PointMBF: A Multi-scale Bidirectional Fusion Network for Unsupervised RGB-D Point Cloud Registration

Mingzhi Yuan[1,2], Kexue Fu[3*], Zhihao Li[1,2], Yucong Meng[1,2] and Manning Wang[1,2†]
[1]Digital Medical Research Center, School of Basic Medical Sciences, Fudan University, China
[2]Shanghai Key Laboratory of Medical Image Computing and Computer Assisted Intervention, China
[3]Shandong Computer Science Center (National Supercomputer Center in Jinan)
{mzyuan20,fukexue,mnwang}@fudan.edu.cn, {lizhihao21,ycmeng21}@m.fudan.edu.cn

Abstract

Point cloud registration is a task to estimate the rigid transformation between two unaligned scans, which plays an important role in many computer vision applications. Previous learning-based works commonly focus on supervised registration, which have limitations in practice. Recently, with the advance of inexpensive RGB-D sensors, several learning-based works utilize RGB-D data to achieve unsupervised registration. However, most of existing unsupervised methods follow a cascaded design or fuse RGB-D data in a unidirectional manner, which do not fully exploit the complementary information in the RGB-D data. To leverage the complementary information more effectively, we propose a network implementing multi-scale bidirectional fusion between RGB images and point clouds generated from depth images. By bidirectionally fusing visual and geometric features in multi-scales, more distinctive deep features for correspondence estimation can be obtained, making our registration more accurate. Extensive experiments on ScanNet and 3DMatch demonstrate that our method achieves new state-of-the-art performance. Code will be released at `https://github.com/phdymz/PointMBF`.

1. Introduction

Point cloud registration [28] aims at aligning partial views of the same scene, which is a critical component of many computer vision tasks. Commonly, point cloud registration starts from feature extraction [55, 10] and correspondence estimation [46, 43], followed by robust geometric fitting [16, 37, 3, 72]. Among them, feature extraction plays a vital role in point cloud registration, as distinctive features can reduce the occurrence of outlier correspondences, thereby saving time on robust geometric fitting.

Many traditional methods rely on hand-crafted features [55, 34], but they commonly show limited performance. Benefiting from the rapid progress of deep learning, many learning-based features [10, 64, 25] have been proposed in recent years. Compared to hand-crafted features, they are distinctive enough to achieve robust performance in many challenging conditions such as low overlap. However, most deep learning-based features need supervision on poses or correspondences, which limits their practical applications. For unannotated datasets with different distributions from the training set, they tend to suffer from performance degradation.

With the recent advance of inexpensive RGB-D sensors, it has become easier to simultaneously acquire both depth information and RGB images, which inspires unsupervised point cloud registration using additional color information. UR&R [14] proposed a framework for unsupervised RGB-D point cloud registration. It utilizes a differentiable renderer to generate the projections of the transformed point clouds and calculates geometric and photometric losses between the projections and the registration targets. Based on these losses, UR&R can train its deep descriptor without annotations and achieve robust registration on RGB-D video. Similar to UR&R, BYOC [15] proposed a teacher-student framework for unsupervised point cloud registration for RGB-D data, which also shows competitive performance. However, all these RGB-D-based methods use RGB images and depth information separately and do not further exploit the complementary information within RGB-D data. Recently, LLT [67] first utilized a linear transformer [35, 59] to fuse these complementary information and achieved new state-of-the-art performance. However, LLT focuses on using depth information to guide RGB information and neglects the interaction between the two modalities, which hinders better performance.

*Equal contribution.
†Corresponding author.

To fully leverage these complementary modalities, we propose a **m**ulti-scale **b**idirectional **f**usion network named PointMBF for unsupervised RGB-D point cloud registration, which fuses visual and geometric information bidirectionally at both low and high levels. In this work, we process depth images in the form of point clouds and utilize two network branches for RGB images and point clouds, respectively. Both branches follow the U-Shape [54] structure to extract features for information fusion in multiple scales. Unlike the fusion strategy in LLT [67], we perform cross-modalities fusion in all stages rather than only in the last few layers, making fused features more distinctive. Moreover, different from the unidirectional fusion strategy in LLT, we adopt a bidirectional design for more effective fusion. Specifically, in each scale, we first find the regional corresponding points/pixels for each query pixel/point. Then we sample the KNN points/pixels among them and gather their features to a set. The feature set is fed to a PointNet-style module to achieve permutation-invariant aggregation. Finally, the information communication between different modalities can be achieved by fusing the aggregated features with the query feature using a shallow neural network with residue design.

To evaluate our method, we conduct experiments on two popular indoor RGB-D datasets, ScanNet [11] and 3DMatch [73]. Our PointMBF not only achieves new state-of-the-art performance but also shows competitive generalization across different datasets. When tested on an unseen dataset ScanNet, our PointMBF trained on 3DMatch still shows comparable performance to recent advanced methods directly trained on ScanNet. We also conduct comprehensive ablation studies to further demonstrate the effectiveness of each component of our multi-scale bidirectional design.

To summarize, our contributions are as follows:

- We propose a multi-scale bidirectional fusion network for RGB-D point cloud registration, which fully leverages the information in the two complementary modalities. Compared to unidirectional fusion or fusion in the final stage, our fusion strategy can achieve the information communication more effectively, so that it can generate more distinctive features for registration.
- We introduce a simple but effective module for bidirectional fusion, which adapts to density-variant point clouds generated by view-variant depth images.
- We provide a comprehensive comparison between different fusion strategies to analyze their effect empirically.
- Our method achieves new state-of-the-art results on RGB-D point cloud registration on ScanNet [11] using weights trained either on ScanNet or 3DMatch [73].

2. Related Work

2.1. Point Cloud Registration

Point cloud registration aims at aligning partial scan fragments, which is widely used in many tasks such as autonomous driving [44], robotics [49], and SLAM [74]. Except for some ICP-based methods [5, 47], metric-based methods [27, 2, 39], and so on, most methods follow the process of feature extraction [55, 10, 71], correspondence estimation [46, 43], and robust geometric fitting [16, 37, 3]. In the past, many traditional methods were often limited by hand-crafted features [55, 34]. Recently, many learning-based 3D descriptors [10, 8, 1, 4, 70, 52] were proposed. They have achieved impressive performance and some methods [70, 52] are even free of RANSAC. However, most of them rely on pose or correspondence supervision, which limits their practical application. For unannotated datasets, they can only infer using weights trained on other datasets, which tends to degrade their performance. Benefiting from inexpensive RGB-D sensors, many RGB-D video datasets [11, 73] were proposed. The extra color information contains richer semantics and many works achieve unsupervised learning based on it. To the best of our knowledge, UR&R [14] is the first learning-based work using RGB-D data for unsupervised registration. It also follows the above mentioned registration process but it utilizes a differentiable renderer-based loss to optimize its learnable descriptor. Inspired by self-supervised learning [6], BYOC [15] proposed a teacher-student framework for 3D descriptor unsupervised learning. It teaches a 3D descriptor by a 2D descriptor, making the 3D descriptor achieve comparable performance to supervised methods. However, neither of the above two methods fully leveraged the complementary information inside RGB-D data. For UR&R, point clouds are only used for localization but do not participate in feature extraction. For BYOC, their 3D descriptors are limited by their single-modality teacher. To address above the problem, LLT [67] introduced a linear transformer-based attention [35, 59] to embed geometric features into visual features in the last two stages. This fusion improves extracted features and helps LLT achieve the state-of-the-art performance. Whereas we believe unidirectional fusion in late stages does not fully exploit the complementary information in RGB-D data. Therefore, we design a multi-scale bidirectional fusion network, which implements bidirectional fusion in all stages. Benefiting from our fusion strategy, our network can achieve better performance with easily accessible backbones than unidirectional fusion with sophisticated backbones in LLT.

2.2. RGB-D Fusion

RGB image commonly contains rich semantic information, while depth image or point cloud can provide precise

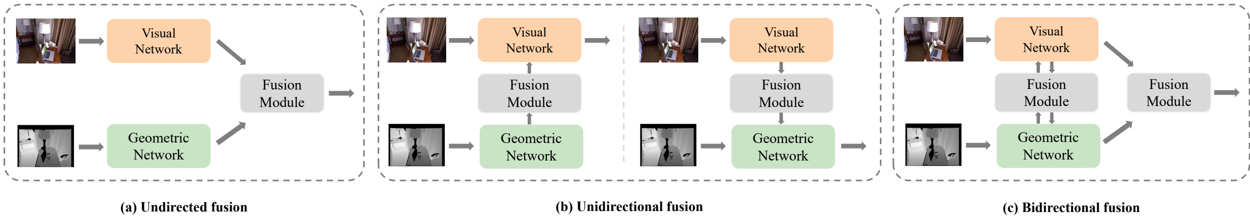


Figure 1. Comparison on fusion strategies.

geometric description. Therefore, fusing these two modalities is a promising direction as they provide complementary information. With the advance of inexpensive RGB-D sensors, many works have studied how to fully leverage this complementary information in many tasks such as detection [40, 61, 62, 68, 42, 26, 41], segmentation [24, 18, 51, 32, 65, 12, 23, 31, 19, 7] and pose estimation [21, 22, 63]. As shown in Figure 1, the common fusion strategies can be roughly divided into three categories according to their information flow direction. The first category is **undirected fusion** [62, 18, 22, 63, 69]. This category is the most intuitive one and is commonly implemented by directly concatenating or adding the separately extracted features. For example, DenseFusion [63] fuses geometric information and texture information by concatenating the embeddings from CNN and PointNet [50] and adding extra channels for global information. The second category is **unidirectional fusion** [40, 68, 42, 26, 67, 51, 32, 65, 12, 30, 29, 19]. This kind of methods usually use one modality to guide the other modality. For instance, DeepFusion [40] sets Lidar features as queries and utilizes a cross-attention-based module called LearnableAlign to embed RGB image features into them. Similar to DeepFusion, LLT [67] adopts a fusion module which is based on linear transformer [35] and fuses high-level features in the last two layers. However, all above methods do not fully exploit the interconnection between different modalities. Therefore, the third category i.e. **bidirectional fusion** [38, 24, 21, 7] was proposed recently. BPNet [24] reveals that joint optimization on different modalities in a bidirectional manner is beneficial to 2D/3D semantic segmentation. It designs a bidirectional projection module to generate a link matrix i.e. the point-pixel-wise map, so that information can interact between two heterogeneous network branches in the decoding stage. FFB6D [21] proposes a network fusing in full stages and outperforms previous methods [22, 63] a lot in pose estimation. Motivated by these success, we believe bidirectional fusion can better leverage the complementary information inside two different domains and propose a bidirectional fusion-based network for RGB-D point cloud registration for the first time.

3. Method

Figure 2 (a) shows the pipeline of our PointMBF, which takes two RGB-D images as inputs and outputs their relative rigid transformation represented by a rotation R^* and a translation t^*. Our PointMBF also follows the standard process of feature extraction, correspondence estimation and geometric fitting. PointMBF first extracts deep features using two heterogeneous network branches for each of the two input RGB-D images, where the visual and geometric features are extracted using different networks and they are fully fused in a bidirectional manner in all stages by fusion modules. Then the fused features are used to generate correspondences based on their Lowe's ratio [43]. Finally, our PointMBF outputs the estimated rigid transformation using these correspondences by a few RANSAC iterations. The correspondence estimation and geometric fitting are free of learnable parameters, and our feature extractor and the fusion modules are trained unsupervisedly by a renderer-based loss. The details of each component of our PointMBF are explained in the following sections.

3.1. Heterogeneous Network Branches

Since there exists a big domain gap between RGB images and depth images, our PointMBF uses two different network branches to process these two modalities separately. As shown in Figure 2 (a), one branch i.e the visual branch takes RGB images as input, while the other i.e. the geometric branch takes point clouds generated from depth images as input. Both branches follows a U-Shape [54] structure to extract multi-scale information, and they are all based on easily accessible backbones including ResNet18 [20] and KPFCN [4, 57]. Since our competitor LLT [67] also has two branches for visual and geometric processing, we introduce the details of our two branches in the following paragraphs and compare them with similar structures in LLT.

Visual branch. LLT designs a dilated convolution-based network as its visual backbone. Although this kind of backbone is competitive, its performance is highly dependent on the hyperparameter setting. To better illustrate the effectiveness of our multi-scale bidirectional fusion strategy

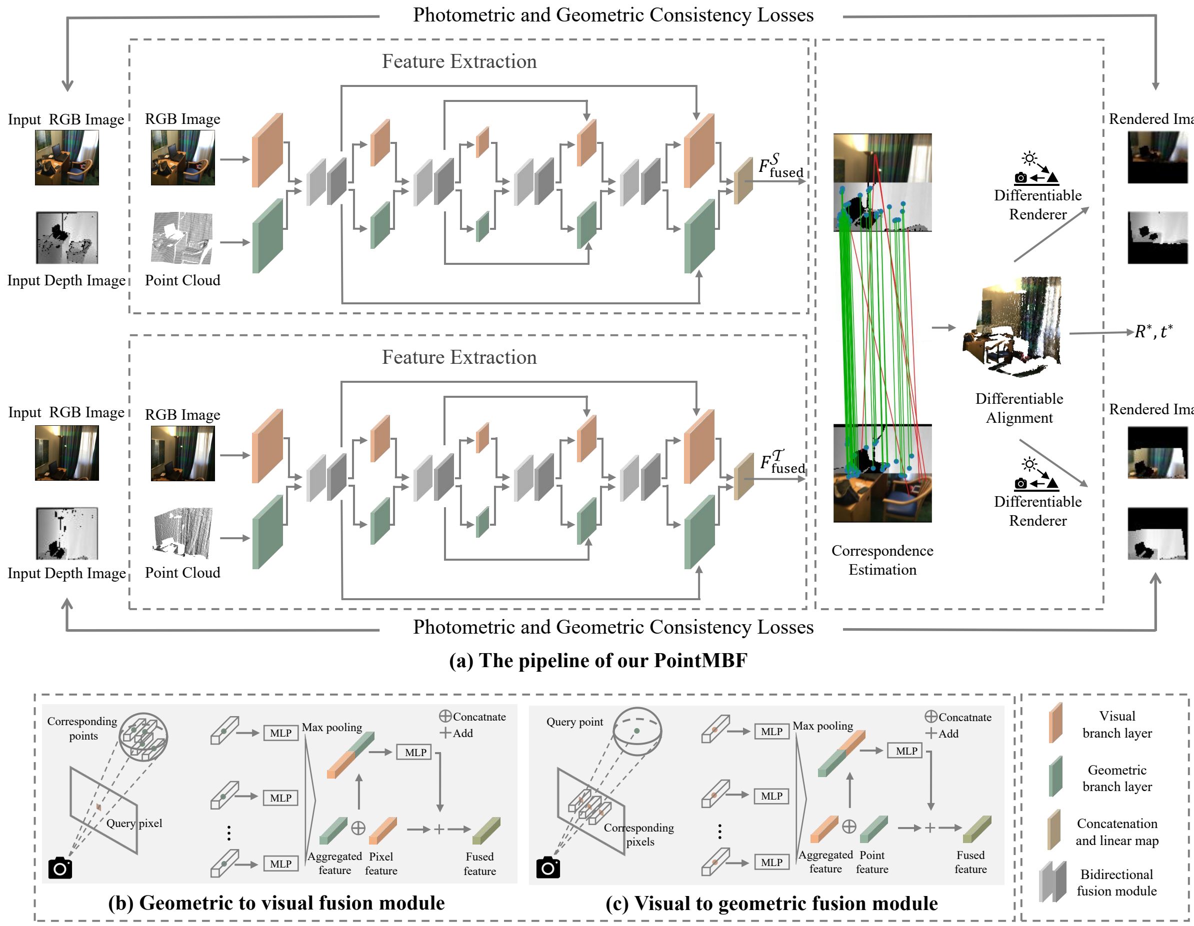


Figure 2. **The overview of our PointMBF.** It takes two RGB-D images as inputs and outputs an estimated rigid transformation. For input RGB-D pairs, it first extracts features using a multi-scale bidirectional fusion-based extractor, which contains two branches and fusion modules (colored in grey) for feature interaction. Then the putative correspondences are determined based on the Lowe's ratio of the extracted features. Once obtaining the correspondences, our model outputs the estimated transformation using several RANSAC iterations. The above model is trained end-to-end by a differentiable renderer.

and save cost on tuning the network architecture, we simply modify a widely used ResNet18 [20] as our visual branch.

As shown in Figure 2 (a), our visual branch follows an U-Shape encoder-decoder architecture with skip connections. Both encoder and decoder extracts features at three different scales. The encoder consists of convolution blocks from ResNet18, while the decoder only contains simple shallow convolution blocks. More details of our visual branch settings are provided in the supplementary materials.

Geometric branch. Different from LLT [67], we process depth images in the form of point clouds rather than the original depth images. There exist many feature extractors for point cloud such as sparse convolution networks [10, 9], point-based networks [50, 66] and so on. However, as shown in Figure 3, there exists severe density variation in the generated point clouds because the sampling density of 3D surfaces is dependent on their distance to the sensor. To extract density-invariant features, we select a shallower KPFCN in D3Feat [4] as the building block of our geometric branch because it introduces a density normalization process to overcome the inherent density variation.

As shown in Figure 2 (a), our geometric branch has a symmetric architecture to the visual branch, so that features from the two branches at the same resolution can be fused and this kind of fusion occurs at every scales. More details of our geometric branch settings are also provided in the supplementary materials.

3.2. Multi-scale Bidirectional Fusion

In this section, we introduce our proposed multi-scale bidirectional fusion in detail. Note that semantics or local geometry are dependent on a certain region rather than a single pixel or point. Therefore, it is intuitive to fuse complementary information by embedding features of a certain region into features of the other modality.

However, embedding regional features faces two challenges. First, as shown in Figure 3, density variation makes the length of regional feature set uncertain. Second, the feature set is not structural data. Inspired by the process for variable length sequence [60] and unstructured data [50], we pad the regional features to a fixed number and design a PointNet-style [50] fusion module for bidirectional fusion. As shown in Figure 2 (b)(c), for a query pixel or a query 3D point at a certain scale, we first find its corresponding region in point cloud/image using the intrinsic matrix of the sensor. Afterward, we sample the KNN corresponding points/pixels in the corresponding region and gather their features to a set. The set is then padded to a certain length and aggregated by a simple PointNet. Since there exists a max-pooling operator in PointNet and grid sampling in the geometric branch, the aggregated feature can achieve density-invariance. Finally, the aggregated feature is further fused with the feature of the query point/pixel by a shallow neural network with residue design. In this way, visual and geometric features can be fully fused in all scales. Besides, as shown in Figure 2 (a), in addition to the above fusion using bidirectional fusion module, we also conduct an undirected fusion in the final stage to further boost the features for correspondence estimation. Details of the visual-to-geometric, the geometric-to-visual, and the final undirected fusion will be introduced in the following subsections.

Visual-to-geometric fusion. Commonly, many ambiguous and repetitive structures exist in point clouds, which makes generated putative correspondences based on only point clouds contain a large proportion of outliers. Incorporating semantic information extracted by the visual branch can make geometric features more distinctive. Here, we utilize visual-to-geometric fusion to embed regional visual features into geometric features.

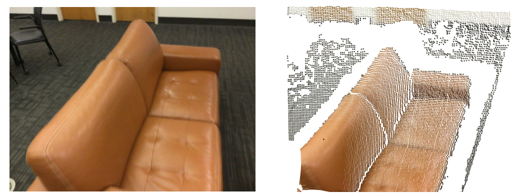
Specifically, given a geometric feature $F^l_{g_i}$ extracted by the geometric branch in the l-th stage for the i-th point, we first find its corresponding region in the image by projecting its neighbor with radius R^l_{v2g} to the image. Then we sample the K_{v2g} nearest neighbor pixels within the corresponding region and gather their visual features $\{F^l_{v_k}\}_{k=1}^{K_{v2g}}$. If there are less than K_{v2g} pixels in the corresponding region, we will pad the null feature $F_{pad} = [0, 0, \ldots, 0] \in \mathbb{R}^{d^l}$ in the gathered features. After that, we use a PointNet-style fusion module to aggregate the regional visual features $\{F^l_{v_k}\}_{k=1}^{K_{v2g}}$:

$$F^l_{v2g_i} = \max_{k=1}^{K_{v2g}} \left(\mathrm{MLP}\left(F^l_{v_k}\right)\right) \tag{1}$$

We then concatenate the aggregated feature $F^l_{v2g_i}$ with the geometric feature $F^l_{g_i}$ and use a linear layer to map them to a fused feature, which has the same dimension as $F^l_{g_i}$. Finally, this fused feature is treated as a residue and added to the original geometric feature $F^l_{g_i}$:

$$F^l_{\text{fused}g_i} = F^l_{g_i} + W^l_{v2g}\left(F^l_{v2g_i} \oplus F^l_{g_i}\right) \tag{2}$$

where $\oplus$ denotes the concatenate operation, W^l_{v2g} denotes a linear map, and $F^l_{\text{fused}g_i}$ denotes the final fused feature, which replaces the original geometric feature $F^l_{g_i}$ to be sent to the next stage. In our bidirectional fusion modules, we adopt a residual design, since our full-stage fusion may cause redundancy. Our following ablation study also verifies it experimentally.



(a) RGB image (b) Generated point cloud

Figure 3. Input RGB image (a) and the point cloud (slightly rotated) generated from the corresponding depth image (b). Severe density variation exists in the generated point cloud, which makes local geometric feature extraction and fusion more challenging.

Geometric-to-visual fusion. Similar to visual-to-geometric fusion, our geometric-to-visual fusion also makes visual features more distinctive. We achieve geometric-to-visual fusion by embedding geometric features into visual features.

Given a visual feature $F^l_{v_i}$ extracted by visual branch in l-th stage for the i-th pixel, we first find its corresponding region in the 3D point clouds by the inverse projection. Then we sample K_{g2v} nearest neighbor points in the corresponding region and gather their geometric features $\{F^l_{g_k}\}_{k=1}^{K_{g2v}}$. We also use the null feature $F_{pad} = [0, 0, \ldots, 0] \in \mathbb{R}^{d^l}$ to pad the gathered features when there are not enough points in the corresponding region and aggregate them into $F^l_{g2v_i}$:

$$F^l_{g2v_i} = \max_{k=1}^{K_{g2v}} \left(\mathrm{MLP}\left(F^l_{g_k}\right)\right) \tag{3}$$

The aggregated feature $F^l_{g2v_i}$ is concatenated with the visual feature $F^l_{v_i}$ and then mapped to a feature, which has the same dimension as $F^l_{v_i}$. Finally, this fused feature is treated as a a residue and added to the original visual feature $F^l_{v_i}$:

$$F^l_{\text{fused}v_i} = F^l_{v_i} + W^l_{g2v}\left(F^l_{g2v_i} \oplus F^l_{v_i}\right) \tag{4}$$

where $\oplus$ denotes the concatenate operation, W^l_{g2v} denotes a linear map, and $F^l_{\text{fused}v_i}$ denotes the final fused feature, which replace the original visual feature $F^l_{v_i}$ to be sent to the next stage.

Undirected fusion. After fully bidirectional fusion in both encoding and decoding stages, we have obtained distinctive features extracted by the visual and geometric branches. To obtain more distinctive features for generating reliable correspondences, we use a simple undirected fusion in the final stage. We concatenate the outputs of both the visual and the geometric branches and fuse them by a linear map:

$$F_{\text{fused}_i} = W_{\text{final}} \left(F^{\text{final}}_{g_i} \oplus F^{\text{final}}_{v_i} \right) \tag{5}$$

where W_{final} denotes a linear map, $F^{\text{final}}_{g_i}$ denotes the geometric feature output by the last layer of geometric branch, $F^{\text{final}}_{v_i}$ denotes the visual feature output by the last layer of visual branch, and F_{fused_i} denotes the final fused features for the following correspondence estimation.

3.3. Correspondence Estimation, Geometric Fitting and Loss Function

Correspondence estimation and geometric fitting. After obtaining the fused features for the source and the target point clouds, we build correspondences using the same method as in UR&R [14] and LLT [67]. Specifically, the correspondences are generated based on the the lowe's ratio [43]. For a point $p^{\mathcal{S}}_i$ in the source point cloud, the lowe's ratio $r^{\mathcal{S}}_i$ is formulated as:

$$r^{\mathcal{S}}_i = \frac{D\left(p^{\mathcal{S}}_i, p^{\mathcal{T}}_{1nn}\right)}{D\left(p^{\mathcal{S}}_i, p^{\mathcal{T}}_{2nn}\right)} \tag{6}$$

where $D(\cdot)$ denotes the Euclidean distance in the feature space and $p^{\mathcal{T}}_{knn}$ is the k-th similar point in the target point cloud. Then we calculate the weight $w = 1 - r$ for each correspondence and select the correspondences with top k weights for source point cloud and target point cloud respectively. The selected correspondences $C = \left\{ \left(p^{\mathcal{S}}, p^{\mathcal{T}}, w\right)_i : 0 \leq i < 2k \right\}$ with their weights are fed into a RANSAC [16] module. The RANSAC module achieves differentiable alignment and outputs an estimated rigid transformation T^* with the minimum error $E(C, T^*)$, where $E(C, T)$ is formulated as:

$$E(C, T) = \sum_{(p^{\mathcal{S}}, p^{\mathcal{T}}, w) \in C} w \left(p^{\mathcal{S}} - T\left(p^{\mathcal{T}}\right) \right)^2 / 2k \tag{7}$$

Loss function. In this work, we use the same loss function as [14, 67] to train the model without the need for annotation. The loss function consists three components:

$$\mathcal{L} = l_{geo} + l_{vis} + \lambda E\left(C, T^*\right) \tag{8}$$

where l_{geo} and l_{vis} denote the geometric and photometric losses based on a differentiable renderer, λ represents a coefficient and we set $\lambda = 0.1$. More details about the loss function can be found in UR&R [14].

4. Experiment

We follow the setting in UR&R [14] and use two indoor RGB-D datasets 3DMatch [73] and ScanNet [11] to conduct our experiments. The following sections are organized as follows. First, we illustrate the details of our experimental settings including datasets, implementation, evaluation metrics, and competitors in section 4.1. Next, we evaluate our method on ScanNet in section 4.2. In this section, we conduct two experiments. The former tests the performance of our method trained on ScanNet [11] and the latter tests our method trained on 3DMatch [73] to verify its generalization. To further understand the effect of our multi-scale bidirectional fusion, we conduct comprehensive ablation studies in section 4.3. We also provide more visualizations and extra experiments in the supplementary material.

4.1. Experimental Settings

Datasets. We use two widely-used RGB-D datasets ScanNet [11] and 3DMatch [73], which contain RGB-D images, camera intrinsics, and ground-truth poses of the camera. For both datasets, we follow settings in [14, 15, 67] to generate view pairs by sampling image pairs which are 20 frames apart. This results in 1594k/12.6k/26k RGB-D pairs for ScanNet and 122k/1.5k/1.5k RGB-D pairs for 3DMatch for train/val/test, respectively.

Implementation. To achieve a fair comparison, we use the same settings as LLT [67] including batch size, learning rate, image size, and so on. We set $K_{v2g} = 16$ for training and $K_{v2g} = 32$ for test. Since pixels are more dense than valid points, we set $K_{g2v} = 1$ to save memory. Before generating point clouds from depth images, we apply the hole completion algorithm [36] to the depth images. Our network is implemented in Pytorch [48] and Pytorch3d [53]. All the experiments are conducted on a single A40 graphic card. For more details of implementation, please see the supplementary material.

Evaluation metrics. Following prior work [14, 15, 67], we evaluate the RGB-D point cloud registration by three evaluation metrics: rotation error, translation error, and chamfer error [45]. For the above metrics, we not only report their mean and median values but also their accuracy under different thresholds.

Competitors. Our competitors can be divided into three categories based on the modalities they use. The first category is only based on point cloud. In addition to previous baselines including ICP [5], FPFH [55], FCGF [10], DGR [8], 3D MV Reg [17] and BYOC [15], we also compare our method to the state-of-the-art point cloud registration method REGTR [70]. We use its officially provided weights, which are obtained by training on 3DMatch, for inference on ScanNet to compare the generalization. The second category is only based on RGB image. It includes many classic baselines such as SIFT [43], SuperPoint [13],

Table 1. **Pairwise registration on ScanNet [11].** Pose Sup indicates the pose or correspondence supervision.

	Train Set	Pose Sup	Rotation (deg)					Translation (cm)					Chamfer (mm)				
			Accuracy↑			Error↓		Accuracy↑			Error↓		Accuracy↑			Error↓	
			5	10	45	Mean	Med.	5	10	25	Mean	Med.	1	5	10	Mean	Med.
ICP [5]	-		31.7	55.6	**99.6**	10.4	8.8	7.5	19.4	74.6	22.4	20.0	8.4	24.7	40.5	32.9	14.1
FPFH [55]	-		34.1	64.0	90.3	20.6	7.2	8.8	26.7	66.8	42.6	18.6	27.0	60.8	73.3	23.3	2.9
SIFT [43]	-		55.2	75.7	89.2	18.6	4.3	17.7	44.5	79.8	26.5	11.2	38.1	70.6	78.3	42.6	1.7
SuperPoint [13]	-		65.5	86.9	96.6	8.9	3.6	21.2	51.7	88.0	16.1	9.7	45.7	81.1	88.2	19.2	1.2
FCGF [10]	-	✓	70.2	87.7	96.2	9.5	3.3	27.5	58.3	82.9	23.6	8.3	52.0	78.0	83.7	24.4	0.9
DGR [8]	3DMatch	✓	81.1	89.3	94.8	9.4	1.8	54.5	76.2	88.7	18.4	4.5	70.5	85.5	89.0	13.7	0.4
3D MV Reg [17]	3DMatch	✓	87.7	93.2	97.0	6.0	1.2	69.0	83.1	91.8	11.7	2.9	78.9	89.2	91.8	10.2	0.2
REGTR [70]	3DMatch	✓	86.0	93,9	98.6	4.4	1.6	61.4	80.3	91.4	14.4	3.8	80.9	90.9	93.6	13.5	0.2
UR&R [14]	3DMatch		87.6	93.1	98.3	4.3	1.0	69.2	84.0	93.8	9.5	2.8	79.7	91.3	94.0	7.2	0.2
UR&R (RGB-D)	3DMatch		87.6	93.7	98.8	3.8	1.1	67.5	83.8	94.6	8.5	3.0	78.6	91.7	94.6	6.5	0.2
UR&R (Supervised)	3DMatch	✓	92.3	95.3	98.2	3.8	**0.8**	77.6	89.4	95.5	7.8	2.3	86.1	94.0	95.6	6.7	**0.1**
BYOC [15]	3DMatch		66.5	85.2	97.8	7.4	3.3	30.7	57.6	88.9	16.0	8.2	54.1	82.8	89.5	9.5	0.9
LLT [67]	3DMatch		93.4	96.5	98.8	**2.5**	**0.8**	76.9	90.2	96.7	**5.5**	2.2	86.4	95.1	96.8	**4.6**	**0.1**
Ours	3DMatch		**94.6**	**97.0**	98.7	3.0	**0.8**	**81.0**	**92.0**	**97.1**	6.2	**2.1**	**91.3**	**96.6**	**97.4**	4.9	**0.1**
UR&R [14]	ScanNet		92.7	95.8	98.5	3.4	0.8	77.2	89.6	96.1	7.3	2.3	86.0	94.6	96.1	5.9	**0.1**
UR&R (RGB-D)	ScanNet		94.1	97.0	99.1	2.6	0.8	78.4	91.1	97.3	5.9	2.3	87.3	95.6	97.2	5.0	**0.1**
BYOC [15]	ScanNet		86.5	95.2	99.1	3.8	1.7	56.4	80.6	96.3	8.7	4.3	78.1	93.9	96.4	5.6	0.3
LLT [67]	ScanNet		95.5	**97.6**	99.1	**2.5**	0.8	80.4	92.2	97.6	**5.5**	2.2	88.9	96.4	97.6	**4.6**	**0.1**
Ours	ScanNet		**96.0**	**97.6**	98.9	**2.5**	**0.7**	**83.9**	**93.8**	**97.7**	5.6	**1.9**	**92.8**	**97.3**	**97.9**	4.7	**0.1**

Table 2. **Single branch performance of our method and LLT [67] (upper rows) and comparison with other fusion strategies (lower rows).** Visual (Ours) and Geo (Ours) denote the visual and geometric branches of our PointMBF, respectively. Visual (LLT) denotes the visual branch in LLT, which is based on the dilated convolution. Visual (RGB-D) denotes our visual branch with an additional channel for depth images. All these networks can resemble augmented version of UR&R [14] with different feature extractors. CAT denotes fusion using direct concatenation. DF denotes fusion using DenseFusion [63]. Trans denotes fusion using transformer [59, 40] in high-level feature space like DeepFusion [40]. Ours wo res denotes removing the residue design in our fusion modules.

	Rotation (deg)					Translation (cm)					Chamfer (mm)				
	Accuracy↑			Error↓		Accuracy↑			Error↓		Accuracy↑			Error↓	
	5	10	45	Mean	Med.	5	10	25	Mean	Med.	1	5	10	Mean	Med.
Visual (Ours)	89.9	94.3	98.4	3.9	1.0	72.4	86.7	94.9	8.4	2.6	82.7	92.8	95.1	6.7	0.2
Geo (Ours)	32.8	61.9	93.4	15.9	7.5	11.7	27.6	62.1	36.1	18.5	24.1	54.0	67.7	21.8	4.1
Visual (LLT [67])	90.4	95.0	98.6	3.6	1.0	70.8	86.5	95.3	8.1	2.8	81.8	93.1	95.4	6.2	0.2
Visual (RGB-D)	85.0	92.1	98.2	4.7	1.1	64.1	80.6	92.7	10.2	3.3	75.8	89.3	92.8	7.7	0.2
CAT	93.1	96.1	**98.7**	3.2	**0.8**	78.5	90.5	96.4	6.7	2.2	89.7	95.7	96.9	5.6	**0.1**
DF [63]	92.9	96.0	98.6	3.3	**0.8**	78.2	90.3	96.3	6.8	2.2	89.2	95.6	96.8	5.4	**0.1**
Trans [40]	91.5	95.2	98.3	3.6	0.9	74.7	88.1	95.6	7.7	2.5	87.3	94.8	96.3	5.6	**0.1**
Ours wo res	94.0	96.6	**98.7**	3.1	**0.8**	80.3	91.3	96.8	6.3	**2.1**	90.7	96.2	97.2	5.3	**0.1**
Ours	**94.6**	**97.0**	**98.7**	**3.0**	**0.8**	**81.0**	**92.0**	**97.1**	**6.2**	**2.1**	**91.3**	**96.6**	**97.4**	**4.9**	**0.1**

and the recently proposed UR&R [14]. To further verify the effectiveness of our method, we also incorporate a supervised version of UR&R as our competitor. The last category is based on RGB-D images. We compare our method to the state-of-the-art method LLT [67] and an RGB-D version of UR&R, which treats depth information as an additional input channel.

4.2. Evaluation on ScanNet

To fully evaluate the proposed method, we train our PointMBF on ScanNet [11] and 3DMatch [73], respectively, and test them on ScanNet. The former experiment closely resembles the cases of processing unannotated datasets, while the latter evaluates the generalization.

Trained on ScanNet. As shown in Table 1, when the training set and test set come from the same domain, our proposed method achieves new state-of-the-art performances on almost all metrics, especially in terms of accuracy under small thresholds. Compared to previous state-of-the-art method LLT [67], our method gains large improvement in translation, which is the bottleneck of the registration on ScanNet. Moreover, by comparing our method with the RGB-D version of UR&R and LLT, we find that the fusion strategy plays an important role in RGB-D point cloud registration. Unidirectional fusion in LLT leverages the complementary information of RGB-D data, but still does not fully exploit them. Our multi-scale bidirectional fusion is a better choice for RGB-D fusion, which can achieve better performance even without sophisticated branches as other methods [67]. This will be further demonstrated in our ablation studies.

Trained on 3DMatch. The generalization results of learning-based methods are also shown in Table 1, where the models are trained on 3DMatch and tested on ScanNet. It can be observed that our method not only achieves the state-of-the-art performance on almost all metrics but also outperforms several recent supervised methods such as REGTR [70] and the supervised UR&R by a large mar-

Table 3. **Ablation on fusion stages.** Encode denotes bidirectional fusion during encoding stage. Decode denotes bidirectional fusion during decoding stage. Concat denotes the concatenation and linear map in the final stage.

Encode	Decode	Concat	Rotation (deg)					Translation (cm)					Chamfer (mm)				
			Accuracy↑			Error↓		Accuracy↑			Error↓		Accuracy↑			Error↓	
			5	10	45	Mean	Med.	5	10	25	Mean	Med.	1	5	10	Mean	Med.
✓			94.1	96.7	**98.7**	3.1	**0.8**	80.1	91.5	96.8	6.4	**2.1**	90.7	96.3	97.3	5.5	**0.1**
	✓		93.9	96.6	98.5	3.2	0.8	79.8	91.5	96.8	6.7	2.2	90.6	96.2	97.2	5.4	**0.1**
		✓	93.1	96.1	**98.7**	3.2	**0.8**	78.5	90.5	96.4	6.7	2.2	89.7	95.7	96.9	5.6	**0.1**
✓		✓	94.3	96.8	**98.7**	**3.0**	**0.8**	80.5	**92.0**	97.0	**6.1**	**2.1**	91.0	96.4	**97.4**	5.1	**0.1**
	✓	✓	94.1	96.5	98.6	3.2	**0.8**	80.6	91.6	96.7	6.5	**2.1**	91.0	96.2	97.2	5.5	**0.1**
✓	✓		94.2	96.7	**98.7**	3.1	**0.8**	80.3	91.7	96.9	6.3	2.2	90.8	96.3	97.3	5.4	**0.1**
✓	✓	✓	**94.6**	**97.0**	**98.7**	**3.0**	**0.8**	**81.0**	**92.0**	**97.1**	6.2	**2.1**	**91.3**	**96.6**	**97.4**	**4.9**	**0.1**

Table 4. **Ablation on fusion direction.** V2G denotes reserving fusion from the visual branch to the geometric branch. G2V denotes reserving fusion from the geometric branch to the visual branch. CAT denotes reserving final undirected fusion i.e. concatenation.

V2G	G2V	CAT	Rotation (deg)					Translation (cm)					Chamfer (mm)				
			Accuracy↑			Error↓		Accuracy↑			Error↓		Accuracy↑			Error↓	
			5	10	45	Mean	Med.	5	10	25	Mean	Med.	1	5	10	Mean	Med.
✓			91.4	95.8	98.4	3.8	1.1	72.6	88.3	96.0	7.8	2.8	87.1	95.2	96.5	5.9	**0.1**
	✓		93.1	96.1	98.5	3.4	**0.8**	78.1	90.4	96.4	7.0	2.3	89.5	95.9	96.8	5.5	**0.1**
		✓	93.1	96.1	**98.7**	3.2	**0.8**	78.5	90.5	96.4	6.7	2.2	89.7	95.7	96.9	5.6	**0.1**
✓		✓	94.1	96.7	**98.7**	3.1	**0.8**	80.3	91.7	96.9	6.3	**2.1**	91.0	96.4	97.3	5.2	**0.1**
	✓	✓	93.4	96.4	**98.7**	3.1	**0.8**	79.2	90.9	96.7	6.4	**2.1**	89.9	96.1	97.1	5.1	**0.1**
✓	✓	✓	**94.6**	**97.0**	**98.7**	**3.0**	**0.8**	**81.0**	**92.0**	**97.1**	**6.2**	**2.1**	**91.3**	**96.6**	**97.4**	**4.9**	**0.1**

gin. Overall, our method shows competitive generalization. What is noticeable is that when the proposed method is trained on 3DMatch and tested on unseen ScanNet, its performance is comparable or even superior to LLT trained directly on ScanNet on some metrics.

4.3. Ablation Studies

To further verify the effectiveness of our multi-scale bidirectional fusion, we conduct comprehensive ablation studies. All the models in our ablation studies are trained on 3DMatch [73] and tested on ScanNet [11].

Comparison with other fusion strategies. As discussed in the related work section, there exist many other fusion strategies including undirected and unidirectional fusion. To fully show the effectiveness of our fusion strategy, we compare our multi-scale bidirectional fusion with many other fusion strategies.

The results are shown in Table 2. It can be seen that our fusion strategy outperforms other strategies including undirected fusion (direct concatenation, DenseFusion [63]) and unidirectional fusion (transformer-based fusion like DeepFusion [40]), indicating the effectiveness of multi-scale bidirectional fusion. Besides, Table 2 also shows the performance of the two single branches of PointMBF and the visual branch of LLT, and according to Table 1 and Table 2, we find our visual branch performs worse than the visual branch of LLT [67] in rotation and our geometric branch has poor performance, but our PointMBF still outperforms LLT. These results strongly suggest that our multi-scale fusion can exploit the complementary information between different modalities more effectively. We also find that the residue design in our fusion module plays an essential role. This is because we use fusion in all stages, which tends to cause redundancy in the network. Moreover, most fusion strategies successfully boost the performances, but fusion by treating depth information as an additional channel input causes performance degradation. We speculate that the shared network can not deal with the big domain gap between RGB and depth. It is more appropriate to use two different networks to handle different modalities. This also reveals that the design of fusion strategy plays a vital role in RGB-D point cloud registration.

Effect of multi-scale fusion. In this work, we fuse information in all stages rather than in the last layers as LLT [67]. We believe that fusion in all stages can promote the exchange of complementary information in multiple scales, making features more distinctive. To verify this, we conduct an ablation on fusion stages.

The results are shown in Table 3. We find fusion in each stage all contributes to the feature extraction. By gradually stacking fusion at different stages, our method finally achieves the best performance. It also can be seen that our bidirectional fusion is powerful as only bidirectional fusion in the encoding or decoding stage shows competitive performance.

Effect of bidirectional fusion. There are three types of information fusion in our proposed framework, namely the multi-scale visual-to-geometric (V2G) fusion, multi-scale geometric-to-visual fusion (G2V) fusion, and the fusion using direct concatenation (CAT) at the end of the two branches. To further confirm the effectiveness of each fusion, we conduct another ablation on fusion directions. Specifically, we reserve one or two of the three fusion types and compare their performance to our whole model. The results are shown in Table 4. When only using one type of fusion, the performance of CAT is similar to that of G2V and they are superior to V2G. On base of CAT, adding either of V2G or G2V can help improve the performance and the highest performance is achieved by adding bidirectional fusion to CAT, as shown in the last row of Table 4.

5. Conclusion

In this work, we propose a multi-scale bidirectional fusion network for unsupervised RGB-D point cloud registration. Different from other networks for RGB-D point cloud registration, our method implements bidirectional fusion in all stages rather than unidirectional fusion only at some stages, which can leverage the complementary information in RGB-D data more effectively. The extensive experiments also show that our multi-scale bidirectional fusion not only helps network achieve new state-of-the-art performance but also outperforms a series of fusion strategies using the same network branches for feature extraction. Furthermore, we believe our multi-scale bidirectional network is a general framework, which can be transferred to more applications such as reconstruction, tracking, etc in the future.

Acknowledgements

This work was supported by the National Natural Science Foundation of China [grant number 62076070], and the Science and Technology Innovation Plan of Shanghai Science and Technology Commission [grant number 23S41900400].

References

[1] Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11753–11762, 2021.

[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.

[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.

[4] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6359–6367, 2020.

[5] 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.

[6] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[7] Xiaokang Chen, Kwan-Yee Lin, Jingbo Wang, Wayne Wu, Chen Qian, Hongsheng Li, and Gang Zeng. Bi-directional cross-modality feature propagation with separation-and-aggregation gate for rgb-d semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI*, pages 561–577. Springer, 2020.

[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.

[9] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[10] 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.

[11] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[12] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–468, 2018.

[13] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018.

[14] Mohamed El Banani, Luya Gao, and Justin Johnson. Unsupervisedr&r: Unsupervised point cloud registration via differentiable rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7129–7139, 2021.

[15] Mohamed El Banani and Justin Johnson. Bootstrap your own correspondences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6433–6442, 2021.

[16] 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.

[17] 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.

[18] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for ob-

ject detection and segmentation. In *European conference on computer vision*, pages 345–360. Springer, 2014.

[19] Caner Hazirbas, Lingni Ma, Csaba Domokos, and Daniel Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part I 13*, pages 213–228. Springer, 2017.

[20] 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.

[21] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3003–3013, 2021.

[22] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11632–11641, 2020.

[23] Ji Hou, Saining Xie, Benjamin Graham, Angela Dai, and Matthias Nießner. Pri3d: Can 3d priors help 2d representation learning? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5693–5702, 2021.

[24] Wenbo Hu, Hengshuang Zhao, Li Jiang, Jiaya Jia, and Tien-Tsin Wong. Bidirectional projection network for cross dimension scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14373–14382, 2021.

[25] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 4267–4276, 2021.

[26] Tengteng Huang, Zhe Liu, Xiwu Chen, and Xiang Bai. Epnet: Enhancing point features with image semantics for 3d object detection. In *European Conference on Computer Vision*, pages 35–52. Springer, 2020.

[27] Xiaoshui Huang, Guofeng Mei, and Jian Zhang. Featuremetric 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.

[28] Xiaoshui Huang, Guofeng Mei, Jian Zhang, and Rana Abbas. A comprehensive survey on point cloud registration. *arXiv preprint arXiv:2103.02690*, 2021.

[29] Xiaoshui Huang, Wentao Qu, Yifan Zuo, Yuming Fang, and Xiaowei Zhao. Gmf: General multimodal fusion framework for correspondence outlier rejection. *IEEE Robotics and Automation Letters*, 7(4):12585–12592, 2022.

[30] Xiaoshui Huang, Wentao Qu, Yifan Zuo, Yuming Fang, and Xiaowei Zhao. Imfnet: Interpretable multimodal fusion for point cloud registration. *IEEE Robotics and Automation Letters*, 7(4):12323–12330, 2022.

[31] Andrej Janda, Brandon Wagstaff, Edwin G Ng, and Jonathan Kelly. Self-supervised pre-training of 3d point cloud networks with image data. *arXiv preprint arXiv:2211.11801*, 2022.

[32] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[33] Wei Jiang, Weiwei Sun, Andrea Tagliasacchi, Eduard Trulls, and Kwang Moo Yi. Linearized multi-sampling for differentiable image transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2988–2997, 2019.

[34] 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.

[35] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020.

[36] Jason Ku, Ali Harakeh, and Steven L Waslander. In defense of classical image processing: Fast depth completion on the cpu. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 16–22. IEEE, 2018.

[37] Junha Lee, Seungwook Kim, Minsu Cho, and Jaesik Park. Deep hough voting for robust global registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15994–16003, 2021.

[38] Haoran Li, Yaran Chen, Qichao Zhang, and Dongbin Zhao. Bifnet: Bidirectional fusion network for road segmentation. *IEEE transactions on cybernetics*, 2021.

[39] Xueqian Li, Jhony Kaesemodel Pontes, and Simon Lucey. Pointnetlk revisited. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12763–12772, 2021.

[40] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17182–17191, 2022.

[41] Zhenyu Li, Zehui Chen, Ang Li, Liangji Fang, Qinhong Jiang, Xianming Liu, Junjun Jiang, Bolei Zhou, and Hang Zhao. Simipu: Simple 2d image and 3d point cloud unsupervised pre-training for spatial-aware visual representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1500–1508, 2022.

[42] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 641–656, 2018.

[43] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[44] 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.

[45] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021.

[46] Guofeng Mei, Xiaoshui Huang, Litao Yu, Jian Zhang, and Mohammed Bennamoun. Cotreg: Coupled optimal transport based point cloud registration. *arXiv preprint arXiv:2112.14381*, 2021.

[47] Hao Men, Biruk Gebre, and Kishore Pochiraju. Color point cloud registration with 4d icp algorithm. In *2011 IEEE International Conference on Robotics and Automation*, pages 1511–1516. IEEE, 2011.

[48] 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.

[49] 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.

[50] 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.

[51] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3d graph neural networks for rgbd semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5199–5208, 2017.

[52] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11143–11152, 2022.

[53] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv preprint arXiv:2007.08501*, 2020.

[54] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[55] 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.

[56] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020.

[57] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019.

[58] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[60] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[61] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4604–4612, 2020.

[62] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11794–11803, 2021.

[63] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3343–3352, 2019.

[64] Haiping Wang, Yuan Liu, Zhen Dong, and Wenping Wang. You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1630–1641, 2022.

[65] Weiyue Wang and Ulrich Neumann. Depth-aware cnn for rgb-d segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 135–150, 2018.

[66] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019.

[67] Ziming Wang, Xiaoliang Huo, Zhenghao Chen, Jing Zhang, Lu Sheng, and Dong Xu. Improving rgb-d point cloud registration by learning multi-scale local linear transformation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 175–191. Springer, 2022.

[68] Zhixin Wang and Kui Jia. Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1742–1749. IEEE, 2019.

[69] Zongwei Wu, Shriarulmozhivarman Gobichettipalayam, Brahim Tamadazte, Guillaume Allibert, Danda Pani Paudel, and Cédric Demonceaux. Robust rgb-d fusion for saliency detection. *arXiv preprint arXiv:2208.01762*, 2022.

[70] Zi Jian Yew and Gim Hee Lee. Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6677–6686, 2022.

[71] Mingzhi Yuan, Xiaoshui Huang, Kexue Fu, Zhihao Li, and Manning Wang. Boosting 3d point cloud registration by transferring multi-modality knowledge. *arXiv preprint arXiv:2302.05210*, 2023.

[72] Mingzhi Yuan, Zhihao Li, Qiuye Jin, Xinrong Chen, and Manning Wang. Pointclm: A contrastive learning-based framework for multi-instance point cloud registration. In *European Conference on Computer Vision*, pages 595–611. Springer, 2022.

[73] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1802–1811, 2017.

[74] Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2174–2181. IEEE, 2015.