterior correspondence probabilities between $\mathcal{X}$ and $\mathcal{Y}$. These probabilities are organized into a matrix $\mathbf{P} \in \mathbb{R}^{M \times N}$, where each element $\mathbf{P}_{mn}$ represents the posterior probability that point $\mathbf{x}_n$ corresponds to the m -th component. The $\mathbf{P}_{mn}$ can be calculated as

$$\mathbf{P}_{mn} = \frac{\pi_{mn} p(\mathcal{T}_{\text{old}}(\mathbf{s}_n)|m)}{\pi_o + \sum_{m=1}^M \pi_{mn} p(\mathcal{T}_{\text{old}}(\mathbf{s}_n)|m)}, \quad (9)$$

where π_{mn} denotes the prior probability that point $\mathbf{x}_n$ belongs to the m -th component. The π_o is the probability density associated with outliers, whose setting follows the approach in [13]. We accelerate the computation of matrix $\mathbf{P}$ through GPU parallelization and simplify the calculation of $\mathbf{P}_{mn}$ by leveraging prior probabilities to focus only on relevant indices. Furthermore, we accelerate index-based matrix rearrangement by implementing custom CUDA kernels, overcoming the inherent inefficiency of general-purpose GPU operations for such tasks.

M step: In the presence of outliers, the neighbor set of aligned points between the original point cloud and the target point cloud may become inconsistent, resulting in deviations in normal vector estimation and high-frequency components. Directly incorporating this information into transformation updates can introduce errors and reduce registration accuracy. To address this, in the M-step, we estimate the rigid transformation using only point coordinates, thereby mitigating the impact of noise in surface normals and high-frequency components caused by local neighborhood variations. The transformation $(\mathbf{R}, \mathbf{t})$ is computed by minimizing the following objective function:

$$\begin{aligned} (\mathbf{R}, \mathbf{t}) &= \arg \min_{\mathbf{R}, \mathbf{t}} Q(\mathbf{R}, \mathbf{t}, \sigma), \\ \text{where } Q(\mathbf{R}, \mathbf{t}, \sigma) &= - \sum_{n=1}^N \sum_{m=1}^M \mathbf{P}_{mn} \log(\pi_{mn} p(\mathbf{x}_n|m)), \end{aligned} \quad (10)$$

with $p(\mathbf{x}_n|m) = c_m \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y}_m - \mathbf{R}\mathbf{x}_n - \mathbf{t}\|_{\mathbf{L}_m^{\text{sub}}}^2\right)$. Here, $\mathbf{L}_m^{\text{sub}}$ denotes the upper-left 3×3 submatrix of $\mathbf{L}_m$, corresponding to the spatial coordinate block. As described in [18], Newton's method is employed to iteratively update the rigid transformation on the Lie group. Let $\mathbf{E}_1, \mathbf{E}_2, \dots, \mathbf{E}_6$ represent the basis matrices of the Lie algebra. The gradient and Hessian of the objective function

Q are given by

$$\begin{aligned} [\nabla Q]_i &= 2 \sum_{m=1}^M \sum_{n=1}^N \mathbf{P}_{mn} (\tilde{g} \tilde{\mathbf{x}}_n - \tilde{\mathbf{y}}_m)^T \tilde{\mathbf{L}}_m^{\text{sub}} \tilde{g} \mathbf{E}_i \tilde{\mathbf{x}}_n, \\ [\mathbf{H}]_{ij} &= 2 \sum_{m=1}^M \sum_{n=1}^N \mathbf{P}_{mn} (\tilde{\mathbf{x}}_n^T \mathbf{E}_j^T \tilde{g}^T \tilde{\mathbf{L}}_m^{\text{sub}} \tilde{g} \mathbf{E}_i \tilde{\mathbf{x}}_n + (\tilde{g} \tilde{\mathbf{x}}_n - \tilde{\mathbf{y}}_m)^T \tilde{\mathbf{L}}_m^{\text{sub}} \tilde{g} \mathbf{E}_j \mathbf{E}_i \tilde{\mathbf{x}}_n). \end{aligned} \quad (11)$$

In homogeneous coordinates, we denote $\tilde{g}$, $\tilde{\mathbf{x}}_n$ and $\tilde{\mathbf{y}}_m$ as the augmented representations of rigid transformation g and points $\mathbf{x}_n$, $\mathbf{y}_m$, respectively. The augmented inverse covariance matrix is constructed through direct sum operation, i.e., $\tilde{\mathbf{L}}_m^{\text{sub}} = \mathbf{L}_m^{\text{sub}} \oplus \mathbf{0}$. Use Newton's method to update the transformation until it converges:

$$\tilde{g}_{i+1} = \tilde{g}_i \circ \exp\left(\frac{1}{2}(\mathbf{H} + \mathbf{H}^T)^{-1} \nabla Q\right)^\wedge, \quad (12)$$

where $\wedge$ denotes the mapping from a vector to its associated matrix Lie algebra element [25]. After the transformation g is updated, the Laplacian scaling operator σ^2 is also updated accordingly. Due to space constraints, we present only the key computational formulas here. Please refer to the supplementary material for detailed derivations.

3.5 Efficient Prior Construction from High-frequency Component

In conventional GMM-based point cloud registration methods, the prior correspondence probability π_{mn} between points $\mathbf{x}_n \in \mathcal{X}$ and $\mathbf{y}_m \in \mathcal{Y}$ is typically unknown. In the absence of prior knowledge, each point in $\mathcal{X}$ must be exhaustively compared with all points in $\mathcal{Y}$, resulting in high computational cost. Thanks to the invariance of high-frequency components under rigid transformation proved in Theorem 1, we propose a strategy to construct prior correspondence probabilities. We build a KD-tree using high-frequency components and retrieve the k_{match} nearest candidate points in $\mathcal{Y}$ for each point in $\mathcal{X}$. The prior correspondence probability is then defined as $\pi_{mn} = 1/k_{\text{match}}$, if $\mathbf{y}_m \in \mathcal{C}_n$ and 0 otherwise, where $\mathcal{C}_n$ denotes the set of k_{match} nearest candidates of $\mathbf{x}_n$ in $\mathcal{Y}$. By incorporating prior information, each point only needs to be matched with a small subset of potential candidates, thereby reducing the search space of the point cloud and greatly improving the registration efficiency.

4 Result

In this section, a series of experiments are conducted on benchmark 3D point cloud datasets to evaluate the robustness, accuracy and computational efficiency of the proposed method. All evaluations are performed in MATLAB on a workstation equipped with an NVIDIA Titan Xp GPU. The baseline methods are implemented either using code provided by the original authors or obtained from widely adopted, performance-optimized open-source libraries. The parameter settings for the baseline methods follow the authors' recommendations or software defaults when available, and are otherwise carefully tuned to ensure

fairness and optimal performance. For runtime fairness, we use the same input resolution for all methods whenever feasible. Lower resolutions are adopted only when some baselines, such as ECMPR and LSG-CPD, cannot process large point clouds due to memory overflow or prohibitive runtime.

The parameter settings for our method are as follows. Similar to CPD, the initial value of σ is automatically estimated from the data used in each experiment. In our method, the parameter was set as $\varepsilon = 0.01$. The global graphs $(\mathcal{G}_X, \mathcal{G}_Y)$ are constructed using $K = 10$ nearest neighbors, while the local graphs $(\mathbf{G}_m, \forall m = 1, \dots, M)$ are constructed using $k = 10$ nearest neighbors. The parameters σ_p, σ_n and σ_f were each set to the median of their respective pairwise distances. Unless k_{match} is explicitly varied for sensitivity analysis, it is set to 100, 50, 50, 200 and 200 for the accuracy test, robustness test, invariance property assessment, multi-view point set registration and LiDAR dataset, respectively. Following [12], we evaluate the registration accuracy by computing the angular error (in degrees) between the estimated rotation matrix $\mathbf{R}$ and the ground-truth rotation matrix $\hat{\mathbf{R}}$, i.e., $\text{AngErr}(\hat{\mathbf{R}}, \mathbf{R}) = \frac{180^\circ}{\pi} \arccos\left(\frac{1}{2}(\text{Tr}(\hat{\mathbf{R}}\mathbf{R}^T) - 1)\right)$.

4.1 Registration on Range Datasets

1) Accuracy test: To evaluate the registration accuracy of the proposed method, two representative datasets were selected from the Stanford 3D Scanning Repository: Dragon and Armadillo [5]. To demonstrate the effectiveness of our method on large-scale point clouds, we downsample each original point cloud to approximately 20,000 points. Both datasets are captured using the Cyberware 3030MS laser scanner from two different viewpoints, with angular deviations of approximately 24° and 30° , respectively. Due to the large difference in viewing angles, the two scans only showed partial overlap, as illustrated in Fig. 3.

We compared our method with six baseline methods: GraphReg [38], CPD [26], FilterReg (point-to-point version) [9], TrICP [4], LSG-CPD [18] and ECMPR [13]. CPD and FilterReg represent probabilistic method with isotropic covariance, while LSG-CPD and ECMPR represent probabilistic methods with anisotropic covariance. The TrICP serves as a robust extension of classical ICP methods, while GraphReg uses GSP to accelerate and improve the performance of physics-based point cloud registration. Due to memory limitations based on probabilistic methods, we downsampled the point clouds of ECMPR and LSG-CPD methods to approximately 8,000 points.

Tab. 1 summarizes the registration accuracy and runtime. The results show that GraphCPD achieves accurate registration while maintaining efficient computation. Although probabilistic methods such as FilterReg and ECMPR obtain relatively accurate results, their computational costs become high on large-scale point clouds. In contrast, TrICP and GraphReg are faster but less accurate. Overall, GraphCPD provides a favorable trade-off between accuracy and efficiency.

We further analyze the influence of two neighborhood-related parameters. The parameter k_{match} controls the search range for constructing prior correspon-

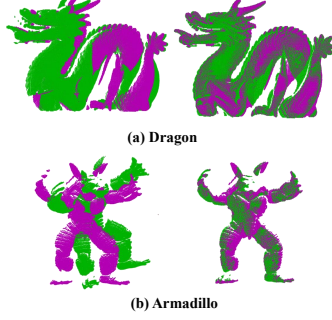


Fig. 3: Registration results of the proposed method on Dragon and Armadillo datasets.

Table 1: Registration accuracy and runtime for Dragon and Armadillo datasets.

Method	Dragon		Armadillo	
	AngErr ($^{\circ}$)	Time (s)	AngErr ($^{\circ}$)	Time (s)
GraphReg	0.403	4.30	0.284	5.63
CPD	0.624	96.57	0.597	97.20
LSG-CPD	0.155	0.862	0.138	0.758
FilterReg	0.297	20.05	0.171	13.51
TrICP	0.800	0.50	0.119	0.55
ECMPR	0.203	133.40	0.173	137.86
GraphCPD	0.091	2.950	0.095	2.846

Table 2: Registration accuracy and runtime with different k_{match} .

Dataset	Metric	$k_{\text{match}} = 50$	$k_{\text{match}} = 100$	$k_{\text{match}} = 150$	$k_{\text{match}} = 200$	$k_{\text{match}} = 250$
Dragon	AngErr ($^{\circ}$)	0.242	0.091	0.089	0.074	0.073
	Time (s)	2.559	2.950	5.951	7.045	10.045
Armadillo	AngErr ($^{\circ}$)	0.215	0.095	0.084	0.078	0.071
	Time (s)	2.354	2.846	5.034	7.587	11.592

Table 3: Sensitivity to local graph size k .

Dataset	Metric	$k = 5$	$k = 10$	$k = 15$	$k = 20$
Dragon	AngErr ($^{\circ}$)	0.103	0.091	0.156	0.362
	Time (s)	3.212	2.950	2.732	2.683
Armadillo	AngErr ($^{\circ}$)	0.148	0.095	0.122	0.244
	Time (s)	3.013	2.846	2.014	1.934

Table 4: Registration accuracy comparison of different configurations (AngErr, $^{\circ}$).

Configuration	Dragon	Armadillo
Local Laplacian density model, Weighted global graph (Ours)	0.091	0.095
Isotropic covariance, Weighted global graph	0.1542	0.3836
Local Laplacian density model, Unweighted global graph	0.232	0.308

dence probabilities. As shown in Tab. 2, increasing k_{match} improves accuracy by considering more candidate correspondences, but also increases the runtime. In addition, we evaluate the local graph neighborhood size k used for Laplacian construction. As shown in Tab. 3, $k = 10$ gives the best accuracy. A smaller k provides insufficient local geometry, whereas a larger k may over-smooth local variations. Therefore, we set $k = 10$ in all experiments.

We also conduct an ablation study to quantify the contributions of the local Laplacian density model and the weighted global graph for high-pass filtering. Specifically, we replace the local Laplacian density model with isotropic Gaussian covariance and replace the weighted global graph with a unweighted global graph for high-pass filtering. As summarized in Tab. 4, the full model achieves the best performance on both datasets. The comparison shows that the local Laplacian

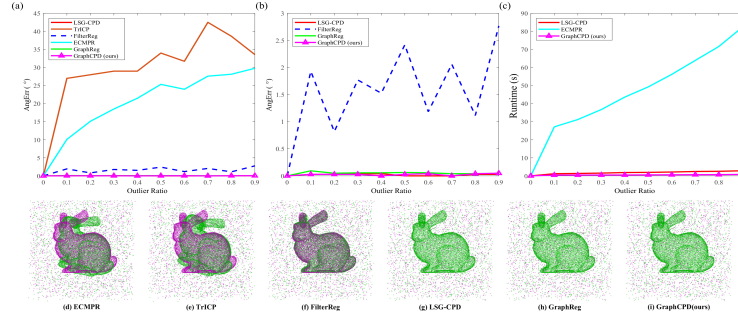


Fig. 4: Outlier results. (a) Angular errors of various methods across increasing outlier ratios. (b) Zoomed-in view comparing LSG-CPD, FilterReg, GraphReg and GraphCPD for better clarity. (c) Average runtime comparison of LSG-CPD, ECMPR and GraphCPD under varying outlier ratios. (d)–(i) Visualization of registration results on data with 0.5 outlier ratio.



Fig. 5: Verification of the graph invariance property on the Utah Teapot dataset.

density model is more effective than isotropic covariance, and the weighted global graph is more effective than the binary-edge global graph.

2) *Robustness test:* To evaluate the robustness of the proposed method, experiments were conducted on the Bunny dataset from the Stanford 3D Scanning Repository. The original point cloud is uniformly downsampled to approximately 25,000 points, and then subjected to random rotations of around 40° along the x , y and z axes. To simulate varying levels of disturbance, we introduce Gaussian outliers at different ratios, defined as the number of outliers relative to the number of original points. The result in Fig. 4 shows that GraphCPD, LSG-CPD, FilterReg and GraphReg consistently achieve low angular errors across various outlier ratios, demonstrating strong robustness against outlier contamination. The performance of TrICP and ECMPR methods deteriorates rapidly as the outlier ratio increases, which are sensitive to outliers. To further evaluate the computational efficiency, a comparison is made of the runtime of several probabilistic methods in Fig. 4(c). The results indicate that GraphCPD maintains excellent robustness while ensuring low computation time.

4.2 Assessment of the Invariance Property

To assess the effectiveness of incorporating high-frequency components into point cloud registration, we conducted experiments on the Utah Teapot dataset [21] and compared our method against several baseline approaches. In the experiment, the target point cloud represents the teapot, while the source point cloud corresponds to its pot cover. The objective is to accurately align the pot cover

Table 5: Registration accuracy and runtime comparison on the Utah Teapot dataset.

Method	AngErr ($^{\circ}$)	Time (s)
TrICP	16.739	0.353
CPD	0.657	56.84
ECMPR	39.038	23.678
GraphReg	0.067	2.344
GraphCPD	0.018	0.370

Table 6: Registration accuracy on multi-view point set.

Method	Dragon	Happy	Armadillo
MATrICP	2.055	1.903	2.195
FilterReg	0.405	0.230	0.177
EMPMR	0.533	0.235	0.114
GraphCPD	0.236	0.264	0.095

to the teapot, with the pot body treated as outliers. As illustrated in Fig. 5 and Tab. 5, our method achieves superior registration accuracy while reducing computational time. This performance improvement is mainly attributed to the richer prior information extracted from high-frequency components. In addition, high-frequency components are incorporated into the graph signal and participate in registration, effectively improving accuracy. The GraphReg method also relies on the transformation invariance of graph structure representation, thus demonstrating good estimation accuracy in this experiment. However, GraphReg adopts a one-to-one matching strategy, which makes it more susceptible to interference when facing outliers, thereby limiting its stability in high outlier scenarios.

4.3 Evaluation on Multi-view Point Set Registration

In this subsection, we conduct experiments on three Stanford 3D Scanning Repository benchmarks: the Dragon Stand, Happy Stand, and Armadillo datasets. Each dataset comprises multiple views of a single object captured from different perspectives, along with ground-truth rigid transformations for alignment. We compare our approach against several representative baselines, including MA-TrICP [17], EMPMR [39] and FilterReg. Following [9], random perturbations are applied to the ground-truth poses, with rotation errors limited to ± 0.05 radians and translation errors within ± 0.02 meters. All point clouds are voxelized and downsampled to approximately 15,000 points per view. The average rotational error is adopted as the evaluation metric to quantify the registration accuracy.

Tab. 6 shows the average rotation error of multiple perspectives using different methods on three datasets. To more intuitively illustrate the registration performance, Fig. 6 visualizes the aligned point clouds using cross-sectional views. Experimental results show that GraphCPD achieves superior registration accuracy on two of the datasets. It is worth noting that, since our method adopts a pairwise registration strategy, error accumulation may occur along the chain of sequential transformations. Nevertheless, in comparison with existing joint multi-view registration approaches, our method still demonstrates competitive overall performance, indicating its robustness in terms of registration accuracy.

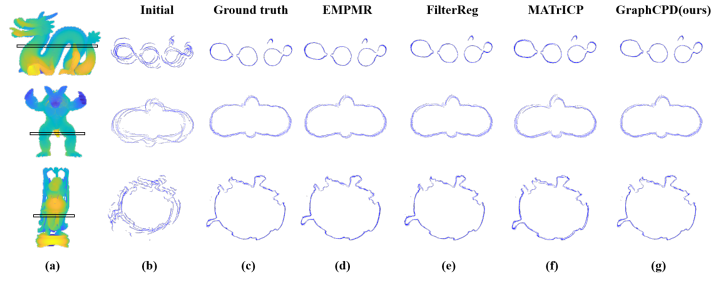


Fig. 6: Multi-view registration results. The complete target model reconstructed by our method is shown, with annotated multi-view cross-sections. (b) and (c) are the initial states and corresponding ground truth. The remaining figures show the registration performance of each method in the cross-section.

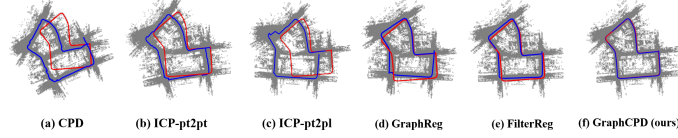


Fig. 7: Qualitative evaluation of registration performance on the KITTI dataset, where red lines represent the ground truth and blue lines indicate the aligned trajectories.

As future work, we aim to explore globally consistent multi-view point cloud registration within the framework of GSP, enabling simultaneous optimization across all views and further improving overall alignment consistency and accuracy.

4.4 Low-overlap Indoor Registration

To further evaluate the robustness of GraphCPD under low-overlap indoor scenes, we conduct an additional experiment on the first scan pair from the 3DMatch benchmark [37]. 3DMatch contains real indoor RGB-D fragments with partial scene observations, so different fragments often share only limited overlap, making it suitable for evaluating low-overlap registration. As shown in Tab. 7, GraphCPD achieves the lowest angular error among all compared baselines, demonstrating the effectiveness of the proposed graph-prior-guided probabilistic registration in low-overlap scenarios. We also vary k_{match} to examine the effect of the correspondence search space. Increasing k_{match} from 100 to 500 reduces the angular error from 1.350° to 0.947° , with a moderate runtime increase from 1.543 s to 2.562 s, indicating that a larger candidate range can alleviate over-restriction while preserving efficiency.

4.5 LiDAR Point Cloud Registration Evaluation

In this experiment, we evaluate our algorithm using the KITTI dataset [10], focusing on sequence 07, which consists of a driving trajectory of nearly 700

Table 7: Registration results on a low-overlap 3DMatch scan pair.

Metric	GraphReg	CPD	LSG	FReg	TrICP	ECMPR	GraphCPD		
							$k_{\text{match}}=100$	$k_{\text{match}}=200$	$k_{\text{match}}=500$
AngErr ($^{\circ}$)	6.678	11.741	10.985	26.899	20.021	1.666	1.350	1.340	0.947
Time (s)	5.154	103.240	3.150	60.440	0.690	164.510	1.543	1.854	2.562

Table 8: Registration performance comparison on the KITTI dataset.

Method	Rel. Rot. ($^{\circ}$)	Rel. Tran. (m)	Abs. Rot. ($^{\circ}$)	Abs. Tran. (m)	Last Rot. ($^{\circ}$)	Last Tran. (m)
CPD	0.388 (25.741)	0.345 (6.055)	19.296 (28.999)	46.089 (77.362)	21.942	32.901
ICP-pt2pt	0.218 (5.958)	0.289 (3.710)	8.739 (17.833)	16.848 (35.418)	6.117	18.279
ICP-pt2pl	0.450 (17.639)	0.178 (17.050)	14.760 (23.590)	44.978 (78.234)	9.308	52.650
GraphReg	0.171 (11.727)	0.183 (0.665)	14.293 (10.838)	6.778 (28.488)	7.868	14.513
FilterReg	0.146 (3.250)	0.074 (0.291)	3.548 (5.178)	9.470 (14.187)	3.250	11.999
GraphCPD	0.0292 (0.085)	0.108 (0.883)	1.831 (3.581)	3.744 (8.738)	2.094	6.850

meters. Registration pairs are constructed by sampling every two frames, yielding a total of 550 pairs for testing. To ensure computational efficiency, each point cloud is downsampled to approximately 5,000 points using a voxel grid filter. All frames are registered to the first frame to reconstruct the complete trajectory. Following the evaluation protocol in [18], we assess both relative and absolute rotation and translation errors, as well as the final deviation between the last and first frames. Translation error is defined as the Euclidean distance between the estimated and ground-truth translations, while rotation error is computed as the angular difference in the axis-angle representation. As demonstrated in Fig. 7 and Tab. 8, our proposed method exhibits superior performance in terms of rotation accuracy when compared to existing baselines.

5 Conclusion

We propose a new probabilistic registration GMM-based method, which demonstrates strong robustness, accuracy and computational efficiency. Our method utilizes a high-pass graph filter to extract high-frequency components for constructing prior probabilities, thereby improving the speed of registration. By combining high-frequency components with coordinates and normals, high dimensional graph signals are obtained, and graph Laplacian matrices are constructed for each node to replace the traditional isotropic matrix in the probabilistic method, providing a more discriminative geometric description. In the future, we will explore globally consistent multi-view point cloud registration within the GSP framework to achieve simultaneous optimization of all views.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62550085 and in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LZ24F030006.

References

1. Bastico, M., Decencière, E., Corté, L., Tillier, Y., Ryckelynck, D.: Coupled Laplacian eigenmaps for locally-aware 3D rigid point cloud matching. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 3447–3458 (2024)
2. Bishop, C.M., Nasrabadi, N.M.: Pattern Recognition and Machine Learning, vol. 4 (2006)
3. Chen, H., Yan, P., Xiang, S., Tan, Y.: Dynamic cues-assisted transformer for robust point cloud registration. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 21698–21707 (2024)
4. Chetverikov, D., Svirko, D., Stepanov, D., Krsek, P.: The trimmed iterative closest point algorithm. In: Proceedings of the International Conf. on Pattern Recognition (ICPR). pp. 545–548 (2002)
5. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: Proceedings of the Annual Conf. on Computer Graphics and Interactive Techniques. pp. 303–312 (1996)
6. Dinesh, C., Cheung, G., Bajić, I.V.: Point cloud sampling via graph balancing and gershgorin disc alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(1), 868–886 (2022)
7. Eckart, B., Kim, K., Kautz, J.: HGMR: Hierarchical gaussian mixtures for adaptive 3D registration. In: Proceedings of the European Conf. on Computer Vision (ECCV). pp. 705–721 (2018)
8. Evangelidis, G.D., Kounades-Bastian, D., Horaud, R., Psarakis, E.Z.: A generative model for the joint registration of multiple point sets. In: Proceedings of the European Conf. on Computer Vision (ECCV). pp. 109–122 (2014)
9. Gao, W., Tedrake, R.: FilterReg: Robust and efficient probabilistic point-set registration using gaussian filter and twist parameterization. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 11095–11104 (2019)
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the KITTI vision benchmark suite. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 3354–3361 (2012)
11. Granger, S., Pennec, X.: Multi-Scale EM-ICP: A fast and robust approach for surface registration. In: Proceedings of the European Conf. on Computer Vision (ECCV). pp. 418–432 (2002)
12. Hirose, O.: A Bayesian formulation of coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(7), 2269–2286 (2020)
13. Horaud, R., Forbes, F., Yguel, M., Dewaele, G., Zhang, J.: Rigid and articulated point registration with expectation conditional maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(3), 587–602 (2010)
14. Huang, X., Mei, G., Zhang, J.: Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 11366–11374 (2020)
15. Jian, B., Vemuri, B.C.: Robust point set registration using Gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8), 1633–1645 (2010)
16. Jiang, H., Dang, Z., Gu, S., Xie, J., Salzmann, M., Yang, J.: Center-based decoupled point-cloud registration for 6D object pose estimation. In: Proceedings of the Intl. Conf. on Computer Vision (ICCV). pp. 3427–3437 (2023)

17. Li, Z., Zhu, J., Lan, K., Li, C., Fang, C.: Improved techniques for multi-view registration with motion averaging. In: *Proceedings of the Intl. Conf. on 3D Vision (3DV)*. pp. 713–719 (2014)
18. Liu, W., Wu, H., Chirikjian, G.S.: LSG-CPD: Coherent point drift with local surface geometry for point cloud registration. In: *Proceedings of the Intl. Conf. on Computer Vision (ICCV)*. pp. 15293–15302 (2021)
19. Luo, B., Hancock, E.R.: A unified framework for alignment and correspondence. *Computer Vision and Image Understanding* **92**(1), 26–55 (2003)
20. McNeill, G., Vijayakumar, S.: A probabilistic approach to robust shape matching. In: *Proceedings of the Intl. Conf. on Image Processing (ICIP)*. pp. 937–940 (2006)
21. Mellado, N., et al.: OpenGR: A C++ library for 3D global registration (2017), accessed: 2025-05-01
22. Min, Z., Pan, J., Zhang, A., Meng, M.Q.H.: Robust non-rigid point set registration algorithm considering anisotropic uncertainties based on coherent point drift. In: *Proceedings of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. pp. 7903–7910 (2019)
23. Min, Z., Wang, J., Meng, M.Q.H.: Robust generalized point cloud registration with orientational data based on expectation maximization. *IEEE Transactions on Automation Science and Engineering* **17**(1), 207–221 (2019)
24. Mundermann, L., Corazza, S., Andriacchi, T.P.: Accurately measuring human movement using articulated ICP with soft-joint constraints and a repository of articulated models. In: *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. pp. 1–6 (2007)
25. Murray, R.M., Li, Z., Sastry, S.S.: *A Mathematical Introduction to Robotic Manipulation*. CRC Press (2017)
26. Myronenko, A., Song, X.: Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(12), 2262–2275 (2010)
27. Ortega, A., Frossard, P., Kovačević, J., Moura, J.M., Vandergheynst, P.: Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE* **106**(5), 808–828 (2018)
28. Poiesi, F., Boscaini, D.: Learning general and distinctive 3D local deep descriptors for point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(3), 3979–3985 (2022)
29. Ravikumar, N., Gooya, A., Frangi, A.F., Taylor, Z.A.: Generalised coherent point drift for group-wise registration of multi-dimensional point sets. In: *Proceedings of the Medical Image Computing and Computer Assisted Intervention*. pp. 309–316 (2017)
30. Sandryhaila, A., Moura, J.M.: Discrete signal processing on graphs. *IEEE Transactions on Signal Processing* **61**(7), 1644–1656 (2013)
31. Sandryhaila, A., Moura, J.M.: Discrete signal processing on graphs: Graph filters. In: *Proceedings of the Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*. pp. 6163–6166 (2013)
32. Serafin, J., Grisetti, G.: NlCP: Dense normal based point cloud registration. In: *Proceedings of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. pp. 742–749 (2015)
33. Stoyanov, T., Magnusson, M., Andreasson, H., Lilienthal, A.J.: 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)

- 34. Whelan, T., Salas-Moreno, R.F., Glocker, B., Davison, A.J., Leutenegger, S.: ElasticFusion: Real-time dense SLAM and light source estimation. *The International Journal of Robotics Research* **35**(14), 1697–1716 (2016)
- 35. Yan, S., Shi, P., Li, J.: Ml-semreg: Boosting point cloud registration with multi-level semantic consistency. In: *Proceedings of the European Conf. on Computer Vision (ECCV)*. pp. 19–37 (2024)
- 36. Yang, H., Shi, J., Carlone, L.: TEASER: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics* **37**(2), 314–333 (2020)
- 37. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3DMatch: Learning local geometric descriptors from RGB-D reconstructions. In: *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. pp. 1802–1811 (2017)
- 38. Zhao, M., Ma, L., Jia, X., Yan, D.M., Huang, T.: GraphReg: Dynamical point cloud registration with geometry-aware graph signal processing. *IEEE Transactions on Image Processing* **31**, 7449–7464 (2022)
- 39. Zhu, J., Guo, R., Li, Z., Zhang, J., Pang, S.: Registration of multi-view point sets under the perspective of expectation-maximization. *IEEE Transactions on Image Processing* **29**, 9176–9189 (2020)