HDMNet: A Hierarchical Matching Network with Double Attention for Large-scale Outdoor LiDAR Point Cloud Registration

Weiyi Xue, Fan Lu, Guang Chen*
Tongji University
{xwy,lufan,guangchen}@tongji.edu.cn

Abstract

Outdoor LiDAR point clouds are typically large-scale and complexly distributed. To achieve efficient and accurate registration, emphasizing the similarity among local regions and prioritizing global local-to-local matching is of utmost importance, subsequent to which accuracy can be enhanced through cost-effective fine registration. In this paper, a novel hierarchical neural network with double attention named HDMNet is proposed for large-scale outdoor LiDAR point cloud registration. Specifically, A novel feature consistency enhanced double-soft matching network is introduced to achieve two-stage matching with high flexibility while enlarging the receptive field with high efficiency in a patch-to-patch manner, which significantly improves the registration performance. Moreover, in order to further utilize the sparse matching information from deeper layer, we develop a novel trainable embedding mask to incorporate the confidence scores of correspondences obtained from pose estimation of deeper layer, eliminating additional computations. The high-confidence keypoints in the sparser point cloud of the deeper layer correspond to a high-confidence spatial neighborhood region in shallower layer, which will receive more attention, while the features of non-key regions will be masked. Extensive experiments are conducted on two large-scale outdoor LiDAR point cloud datasets to demonstrate the high accuracy and efficiency of the proposed HDMNet.

1. Introduction

Point cloud registration is a fundamental task in 3D computer vision, aiming to estimate the optimal rigid transformation for aligning two point clouds. This task shares similarities with tasks such as LIDAR odometry [10], and has been utilized in diverse practical applications such as intelligent robotics [31], autonomous driving [28].
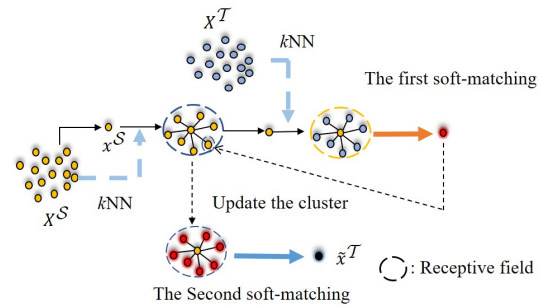
In certain registration algorithms, the performance is

---

*Corresponding author: guangchen@tongji.edu.cn

Figure 1. Illustration of double-soft matching strategy. Given the source point cloud X^S and the target point cloud X^T, we perform two rounds of soft-matching. The first round of soft-matching is conducted between the source and target point clouds to update the points in the source point cloud. Subsequently, the second round of soft-matching is performed within the updated source point cloud.

hindered due to the characteristics of outdoor Lidar point clouds, including their heightened sparsity, expanded spatial extent, and intricate distribution. The Iterative Closest Point(ICP) [6] and its variants [36, 38, 49] are widely recognized as the most prominent approaches. However, ICP are heavily relies on the initial alignment and easily converge to a local minimum. Most of learning-based methods primarily concentrate on object-level [2, 22, 45, 46] or indoor point clouds [8, 14, 17] and rely on assumptions regarding point cloud distributions. Recently, HRegNet [25] have demonstrated remarkable efficacy in addressing the challenges posed by sparse features. However, considering the presence of errors in keypoint descriptors, the utilization of point-to-point [22] and point-to-patch [21, 24, 25] matching strategies can introduce erroneous correspondences. As a result, existing approaches either lack reliability or are time-consuming when applied to outdoor LiDAR point cloud registration.

To address the aforementioned challenges, we propose HDMNet for large-scale outdoor point cloud registration. HDMNet performs registration hierarchically, leveraging the generation of virtual keypoints. In global local-to-local
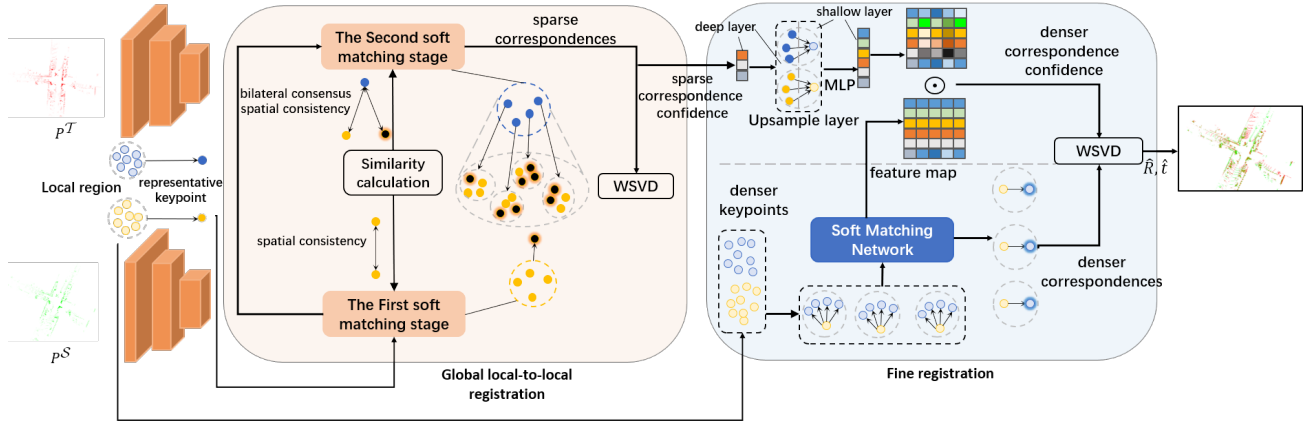
Figure 2. Network architecture of the proposed HDMNet. HDMNet consists of two main components: global local-to-local and fine registration. We use a feature extractor to hierarchically abstract the local regions into keypoints and descriptors, leveraging the reliable features at the deeper layer along with precise position information at the shallower layer. We employ the feature consistency enhanced double-soft matching module for global registration while simultaneously outputting the confidence scores of the correspondences. After that, we utilize the confidence scores to generate an initial mask for local regions, enhancing the focus on key regions and establish a denser set of correspondences to perform fine registration.

registration process of the deepest layer, considering the potential errors in descriptors that may result in a number of mismatches, we employ a double-soft matching strategy, as depicted in Fig. 1. This strategy involves two consecutive matching stages, wherein the points from the target point cloud participate in both matching steps. To incorporate them into the registration pipeline, we introduce a learning-based correspondence network with a feature consistency enhanced double-soft matching module, achieving two-stage matching with high flexibility while enlarging the receptive field. Furthermore, in the global registration, the sparsity of keypoints enables us to employ more robust strategies while maintaining high efficiency, thus we incorporate consistency features and sparse-to-denser matching strategy into the double-soft matching network, where sparse-to-denser matching refers to matching the sparse source point clouds with our updated, denser target point clouds. The entire double-soft matching network significantly improves the registration accuracy due to the patch-to-patch approach to search for correspondences among super points.

The overall network architecture is illustrated in Fig. 2. During the estimation of pose transformation in local-to-local registration, we simultaneously learn a set of confidence scores for correspondences, we can infer that each keypoint in sparser point clouds of the deeper layer correspond to a spatial neighborhood region in denser point cloud of shallower layer. Therefore, these scores, which represent the confidence of the local region of shallower layer, are reused in fine registration, thereby avoiding repetitive calculations and enhancing overall efficiency. Extensive experiments are conducted on two large-scale

outdoor LiDAR point cloud datasets, namely the KITTI odometry dataset [13] and the NuScenes dataset [7]. The result demonstrate that HDMNet surpasses existing approaches in accuracy while maintaining high efficiency, presenting remarkable improvements.

Overall, our contributions are as follows:

- Our HDMNet for large-scale outdoor point cloud registration achieves state-of-the-art performance with high computational efficiency.

- We design feature consistency enhanced double-soft matching network, achieving two-stage matching with high flexibility while enlarging the receptive field in a patch-to-patch manner, significantly improving the performance.

- We design a mask prediction module to prioritize crucial regions with higher correspondence confidence, effectively leveraging the confidence from upper-layer to enhance accuracy while avoiding duplicated computations.

2. Related Works

2.1. Conventional point cloud registration

The Iterative Closest Point (ICP) [6] is widely recognized as the most prominent method for point cloud registration. However, its heavy reliance on the initial estimate and susceptibility to local minima have motivated the development of several variants [36, 38, 49]. FGR and TEASER [48, 51] are tolerant to outliers from robust cost functions.

Some approaches focus on feature extraction from point clouds [12, 18, 37, 39, 40]. For instance, Fast Point Feature Histogram (FPFH) [37] constructs an oriented histogram based on pairwise geometric properties. [15] presents a review of handcrafted features in 3D point clouds. Subsequently, Random sample Consensus (RANSAC) [11] and its variants [4,5,41] are commonly employed to remove outliers from the initial correspondences.

2.2. Learning-based point cloud registration

End-to-end registration. PointNetLK [2] integrates the Lucas & Kanade algorithm [27] with PointNet [32] to perform registration. In Feature-metric registration (FMR) [17], the alignment of two point clouds is achieved by minimizing the error of global feature projection. A transformer network was utilized in Deep Closest Point [45] to estimate soft correspondences while introduces significant computational overhead, leading to low efficiency. IDAM [22] introduces an iterative distance-aware similarity matrix convolution module and proposes a learnable point cloud downsampling method by utilizing the hard and hybrid point elimination, which evaluate and filter each point based on the scores.

Feature matching-based registration and deep point correlation. Feature matching methods commonly use precomputed features to establish point correspondences. Learning-based outlier rejection modules are then employed for correspondence filtering [3, 8, 14, 25, 26, 29]. CoFiNet [50] learns feature descriptors in a coarse-to-fine manner. BUFFER [1] takes advantage of both pointwise and patch-wise techniques. GeoTransformer [34] learns geometric feature for robust superpoint matching. Deep Global Registration (DGR) [8] achieves state-of-the-art performance in indoor point cloud registration by introducing Fully Convolutional Geometric Features (FCGF) [9]. DeepVCP [26] leverages virtual points to establish correspondences. Furthermore, Usip [21] and Rskdd-Net [24] employ self-supervised learning methods to generate virtual keypoints and extract descriptor features. Inspired by attentive cost volume [42–44], our objective is to find point correspondences in our registration task. FlowNet3D [23] introduces an embedding layer that learns point correlations in consecutive frames. Wu *et al.* [47] propose a cost volume method for point clouds, incorporating individual point motion patterns. Wang *et al.* further develop the attentive cost volume method, applying it to end-to-end odometry [42,44], which share similarities with registration. Recently, HRegNet [25] achieving superior accuracy and efficiency compared to previous methods. However, Erroneous correspondences can be introduced due to the presence of errors in keypoint descriptors and the utilization of point-to-patch matching strategies.

3. Methodology

The input of HDMNet are source and target point clouds which are standardized to the same number of points. HDMNet makes a prediction of the optimal rotation matrix to align the source point clouds with the target point clouds. The siamese feature pyramid, which will be described in Sec. 3.1, is employed to encode the two point clouds and generate virtual keypoints along with their descriptors. Subsequently, a feature consistency enhanced double-soft matching network is introduced in Sec. 3.2 to achieve global local-to-local alignment via two-stage matching in descriptor space. After that, we incorporate the confidence scores for correspondences obtained from local-to-local registration into the fine registration of the upper layer , which will be explained in Sec. 3.3. After two layers of fine registration, the final estimation $\hat{\mathbf{R}}$ and $\hat{\mathbf{t}}$ are obtained.

3.1. Siamese Point Feature Pyramid

We hierarchically utilize 3-layers of feature extraction, where each layer takes keypoints X_{l-1} (or raw point clouds), uncertainties $\tilde{\Sigma}_{l-1}$, and descriptors $\tilde{D}_{l-1}$ as inputs , and outputs $\tilde{X}_l, \tilde{D}_l, \tilde{\Sigma}_l$. Each level's feature extraction consists of two parts: a keypoint generation module and a descriptor generation module. The detailed network structure can be referred to [24]. Additionally, for each level's input, we select a set of candidate keypoints using the Weighted Farthest Point Sampling (WFPS) [52], which introduces uncertainties for sampling to achieve more reliable keypoint selection. Detailed derivation of this method can be found in [25].

3.2. Global local-to-local Matching

Due to the downsampling method employed, the keypoints in the deepest layer, which are generated by hierarchically aggregating points from local regions in the shallower layers, can be regarded as representations of local regions in the original point cloud and exhibit lower uncertainty and higher reliability. The objective of this module is to achieve local-to-local registration using the sparser keypoints. The pivotal challenge lies in establishing the correspondence between points of two point clouds. However, using a single soft matching strategy to generate matching points by kNN search in a point-to-patch manner fails to consider sufficient information from the source point cloud and not suitable when there are limited overlaps between the two point clouds. To address these issues, we employ a learning-based correspondence network with a feature consistency enhanced double-soft matching model in the local-to-local matching process in layer 3.

3.2.1 Double-soft matching network

Double-soft matching in local-to-local registration. To achieve the double-soft matching strategy in local-to-local registration, as illustrated in Fig. 3, we designed a network consisting of two soft matching modules with identical structures. The double-soft matching network takes keypoints $X^{\mathcal{S}} = \{x_1, x_2, ..., x_i\}_{i=1}^{N}$, $X^{\mathcal{T}} = \{y_1, y_2, ..., y_i\}_{i=1}^{N}$, descriptors $D^{\mathcal{S}}, D^{\mathcal{T}} \in \mathbb{R}^{N \times C}$, and uncertainties $\Sigma^{\mathcal{S}}, \Sigma^{\mathcal{T}} \in \mathbb{R}^{N \times 1}$ as inputs, we omit the subscripts $l = 3$ indicating the layer number in this section. Firstly, for each x_i in $X^{\mathcal{S}}$, we perform kNN search in the source point cloud $X^{\mathcal{S}}$ to find the k_1 nearest neighbor points $\{x_i^1, x_i^2, ..., x_i^{k_1}\}$. Then, for each neighboring point x_i^j, we search for k_2 nearest neighbor points in the target point cloud, forming k_1 point clusters $\{X_i^1, X_i^2, ..., X_i^{k_1}\}$. Each cluster $X_i^j = \{x_i^{j_1}, x_i^{j_2}, ..., x_i^{j_{k_2}}\}$ contains k_2 points, represents the cluster formed by the j-th nearest neighbor points of the i-th point in the source point cloud. In summary, each point x_i in the source point cloud corresponds to k_1 patches of the target point cloud. All the aforementioned kNN searches are conducted in the descriptor space.

The purpose of the first soft matching module is to focus on each cluster X_i^j and aggregate it into a single point to update the target point cloud. This single-soft matching network is inspired by [24, 25], where the features of points within each cluster include similarity features F_S, geometric features F_G and descriptor features F_D. F_G, F_D are described in detail in [25]. In our approach, F_S differs from HRegNet and represents the similarity features between neighboring and center keypoints, incorporating feature consistency similarity and bilateral consensus. Our similarity calculation is simpler and differs between the two soft matching stages, which will be explained in detail in Sec. 3.2.2. We utilize a 3-layer Shared MLP to apply a nonlinear transformation to the Cluster features X_i^j. Subsequent maxpool and the softmax function are used to predict weights $w_i^j = \{w_i^{j_1}, w_i^{j_2}, ..., w_i^{j_{k_1}}\}$ for all candidate keypoint within X_i^j. Then we use w_i^j to compute weighted sum of the points within the cluster. Updated coordinates and descriptors as output is denoted as $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, ...\tilde{x}_{k_1}\}$ and $\tilde{D}$, respectively.

In the second soft matching module, we focus on the k_1 points of $\tilde{X}$ derived from the k_1 clusters in the first soft matching module. Each individual point in $\tilde{X}$ represents a patch within the target point cloud, collectively forming a new cluster. While maintaining a soft matching strategy, the structure of the second soft matching network exhibits similarities to the first module but with some differences: Firstly, the computation of F_S differs. Additionally, by passing the cluster features through a 3-layer Shared-MLP, we obtain a feature map $\bar{F} = \{f_1, ..., f_{K_2}\}$, which will be fed into a new MLP with a sigmoid function to
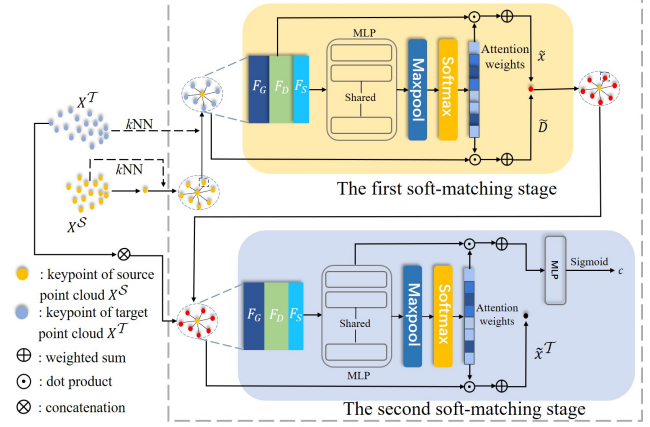

Figure 3. The detailed calculation diagram for our feature consistency enhanced double-soft matching network. The detailed process description is in Sec. 3.2

predict a confidence score $\tilde{c}$ for the correspondence. Finally, the calculation of the optimal transformation $\mathbf{R}^*$ and $\mathbf{t}^*$ can be achieved by solving Eq. (1) in a closed-form using weighted kabsch algorithm [19] based on the corresponding keypoints and confidence scores.

$$\mathbf{R}^*, \mathbf{t}^* = \arg\min_{\mathbf{R}, \mathbf{t}} \sum_{i=1}^{n} \tilde{c}_i \|\tilde{x}_i^{\mathcal{T}} - (\mathbf{R}x_i^{\mathcal{S}} + \mathbf{t})\|^2 \quad (1)$$

where $\|\cdot\|_2$ represents L_2 norm, x_i^S and $\tilde{x}_i^T$ are corresponding points, $\tilde{c}_i$ represents the correspondence confidence. This double soft matching mode adopts a patch-to-patch approach, focusing on k_1 keypoints from the source point cloud and $k_1 k_2$ keypoints from the target point cloud. In comparison to utilizing a single soft matching, it offers enhanced flexibility. Each point in the target point cloud directly participates in two soft matching processes. A higher weight in the first soft matching process indicates its ability to better represent a local region within the target point cloud. Similarly, a higher weight during the second matching process suggests a greater likelihood of this local region matching the point x_i from the source point cloud $X^{\mathcal{S}}$.

Sparse-to-Denser matching strategy. To efficiently implement the Double-soft matching strategy described above and avoid redundant computations caused by the possibility of a point in the target point cloud being a neighbor to multiple points in the source point cloud, in our practical implementation, we reverse the order of kNN searches. We firstly for each point in the source point cloud perform kNN search in the target point cloud to form a cluster, aggregate the cluster into a single point to update the target point cloud. Subsequently, we conduct a same soft matching process between the source point cloud and the updated target point cloud. This computation method enables the implementation of additional sparse-to-denser matching strate-
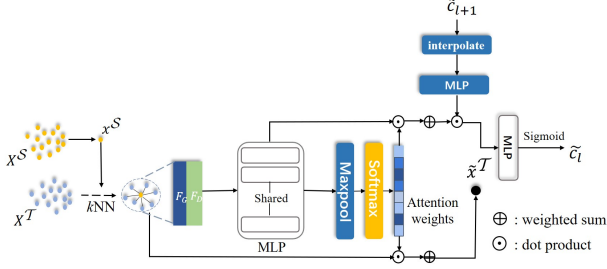
Figure 4. Architecture of correspondence Network in fine registration. Given the source and the target point cloud, along with matching confidence scores of the upper layer, We firstly upsample the confidence scores to obtain an initial mask and utilize the single-soft matching module to generate denser correspondence. The detailed process is described in Sec. 3.3.

gies. Specifically, considering that the target point cloud was updated in the first soft matching model and its structure was also altered, resulting in the loss of some initial information of the target point cloud. Drawing inspiration from ResNet [16], a simple strategy to avoid the aforementioned issue is to concatenate the updated target and original target point cloud before the second round of soft matching. This approach ensures an increase in the density of the target point cloud while maintaining accuracy compared to using a single round of soft matching alone. Consequently, the target point cloud is composed of both the pre-update point clouds and the updated point clouds in the second round of soft matching, leading to sparse-to-denser matching(sparse source point clouds match denser target point clouds).

3.2.2 Similarity Features

Bilateral consensus. Bilateral consensus is used to describe whether the center point of a cluster and its neighboring points are mutual nearest neighbors during the soft matching process, which has been explained in detail in [25, 35]. To ensure efficiency, this similarity is only calculated in the second soft matching module.

Feature consistency similarity. Feature consistency similarity is computed in both soft matching processes. For a point $x_i^{\mathcal{S}}$ in the source point cloud $X^{\mathcal{S}}$ and a cluster $\{x_i^1, x_i^2, ..., x_i^k\}$ in the target or updated target point cloud, we denote their descriptors as d_i^S and d_i^k, respectively. The cosine similarity between them can be calculated as:

$$s_i^k = \frac{d_i^{\mathcal{S}} \cdot d_i^k}{\|d_i^{\mathcal{S}}\| \cdot \|d_i^k\|} \qquad (2)$$

then the similarity is normalized as:

$$s_i^k = \frac{s_i^k}{\max_m s_i^m} \qquad (3)$$

In the first soft matching process, the similarity quantifies the level of similarity between individual points in $X^{\mathcal{S}}$ and their corresponding points in $X^{\mathcal{T}}$, while in the second soft matching process, it measures the level of matching between distinct regions in $X^{\mathcal{T}}$ and keypoint in $X^{\mathcal{S}}$.

3.3. Fine registration with hierarchical mask optimization

Following the coarse registration, we leverage two subsequent layers of fine registration to refine the alignment. In each fine registration layer, we firstly transform the source point cloud using the transformation $\mathbf{R}_{l+1}$, $\mathbf{t}_{l+1}$ obtained from the registration in the deeper layer. A single-soft matching network combined with the weighted kabsch algorithm [19] is employed to obtain refined transformation $\triangle\mathbf{R}_l$ and $\triangle\mathbf{t}_l$. Hence, the output of this layer is given by $\mathbf{R}_l = \triangle\mathbf{R}_l\mathbf{R}_{l+1}$ and $\mathbf{t}_l = \triangle\mathbf{R}_l\mathbf{t}_{l+1} + \triangle\mathbf{t}_l$.

The structure of the fine registration is illustrated in the Fig. 4. We employed the single-soft matching module in the fine registration, as detailed in Section 3.2. Due to the generation of keypoints in the deeper layer from a region in the shallower layer, the high-confidence keypoints in the denser point cloud of the deeper layer correspond to a high-confidence spatial neighborhood region in the shallower layer. We utilized the confidence scores from the deeper layer to design a hierarchical embedding Mask module. Specifically, We adopt the upsampling method inspired by [33] to upsample the confidence scores between layers. For each point x_{l-1}^i in X_{l-1}, we perform a kNN search in X_l. Then, we compute the initial weights c_{l-1}^i for x_{l-1}^i of X_{l-1} as:

$$c_{l-1}^i = \frac{\sum_{j=1}^k w(i,j) \cdot c_l(i,j)}{\sum_{j=1}^k w(i,j)} \qquad (4)$$

where $c_l(i,j)$ represents the j-th nearest neighbor point of x_i in the l-th layer. $w(i,j)$ represents the weight of that point, calculated as the reciprocal of the distance between the two points. The weights are then fed into a MLP, where they undergo a mapping to the feature dimension, enabling the application of a mask to each channel of attentive feature map. In addition, considering the effectiveness of global registration in achieving initial alignment of point clouds, all kNN searches in the fine registration are performed in Euclidean space rather than in descriptor space.

3.4. Loss Function

The loss function of HDMNet is composed of translation loss $\mathcal{L}_{trans}$ and rotation loss $\mathcal{L}_{rot}$. The total loss $\mathcal{L} = \mathcal{L}_{trans} + \alpha\mathcal{L}_{rot}$. $\mathcal{L}_{trans}$ and $\mathcal{L}_{rot}$ can be calculated based on the given estimated and ground truth transformations $\hat{\mathbf{R}}, \hat{\mathbf{t}}$ and $\mathbf{R}, \mathbf{t}$ as

$$\mathcal{L}_{trans} = \|\mathbf{t} - \hat{\mathbf{t}}\|_2 \qquad (5)$$

<table>
<tr><td>(a) Registration result</td><td>(b) Keypoints at each layer</td></tr>
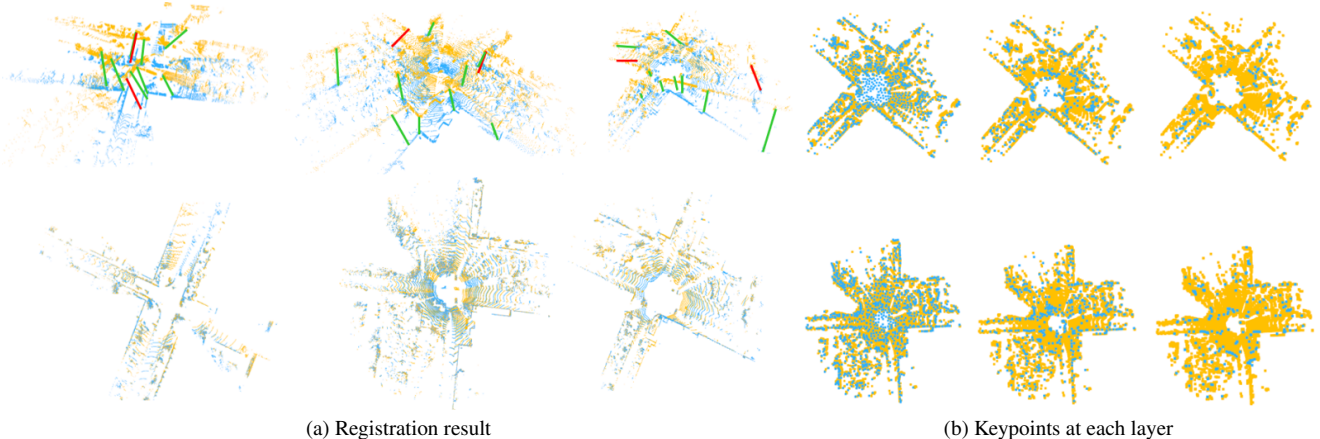</table>

Figure 5. (a) Qualitative visualization of registration result. We display 3 samples of point cloud registration. The first row display the top 5 and bottom 5 point correspondences based on their correspondences confidence obtained from the deepest layer's registration. The green lines and the red lines represent inlier and outlier correspondences. The second raw displays the aligned two point clouds. (b) Qualitative visualization of keypoints. We present the keypoints at each layer, with each row representing one point cloud. The yellow points represent the original point cloud, while the blue points indicate the keypoints generated.

$$\mathcal{L}_{rot} = \|\hat{\mathbf{R}}^{\mathbf{T}}\mathbf{R} - \mathbf{I}\|_2 \qquad (6)$$

where $\mathbf{I}$ denotes identity matrix.

4. Experiments

4.1. Datasets

We conducted experiments on two outdoor LiDAR point cloud datasets: the KITTI odometry dataset [13], the NuScenes dataset [7]. For both datasets, we utilized the point cloud pairs provided in [25] for training, validation, and testing of the proposed method.

4.2. Implementation details

Our HDMNet. In the pre-processing stage, we begin by voxelizing the input point clouds with a voxel size of 0.3m. Subsequently, we randomly sample 16,384 points from the point clouds in the KITTI dataset and 8,192 points in the NuScenes dataset. Our network implementation is based on PyTorch [30] and we employed the Adam optimizer [20] to optimize the network parameters. We initialized the learning rate to 0.0095 and implemented a schedule to decrease it by half every 10 epochs. To achieve a balance between rotation and translation, we set the hyperparameter α to 1.8 for the KITTI dataset and 2.1 for the NuScenes dataset. During the network training process, we follow the approach of [24] to train the detector and descriptor, utilizing the probabilistic chamfer loss as proposed in [21] and the matching loss as introduced in [24]. Subsequently, we fix the weights of the keypoint detector and train the entire network by minimizing the rotation and translation errors. The initial layer contains 1024 keypoints, and each subsequent layer has half

the number of keypoints compared to the upper layer. The descriptor dimension in the first layer is 64, and in each subsequent layer, it is doubled relative to the upper layer. In both coarse and fine registration stages, the value k of kNN search in soft matching module is set to 8.

Baseline method. The performance of the proposed HDMNet is compared with classical methods as well as learning-based methods. All experiments were conducted on an Intel i9-10920X CPU and an NVIDIA RTX 3090 GPU. Following [25], we compare the proposed method with 4 representative traditional registration methods: point-to-point and point-to-plane ICP (ICP (P2P) and ICP (P2Pl)) [6], Fast Global Registration (FGR) [51], RANSAC [11] with FCGF [9] as the feature. The maximum iteration number of RANSAC is set to 2×10e6. Additionally, we compare HDMNet with 5 learning-based methods on KITTI dataset and NuScenes dataset, including two object-level registration methods (Deep Closest Point (DCP) [45] and IDAM [22]), two indoor point cloud registration methods (Feature-metric registration (FMR) [17] and Deep Global Registration (DGR) [8]), and the state-of-the-art LiDAR point cloud registration network HRegNet [25].

4.3. Evaluation

4.3.1 Qualitative visualization

We showcase the final registration results in Fig. 5(a). In the first row, we select the top 5 and bottom 5 point correspondences based on their correspondences confidence obtained from the deepest layer's registration. If the relative positional error is less than the distance threshold $\epsilon_d = 1$m, the corresponding keypoints are considered inliers. This classification is visually represented by the green and red
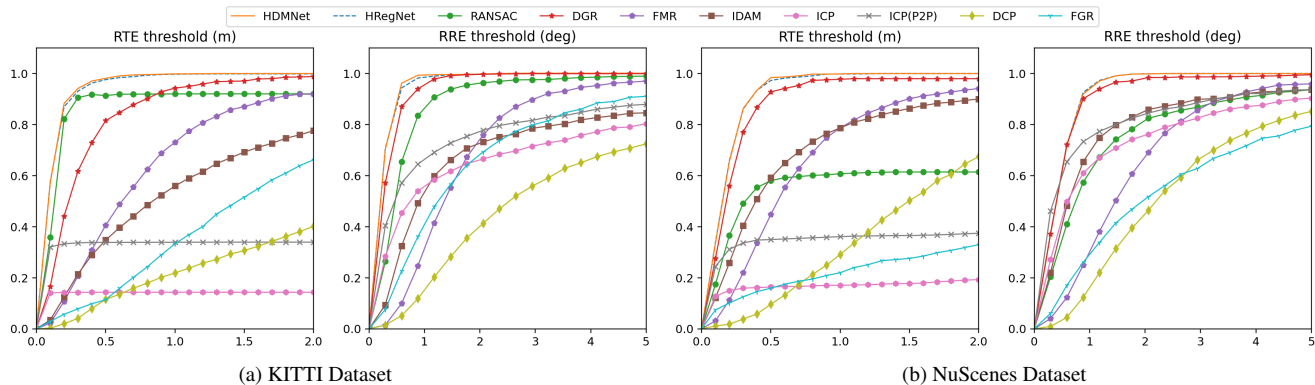
Figure 6. Registration recall with different RRE and RTE thresholds on KITTI dataset and NuScenes dataset.

Table 1. Registration performance on KITTI dataset and NuScenes dataset.

Methods	KITTI dataset				NuScenes dataset			
	RTE(m)	RRE(deg)	Recall	Time(ms)	RTE(m)	RRE(deg)	Recall	Time(ms)
ICP(P2Point) [6]	0.045±0.054	0.112±0.093	14.25%	472.2	0.252±0.510	0.253±0.502	18.78%	82.0
ICP(P2Plane) [6]	0.044±0.041	0.145±0.153	33.56%	461.7	0.153±0.296	0.212±0.306	36.83%	44.5
FGR [51]	0.929±0.592	0.963±0.807	39.43%	506.1	0.708±0.622	1.007±0.924	32.24%	284.6
RANSAC [11]	0.161±0.093	0.424±0.285	100%	459.4	0.171±0.128	0.419±0.257	99.89%	187.4
DCP [45]	1.028±0.506	2.074±1.190	47.29%	46.4	1.087±0.491	2.065±1.142	58.58%	45.5
IDAM [22]	0.659±0.483	1.057±0.939	70.92%	33.4	0.467±0.410	0.793±0.783	87.98%	32.6
FMR [17]	0.657±0.483	1.493±0.847	90.58%	85.5	0.603±0.391	1.610±0.974	92.06%	61.1
DGR [8]	0.322±0.320	0.374±0.302	98.71%	1496.6	0.211±0.183	0.476±0.430	98.41%	523.0
HRegNet [25]	0.056±0.075	0.178±0.196	99.77%	106.2	0.122±0.112	0.273±0.197	100%	87.3
HDMNet	**0.050±0.057**	**0.159±0.152**	**99.85%**	**120.2**	**0.114±0.102**	**0.274±0.206**	**100%**	**102.9**

lines, respectively denoting the inlier and outlier correspondences. We observed that erroneous correspondences are consistently present among the point correspondences ranked in the bottom 5 in terms of confidence, whereas the top 5 correspondences were generally considered inliers. This indicates that the confidence scores predicted by the network have the potential to reject unreliable correspondences. The second row displays the aligned point clouds.

In addition, in Fig. 5(b), we present the keypoints at each layer. It can be observed that as the layers become deeper, the number of keypoints decreases while their reliability increases. The keypoints in the deepest layer are primarily distributed along the road lines, suggesting their higher reliability. This demonstrates the validity of the hierarchical structure and the effective utilization of robust keypoints in the deepest layers for achieving global registration.

4.3.2 Quantitative evaluation

The evaluation of our approach is conducted based on the relative translation error (RTE) and relative rotation error (RRE), which can be calculated as Eq. (5) and $\arccos(\mathrm{Tr}(\mathbf{\hat{R}^T R} - 1)/2)$, respectively. where $\mathbf{\hat{R}}^T$ and $\mathbf{R}$

are the estimated and ground truth rotation matrix. In accordance with [25], we employ the registration recall to measure the success rate of our registration. For a more detailed comparison of the registration performance, we calculate the average RRE, RTE and display the results in Tab. 1. The average RTE and RRE are only calculated for successful registrations. The values of RTE and RRE are set to $\epsilon_{trans} = 2\text{m}$ and $\epsilon_{rot} = 5\text{deg}$, respectively.

The results reveal that without providing a good initial transformation, the ICP algorithm easily falls into local optima, leading to a high failure rate . The recall of FGR is also low, rendering it impractical for applications. The incorporation of the RANSAC algorithm into traditional methods improves both the success rate and accuracy of registration. However, due to multiple iterations, this method's runtime on the KITTI dataset is nearly five times longer than our approach, and its accuracy is also lower than us. Regarding learning-based methods, algorithms designed for indoor or object-level point cloud registration are clearly inadequate for large-scale outdoor scenes. DCP achieves a recall rate of less than 60% on both the KITTI and NuScenes datasets. IDAM performs better than DCP, but its RTE,
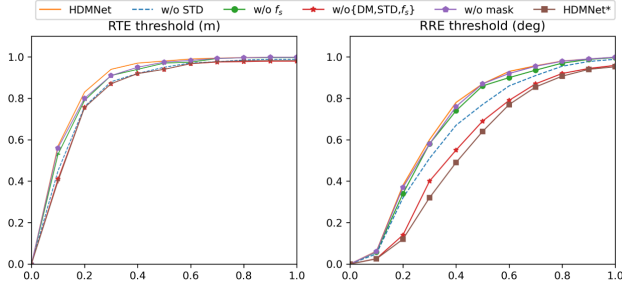
Figure 7. Registration recall of different models on KITTI dataset. STD: Sparse-to-Denser matching strategy. DM: double-soft matching network. f_s: feature consistency similarity.

Table 2. Ablation studies on KITTI dataset.

Model	RTE(m)	RRE(deg)	Recall	Time(ms)
w/o STD	0.064±0.118	0.196±0.311	98.87%	110.6
w/o f_s	0.053±0.086	0.165±0.191	99.68%	118.1
w/o{DM,STD,f_s}	0.071±0.132	0.192±0.272	97.92%	91.8
w/o mask	0.053±0.068	0.163±0.172	99.85%	117.0
HDMNet	**0.050±0.057**	**0.159±0.152**	**99.85%**	**120.2**
HDMNet*	0.076±0.146	0.195±0.297	98.24%	90.1

RRE are still significantly higher than our algorithm. Moreover, the registration success rates of these algorithms range from 50% to 70%, indicating poor feature representation capabilities and limited adaptability to large-scale scenes. FMR does not perform fine registration after global alignment, resulting in lower success rate and accuracy, despite its shorter computation time. DGR has the highest computational cost, but still falls short of achieving sufficient accuracy. HRegNet achieves higher accuracy than other algorithms. However, our proposed HDMNet achieves a reduction of over 12.0% in RTE and 10.7% in RRE on the KITTI dataset, with a higher registration recall rate. Additionally, the RTE and RRE distributions exhibit a smaller standard deviation compared to other methods.

4.4. Ablation study

To analyze the impact of our proposed double-soft matching module, similarity feature, and mask-prediction module on the performance, we conducted extensive ablation studies on the KITTI dataset provided in [25]. The registration recall with different modules is displayed in Fig. 7 and the detailed average RTE and RRE is shown in Tab. 2 and the calculation settings are the same as that in Tab. 1. HDMNet* only utilized a single soft-matching without all the aforementioned modules.

Feature consistency enhanced double-soft matching network. The Feature Consistency Enhanced Double-Soft Matching Network module(DM) incorporates the paradigm of double-soft matching and introduces feature consistency similarity and sparse-to-denser(STD) strategy. To analyze their individual contributions, we conduct ablation studies by separately dropping these three modules and retraining the model. Specifically, in the first experiment, we perform double-soft matching without sparse-to-denser(STD) strategy and in the second experiment we exclude the feature consistency similarity. Additionally, we conducted another experiment using only single-soft matching (without all the aforementioned components).

The results demonstrate that the Feature Consistency Enhanced Double-Soft Matching Network significantly reduces the RRE, RTE and the standard deviation of the error distribution. This indicates that the estimated values become increasingly stable. The registration recall without the similarity and STD strategy is noticeably inferior to the full model, highlighting the importance of these two strategies. Furthermore, the inclusion of STD strategy further improves the performance, indicating that relying solely on double-soft matching may result in the neglect of the distinctive features of the original target keypoints.

Mask-prediction. We drop the Mask Prediction module and retrain the network for comparison with the full model. The experimental results demonstrate the effectiveness of this approach. Specifically, this strategy leads to a reduction in both RRE and RTE while introduces minimal additional parameters and computations, resulting in a nearly unchanged running time. These findings validate the efficacy of this module, as it contributes to the improvement of accuracy without compromising computational efficiency.

5. Conclusion

In this paper, we propose a hierarchical network for outdoor LiDAR point cloud registration, composed of global local-to-local registration and efficient fine registration. To establish reliable correspondences between keypoints, we introduce a double-soft matching network and incorporate feature consistency similarity in the matching process. Additionally, we utilize the confidences of correspondences from deep layer to mask keypoints in the corresponding regions of shallow layers. Abundant ablation experiments demonstrate the effectiveness of our feature consistency enhanced double-soft matching network and mask-prediction module. Furthermore, through extensive experiments on two large-scale outdoor LiDAR point cloud datasets, we have achieved remarkable levels of precision and efficiency with the proposed HDMNet.

References

[1] Sheng Ao, Qingyong Hu, Hanyun Wang, Kai Xu, and Yulan Guo. Buffer: Balancing accuracy, efficiency, and generalizability in point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1255–1264, June 2023. 3

[2] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7163–7172, 2019. 1, 3

[3] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15859–15869, 2021. 3

[4] Daniel Barath, Jiri Matas, and Jana Noskova. Magsac: marginalizing sample consensus. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10197–10205, 2019. 3

[5] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. Magsac++, a fast, reliable and accurate robust estimator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1304–1312, 2020. 3

[6] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. Spie, 1992. 1, 2, 6, 7

[7] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2, 6

[8] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2514–2523, 2020. 1, 3, 6, 7

[9] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8958–8966, 2019. 3, 6

[10] Mahdi Elhousni and Xinming Huang. A survey on 3d lidar localization for autonomous vehicles. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1879–1884. IEEE, 2020. 1

[11] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 3, 6, 7

[12] Alex Flint, Anthony Dick, and Anton Van Den Hengel. Thrift: Local 3d structure recognition. In *9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications (DICTA 2007)*, pages 182–188. IEEE, 2007. 3

[13] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 2, 6

[14] Zan Gojcic, Caifa Zhou, Jan D Wegner, Leonidas J Guibas, and Tolga Birdal. Learning multiview 3d point cloud registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1759–1769, 2020. 1, 3

[15] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, Jianwei Wan, and Ngai Ming Kwok. A comprehensive performance evaluation of 3d local feature descriptors. *International Journal of Computer Vision*, 116:66–89, 2016. 3

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[17] Xiaoshui Huang, Guofeng Mei, and Jian Zhang. Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11366–11374, 2020. 1, 3, 6, 7

[18] Andrew E Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on pattern analysis and machine intelligence*, 21(5):433–449, 1999. 3

[19] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976. 4, 5

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[21] Jiaxin Li and Gim Hee Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 361–370, 2019. 1, 3, 6

[22] Jiahao Li, Changhao Zhang, Ziyao Xu, Hangning Zhou, and Chi Zhang. Iterative distance-aware similarity matrix convolution with mutual-supervised point elimination for efficient point cloud registration. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 378–394. Springer, 2020. 1, 3, 6, 7

[23] Xingyu Liu, Charles R Qi, and Leonidas J Guibas. Flownet3d: Learning scene flow in 3d point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 529–537, 2019. 3

[24] Fan Lu, Guang Chen, Yinlong Liu, Zhongnan Qu, and Alois Knoll. Rskdd-net: Random sample-based keypoint detector and descriptor. *Advances in Neural Information Processing Systems*, 33:21297–21308, 2020. 1, 3, 4, 6

[25] Fan Lu, Guang Chen, Yinlong Liu, Lijun Zhang, Sanqing Qu, Shu Liu, and Rongqi Gu. Hregnet: A hierarchical network for large-scale outdoor lidar point cloud registration.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16014–16023, 2021. 1, 3, 4, 5, 6, 7, 8

[26] Weixin Lu, Guowei Wan, Yao Zhou, Xiangyu Fu, Pengfei Yuan, and Shiyu Song. Deepvcp: An end-to-end deep neural network for point cloud registration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12–21, 2019. 3

[27] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI'81: 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679, 1981. 3

[28] Balázs Nagy and Csaba Benedek. Real-time point cloud alignment for vehicle localization in a high resolution 3d map. In *Proceedings of the european conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 1

[29] G Dias Pais, Srikumar Ramalingam, Venu Madhav Govindu, Jacinto C Nascimento, Rama Chellappa, and Pedro Miraldo. 3dregnet: A deep neural network for 3d point registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7193–7203, 2020. 3

[30] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 6

[31] François Pomerleau, Francis Colas, Roland Siegwart, et al. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends® in Robotics*, 4(1):1–104, 2015. 1

[32] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 3

[33] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 5

[34] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, Slobodan Ilic, Dewen Hu, and Kai Xu. Geotransformer: Fast and robust point cloud registration with geometric transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 3

[35] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. *Advances in neural information processing systems*, 31, 2018. 5

[36] David M Rosen, Luca Carlone, Afonso S Bandeira, and John J Leonard. Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group. *The International Journal of Robotics Research*, 38(2-3):95–125, 2019. 1, 2

[37] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009. 3

[38] Jacopo Serafin and Giorgio Grisetti. Nicp: Dense normal based point cloud registration. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 742–749. IEEE, 2015. 1, 2

[39] Ivan Sipiran and Benjamin Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27:963–976, 2011. 3

[40] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part III 11*, pages 356–369. Springer, 2010. 3

[41] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 78(1):138–156, 2000. 3

[42] Guangming Wang, Xinrui Wu, Shuyang Jiang, Zhe Liu, and Hesheng Wang. Efficient 3d deep lidar odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5749–5765, 2022. 3

[43] Guangming Wang, Xinrui Wu, Zhe Liu, and Hesheng Wang. Hierarchical attention learning of scene flow in 3d point clouds. *IEEE Transactions on Image Processing*, 30:5168–5181, 2021. 3

[44] Guangming Wang, Xinrui Wu, Zhe Liu, and Hesheng Wang. Pwclo-net: Deep lidar odometry in 3d point clouds using hierarchical embedding mask optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15910–15919, 2021. 3

[45] Yue Wang and Justin M Solomon. Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3523–3532, 2019. 1, 3, 6, 7

[46] Yue Wang and Justin M Solomon. Prnet: Self-supervised learning for partial-to-partial registration. *Advances in neural information processing systems*, 32, 2019. 1

[47] Wenxuan Wu, Zhi Yuan Wang, Zhuwen Li, Wei Liu, and Li Fuxin. Pointpwc-net: Cost volume on point clouds for (self-) supervised scene flow estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 88–107. Springer, 2020. 3

[48] Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2020. 2

[49] 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, 2015. 1, 2

[50] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. *Advances in Neural Information Processing Systems*, 34:23872–23884, 2021. 3

[51] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-*

14, 2016, Proceedings, Part II 14, pages 766–782. Springer, 2016. 2, 6, 7

[52] Yao Zhou, Guowei Wan, Shenhua Hou, Li Yu, Gang Wang, Xiaofei Rui, and Shiyu Song. Da4ad: End-to-end deep attention-based visual localization for autonomous driving. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 271–289. Springer, 2020. 3