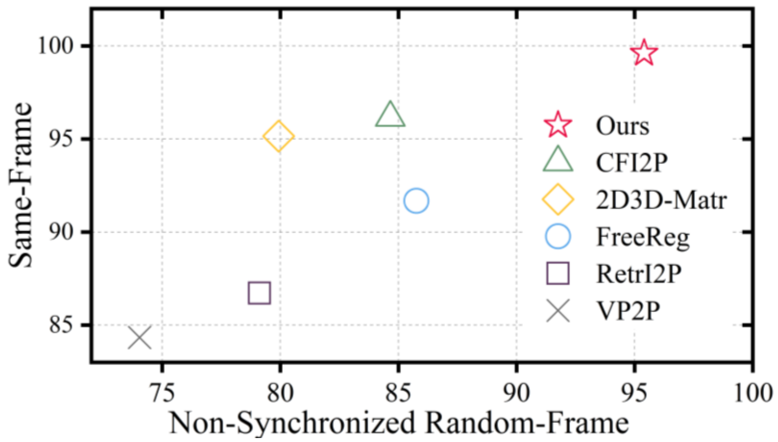
GraphI2P: Image-to-Point Cloud Registration with Exploring Pattern of Correspondence via Graph Learning

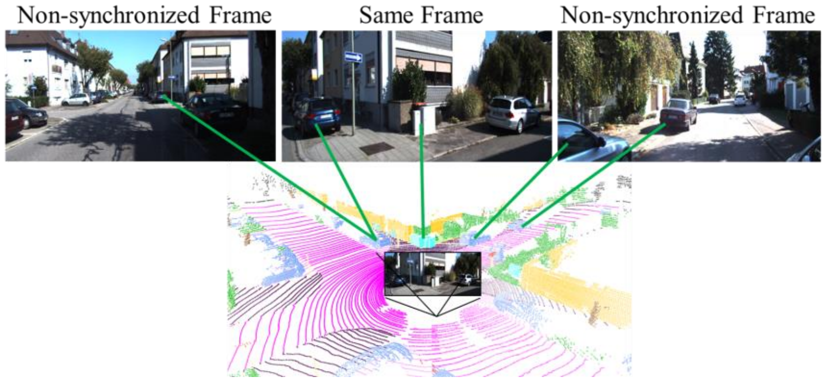
*Lin Bie, Shouan Pan, Siqi Li, Yining Zhao, Yue Gao**

biel20@mails.tsinghua.edu.cn

Introduction

- Image-to-Point Cloud registration between 2D images and 3D LiDAR point clouds is a significant task in computer vision.
- The registration performance is primarily constrained by the accuracy of camera frustum filtering, which neglects the spatial consistency inherent in reliable correspondences.
- Moreover, previous methods are unable to address the registration problem between non-synchronous random frame images and point clouds.
- we propose a graph-based correspondence selection method that develops a global exploration scheme for Image-to-Point Cloud registration task.
- Furthermore, to solve the domain gap issue, we utilize the distribution-based adaptive sample module based on Maximum Likelihood Estimation (MLE) on distance direction to get consistent distribution between LiDAR and virtual point clouds.





Contribution

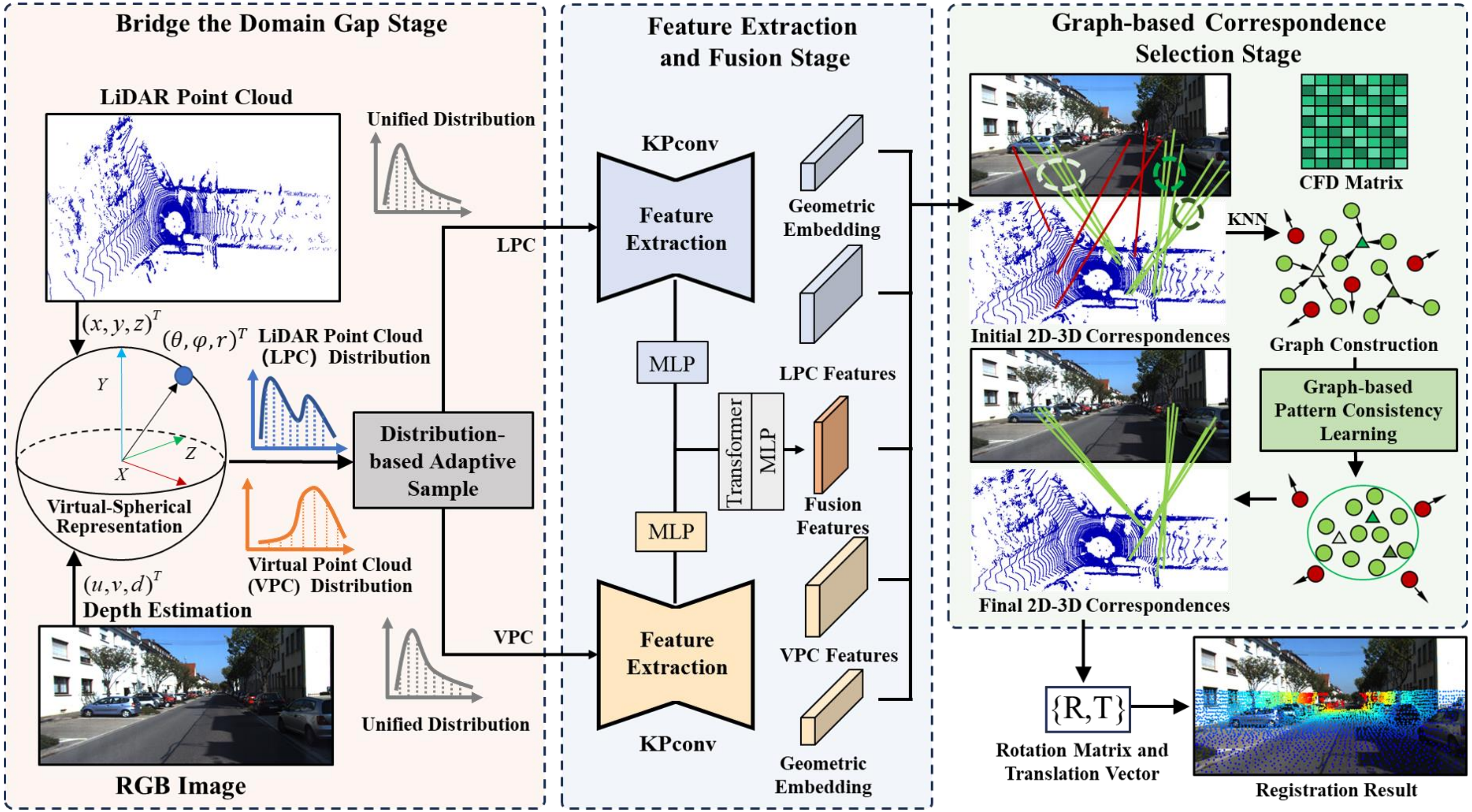
- We introduce a virtual-spherical representation of latent space for cross-modality registration. This enables the virtual point clouds generated from images to exhibit geometric feature representations closer to those of LiDAR point clouds while reducing errors in virtual point clouds caused by scale loss in depth estimation.

- To tackle the domain gap between virtual and LiDAR point clouds, a distribution-based adaptive sample module is proposed to reshape the two types of point clouds in a similar distribution and density for registration.

- We propose a graph-based correspondence selection module to explore the pattern consistency of the correspondence features, which effectively avoids the errors introduced by frustum selection-based methods.

- Extensive experiments on the KITTI Odometry and nuScenes datasets demonstrated superior performance of our proposed method under both the synchronized same-frame and the non-synchronized random-frame settings.

Method


Bridge the Domain Gap Stage
LiDAR Point Cloud
Unified Distribution
LPC
$(x, y, z)^T$
$(\theta, \varphi, r)^T$
LiDAR Point Cloud
(LPC) Distribution
Y
Z
X
Virtual-Spherical
Representation
Distribution-
based Adaptive
Sample
$(u, v, d)^T$
Depth Estimation
Virtual Point Cloud
(VPC) Distribution
VPC
Unified Distribution
RGB Image
Feature Extraction
and Fusion Stage
KPconv
Feature
Extraction
Geometric
Embedding
MLP
LPC Features
Transformer
MLP
Fusion
Features
MLP
VPC Features
Feature
Extraction
KPconv
Geometric
Embedding
Graph-based Correspondence
Selection Stage
CFD Matrix
KNN
Initial 2D-3D Correspondences
Graph Construction
Graph-based
Pattern Consistency
Learning
Final 2D-3D Correspondences
{R,T}
Rotation Matrix and
Translation Vector
Registration Result

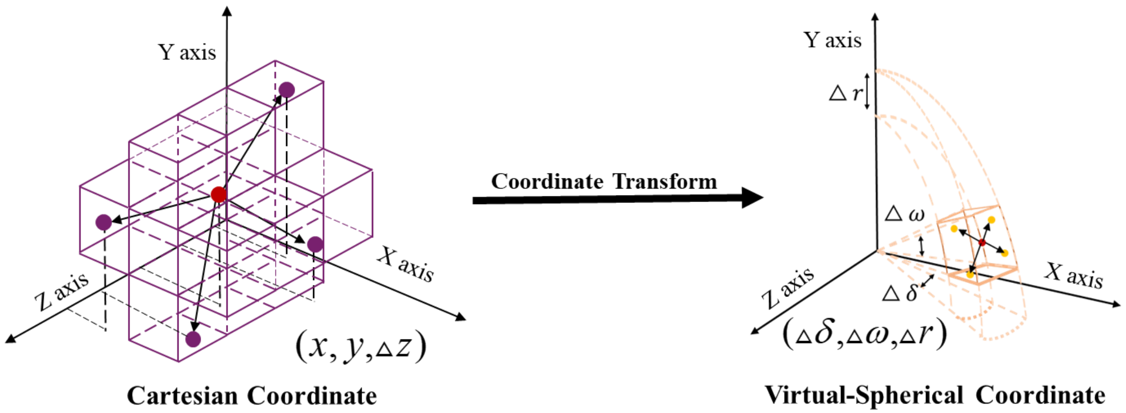
Method

Virtual-Spherical Representation

To perform accurate virtual point cloud generation, we can resort to supervised monocular depth estimation methods to obtain depth maps. Then, virtual points can be converted from depth maps using camera intrinsic parameters.

$$z = D(u,v), x = \frac{(u-c_u)\times z}{f_u}, y = \frac{(v-c_v)\times z}{f_v}.$$

We exploit the 3D spherical representation where the basis is defined by azimuth, elevation, and range. Then, we transform the virtual point cloud 3D location (u, v, d) into (θ, φ, r) where the ϑ is azimuth, φ is elevation, and r stands for the distance the between the point cloud and camera plane.



Method

Distribution-based Adaptive Sample

As the direction and distance of the point cloud are orthogonal, the overall distribution can be represented as a log-normal distribution along the *r* axis in the virtual-spherical representation. The LiDAR PC distribution is:

$$f(x;\mu,\sigma)=\frac{1}{\sqrt{2\pi}\sigma r}e^{\frac{(\ln r-\mu)^2}{2\sigma^2}}$$

Then, we apply the MLE to estimate $\hat{\mu}$ and $\hat{\sigma}$ for LiDAR PC. The log-normal Maximum Likelihood Estimation (MLE) function is:

$$\begin{cases}\dfrac{\partial \ln L(\mu,\sigma^2)}{\partial \mu}=\dfrac{1}{\sigma^2}\displaystyle\sum_{i=1}^{n}(\ln r_i-\mu)=0\\[2ex]\dfrac{\partial \ln L(\mu,\sigma^2)}{\partial \sigma^2}=-\dfrac{n}{2\sigma^2}+\dfrac{1}{2\sigma^4}\displaystyle\sum_{i=1}^{n}(\ln r_i-\mu)^2=0\end{cases}$$

According to above and we get the $\hat{\mu}$ and $\hat{\sigma}$ for the LiDAR point clouds probability density function:

$$\begin{cases}\hat{\mu}=\dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\ln r_i\\[2ex]\hat{\sigma}=\sqrt{\dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\left(\ln r_i-\dfrac{1}{n}\sum_{i=1}^{n}x_i\right)^2}\end{cases}$$

Finally, we sample the LiDAR and virtual point cloud based on the distribution of $LN(\hat{\mu},\hat{\sigma}^2)$, which unified the distribution of two different types of point cloud.

Tsinghua University

Method

Graph-based Correspondence Selection

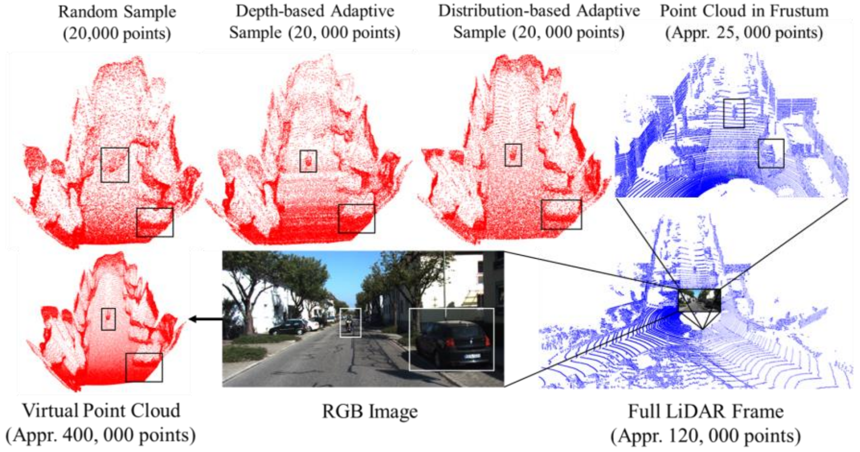
We establish the correspondences according to the point features of two types of point cloud descriptors and re-project them back to the image to generate coarse 2D-3D correspondences. We define the pair-wise feature distance of correspondence as:

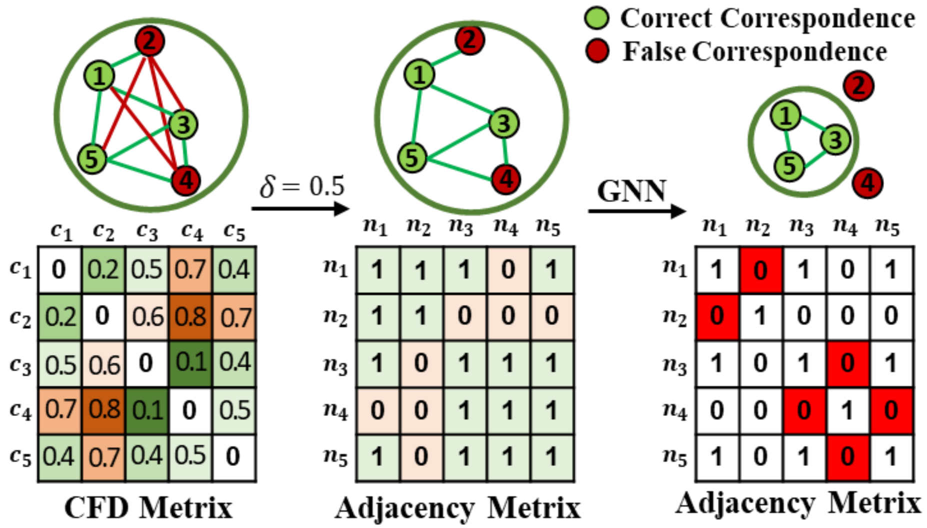
$$S_{dist}(C_i, C_j) = F_s\left(\frac{(f_{C_i} - f_{C_j}) \cdot d_{ij}}{\sigma_{cd}}\right)$$

We conduct KNN based on the correspondence feature distance (CFD) matrix to construct the original graph. We adopt 2 layers of graph neural network (GNN) to make the reliable correspondence features get closer on the same edge and reject false correspondences. The GNN layer is defined as:

$$F^{(l+1)} = \mathrm{ReLU}(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}F^{(l+1)}\mathbf{W})$$

After graph-based learning, the consistencyof the correct correspondences gets better by pruning the false ones in the original graph.





Method

Loss Function

The loss function for our end-to-end training can be divided into two parts: coarse correspondence generation loss and graph-based globe correspondence selection loss. As for the first part, we apply circle loss proposed in to supervise the point-wise position descriptor as follows,

$$\mathcal{L}_{Posn.} = \frac{1}{N}\sum_{k=1}^{N}\log[1+\sum_{i\in c_p}\exp(\lambda_p^i(d_k^i - m_p))\bullet\sum_{j\in\tilde{c}_n}\exp(\lambda_p^j(m_n - d_k^j))]$$

As for the second part, we utilize contrastive loss to supervise graph learning. False correspondences are pruned based on the CFD in the feature space, while correct correspondences have better consistency in pattern. We introduce correspondences selection loss to supervise this process as follows,

$$\mathcal{L}_{Sel.} = -\sum_{i=1}^{N}\log\frac{\exp(W_{y_i} f_g^i + b_{y_i})}{\sum_{k=1}^{K}\exp(W_k f_g^i + b_k)}$$

Then we get the combined loss fuction as,

$$\mathcal{L} = \mathcal{L}_{posn.} + \alpha\mathcal{L}_{Sel.}$$

Experiment

Quantitative Comparison

To evaluate the performance of our proposed Image-to-Point Cloud registration framework, we have conducted experiments on KITTI and nuScenes datasets compared with both Image-to-Point registration methods such as FreeReg and CFI2P, and Point-to-Point registration methods, such as GeoTransformer and PREDATE.

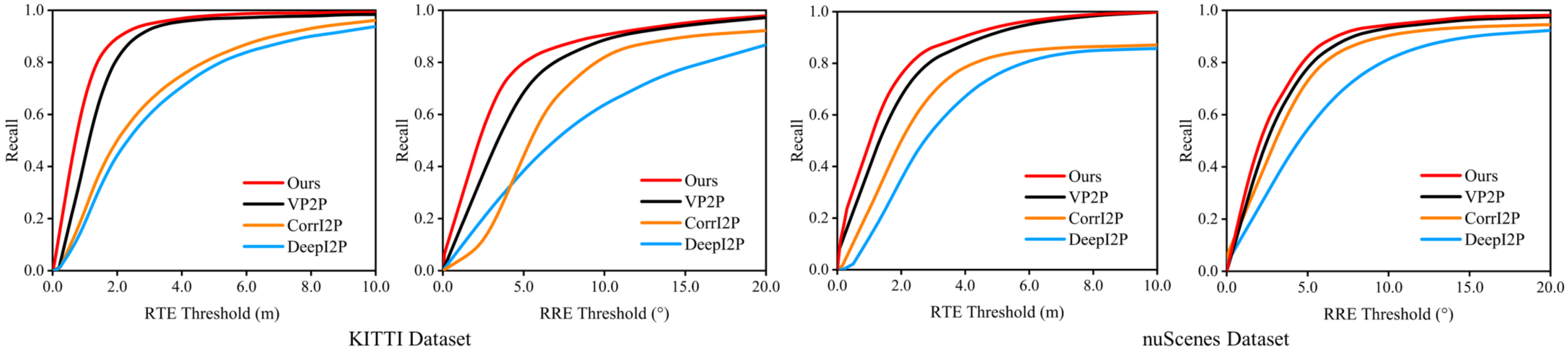
Method	Type	KITTI			nuScenes		
		$RTE(m)\downarrow$	$RRE(°)\downarrow$	RR(%) ↑	$RTE(m)\downarrow$	$RRE(°)\downarrow$	RR(%) ↑
vpc + GeoTransformer[5]	Point-to-Point	4.27 ± 7.14	8.67 ± 8.55	35.28	4.08 ± 2.65	6.30 ± 3.41	16.77
vpc + Hunter	Point-to-Point	4.59 ± 5.22	6.23 ± 5.17	38.61	3.75 ± 4.07	4.22 ± 3.40	31.90
DeepI2P(2D) [9]	Image-to-Point	5.15 ± 7.35	9.14 ± 8.02	10.67	4.85 ± 5.19	7.15 ± 5.82	23.36
CorrI2P [10]	Image-to-Point	4.24 ± 7.26	6.47 ± 5.20	42.31	4.29 ± 5.05	6.40 ± 4.26	36.79
VP2P [15]	Image-to-Point	2.05 ± 3.23	4.01 ± 6.37	73.04	-	-	-
2D-3D Matr [38]	Image-to-Point	1.86 ± 3.79	2.59 ± 4.46	83.93	-	-	-
RetrI2P [14]	Image-to-Point	1.61 ± 2.39	3.16 ± 2.85	79.17	1.73 ± 2.34	3.42 ± 3.41	77.50
FreeReg [13]	Image-to-Point	1.78 ± 1.76	2.89 ± 4.47	83.68	-	-	-
CFI2P [16]	Image-to-Point	1.95 ± 2.97	2.63 ± 3.19	84.63	1.89 ± 3.44	3.20 ± 3.53	81.05
GraphI2P(Ours)	Image-to-Point	$\mathbf{0.68 \pm 0.95}$	$\mathbf{1.91 \pm 2.04}$	**95.40**	$\mathbf{0.83 \pm 1.40}$	$\mathbf{2.21 \pm 3.26}$	**93.58**

The Image-to-Point Cloud registration experimental results on the KITTI and nuScenes datasets. Lower is better for RTE and RRE, higher is better for Accuracy(Acc.). The best results are indicated in bold.

Experiment

Quantitative Comparison

Besides, for a more detailed comparison of the registration performance, we demonstrate the registration recall with different RTE and RRE thresholds on both KITTI and nuScenes datasets.

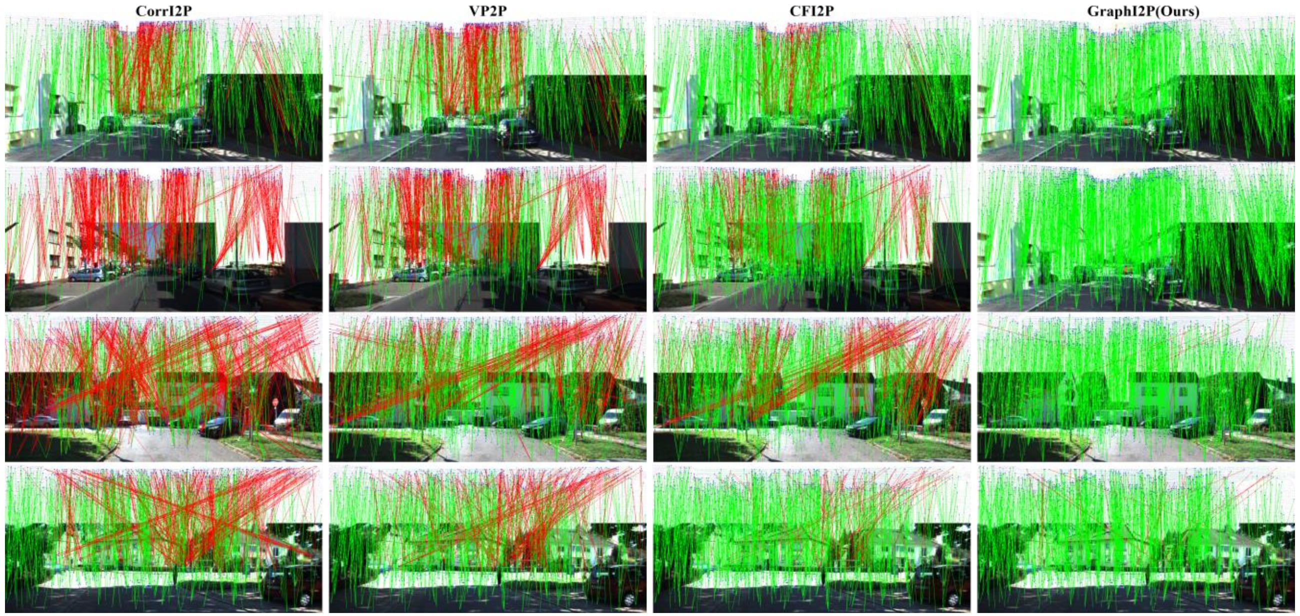


Comparison of the registration recall of different Image-to-Point Cloud registration methods with various RTE and RRE thresholds on KITTI and nuScenes datasets

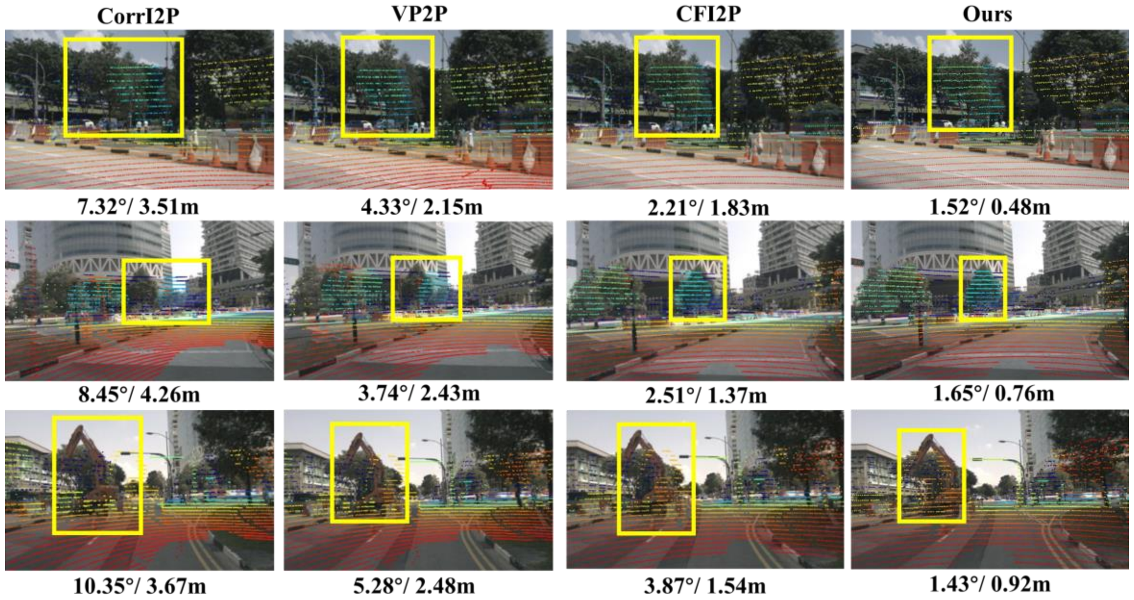
Experiment

Qualitative Comparison

For visual comparison, the point cloud is projected into images with the extrinsic parameters predicted by compared methods on KITTI and nuScenes datasets.



KITTI dataset



nuScenes dataset

Thank you!

biel20@mails.tsinghua.edu.cn