RIGA: Rotation-Invariant and Globally-Aware Descriptors for Point Cloud Registration

Hao Yu, Ji Hou, Zheng Qin, Mahdi Saleh, Ivan Shugurov, Kai Wang, Benjamin Busam, and Slobodan Ilic

Abstract—Successful point cloud registration relies on accurate correspondences established upon powerful descriptors. However, existing neural descriptors either leverage a rotation-variant backbone whose performance declines under large rotations, or encode local geometry that is less distinctive. To address this issue, we introduce RIGA to learn descriptors that are Rotation-Invariant by design and Globally-Aware. From the Point Pair Features (PPFs) of sparse local regions, rotation-invariant local geometry is encoded into geometric descriptors. Global awareness of 3D structures and geometric context is subsequently incorporated, both in a rotation-invariant fashion. More specifically, 3D structures of the whole frame are first represented by our global PPF signatures, from which structural descriptors are learned to help geometric descriptors sense the 3D world beyond local regions. Geometric context from the whole scene is then globally aggregated into descriptors. Finally, the description of sparse regions is interpolated to dense point descriptors, from which correspondences are extracted for registration. To validate our approach, we conduct extensive experiments on both object- and scene-level data. With large rotations, RIGA surpasses the state-of-the-art methods by a margin of 8° in terms of the Relative Rotation Error on ModelNet40 and improves the Feature Matching Recall by at least 5 percentage points on 3DLoMatch.

Index Terms—Point cloud registration, rotation-invariant descrip- tors, globally-aware descriptors, coarse-to-fine correspondences.

I. INTRODUCTION

Our entire world is 3D. Modern depth sensors are able to retrieve distance measures of the environment and represent it as point clouds. Naturally, registering point clouds under different sensor poses, a.k.a. point cloud registration, plays a crucial role in a wide range of real applications such as scene reconstruction, autonomous driving, and simultaneous localization and mapping (SLAM). Given a pair of partially-overlapping point clouds, point cloud registration aims to recover the relative transformation between them. As the relative transformation can be solved in closed-form or estimated by a robust estimator [1] based on putative correspondences, establishing reliable correspondences becomes the key to successful registration.

Correspondences are established by matching points according to their associated descriptors. As dense matching is computationally complex, existing works [2], [3], [6], [7], [8], [9], [10], [11], [12], [13] widely adopt a first-sampling-then-matching paradigm to match sparse nodes that are either uniformly-sampled or saliently-detected from dense points. Although the computational complexity is significantly reduced, it introduces a new problem of repeatability, i.e., the corresponding points of some nodes are excluded after sparse sampling s.t. they can never be correctly matched. Due to this design, a considerable part of true correspondences is automatically dropped before matching, which significantly constrains the reliability of putative correspondences. To tackle the problem, we have

Fig. 1. Feature Matching Recall (FMR) on 3DLoMatch [2] (x-axis) and Rotated 3DLoMatch (y-axis). Methods that only encode local geometry are marked as blue, while approaches with global awareness are drawn in red. The performance drop from the original (x-axis) to the rotated (y-axis) benchmark for each method is also demonstrated. Methods that are more robust against rotations are closer to the 45-degree line. Generally, globally-aware methods perform better on standard benchmarks, while rotation-invariant ones (SpinNet [3], YOHO [4] and RIGA) degenerate less under larger rotations. RIGA performs the best in both cases with a drop of only 0.6 percent points.

0162-8828 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
proposed CoFiNet [14] which extracts hierarchical correspondences from coarse to fine. On a coarse scale, it learns to match uniformly-sampled nodes whose vicinities share more overlap. The coarse matching significantly shrinks the space of correspondence search of the consecutive stage, where finer correspondences are extracted from the overlapping vicinities. It implicitly considers all the possible correspondences in the matching procedure and therefore eliminates the repeatability issue. However, the descriptors upon which correspondences are extracted by CoFiNet lack robustness against rotations by design. As a consequence, although reliable correspondences are extracted via the proposed coarse-to-fine mechanism, the performance of CoFiNet still significantly declines when rotations are enlarged, as illustrated in Fig. 1.

This phenomenon reminds us of the importance of point descriptors and shifts our attention to introducing more powerful descriptors for better registration performance. Recent trends widely adopt neural backbones [15], [16], [17] to obtain more powerful descriptors [2], [3], [4], [6], [7], [8], [9], [10], [11], [13], [14], [18], [19], [20] from raw points, which gains significant improvement over handcrafted features [21], [22], [23]. The most recent deep learning-based methods [2], [3], [4], [13], [14] can be split into two categories according to the way they enhance descriptors. The first one [3], [4] aims at promising the rotational invariance of descriptors learned from local geometry by design. For a point \( x_i \in \mathbb{R}^3 \) from point cloud \( \mathcal{X} \), they propose to guarantee that the local descriptor learned from the support area \( \Omega^X_{x_i} \) around \( x_i \), by a model \( G \) is invariant under arbitrary rotations \( \mathbf{R} \in SO(3) \), i.e., \( G(\mathbf{R}(x_i)\mid \mathbf{R}(\Omega^X_{x_i})) = G(x_i\mid \Omega^X_{x_i}) \). According to [3], [7], these methods are more robust to larger rotations, which is also demonstrated in Fig. 1 (see SpinNet, YOHO, and RIGA). The second one [2], [13], [14] instead focuses on incorporating global awareness into local descriptors to enhance the distinctiveness. Compared to descriptors that only encode local geometry, i.e., \( G(x_i\mid \Omega^X_{x_i}) \), the globally-aware descriptor \( G(x_i\mid \mathcal{X}) \) of point \( x_i \) is more distinctive and much easier to be distinguished from other globally-aware descriptors \( G(x_j\mid \mathcal{X}) \) of points \( x_j \) with \( i \neq j \). As illustrated in Fig. 2, it is hard to distinguish two chairs according to local geometry (Fig. 2(b)). However, global awareness helps to separate their description (Fig. 2(c)). Therefore, globally-aware methods usually perform better on the registration task than approaches that only encode local geometry alone, which is also demonstrated in Fig. 1. However, each category of methods has its specific drawback – rotation-invariant descriptors are usually less distinctive due to the blindness to the global context, while globally-aware methods can produce inconsistent descriptions due to the inherent lack of rotational invariance. The current literature lacks an approach that fulfills both aspects simultaneously, i.e., \( G(\mathbf{R}(x_i)\mid \mathbf{R}(\mathcal{X})) = G(x_i\mid \mathcal{X}) \).

To this end, we propose to bridge the lack of globally-aware descriptors that inherently guarantee rotation invariance for the task of point cloud registration with RIGA. Our proposed method simultaneously strengthens the robustness against rotations and distinctiveness of learned descriptors, from which coarse-to-fine keypoint-free correspondences are consecutively extracted. More specifically, we adopt a PointNet [15] architecture, which takes as input the rotation-invariant handcrafted descriptors to encode rotation-invariant local geometry. To provide a node-specific description of the entire scene in a rotation-invariant fashion, we design global PPF signatures that describe each node by considering the spatial relationship of the remaining nodes w.r.t. it. Subsequently, rotation-invariant structural descriptors are learned from global PPF signatures and leveraged to incorporate awareness of global 3D structures into local descriptors. A Transformer [24] architecture is further added, yielding a Vision Transformer (ViT) [25] architecture to incorporate global awareness of geometric context. Finally, dense point descriptors are obtained by interpolation, and the coarse-to-fine mechanism proposed in CoFiNet [14] is extended to extract reliable correspondences from our rotation-invariant and globally-aware descriptors for point cloud registration.
To the best of our knowledge, RIGA is the first to learn both rotation-invariant and globally-aware descriptors for point cloud registration. Our contributions are summarized as:

- We propose an end-to-end pipeline that guarantees the rotational invariance of globally-aware descriptors by design and extracts coarse-to-fine correspondences for point cloud registration.
- We propose global PPF signatures to provide a node-specific description of the entire scene in a rotation-invariant fashion and further learn global structural descriptors from them to incorporate global structural awareness into local descriptors.
- We empirically show the effectiveness of rotational invariance and global awareness on both object- and scene-level data.

This journal paper extends our NeurIPS publication [14] where the coarse-to-fine correspondence extraction is proposed. In this paper, we mainly concentrate on the registration task with larger rotations, which is more challenging, and propose to tackle it by incorporating inherent rotational invariance into globally-aware descriptors. We subsequently introduce RIGA descriptors with the following summary of difference. First, we leverage a ViT architecture to enable joint rotational invariance and global awareness of learned descriptors. Second, we introduce global PPF signatures that describe the global 3D structures in a rotation-invariant fashion. Third, we provide more extensive experiments, especially on rotated benchmarks to prove the superiority of inherent rotational invariance and global awareness for correspondence estimation. Our proposed RIGA descriptors, combined with the coarse-to-fine correspondence extraction, achieve the state-of-the-art performance on both original and rotated benchmarks.

II. RELATED WORK

A. Rotation-Invariant Descriptors

1) Handcrafted Rotation-Invariant Descriptors: Handcrafted rotation-invariant descriptors [21], [22], [23], [26], [27] have been widely explored in 3D by researchers before the popularity of deep neural networks. To guarantee the invariance under rotations, many handcrafted local descriptors [26], [27] rely on an estimated local reference frame (LRF), which is typically based on the covariance analysis of the local surface, to transform local patches to a defined canonical representation. The major drawback of LRF is its non-uniqueness. The constructed rotational invariance is therefore fragile and sensitive to noise. As a result, the attention shifts to those LRF-free approaches [21], [22], [23]. These methods focus on mining the rotation-invariant components of local surfaces and using them to represent the local geometry. Given a point of interest and its adjacent points within the vicinity area, PPF [23] describes each pairwise relationship using euclidean distances and angles among point vectors and normals. In a similar way, PFH [21] and FPFH [22] encode the geometry of the local surface using the histogram of pairwise geometrical properties. Although these handcrafted descriptors are rotation-invariant by design, all of them are far from satisfactory to be applied in real scenarios with complicated geometry and severe noise.

2) Learning-Based Rotation-Invariant Descriptors: Recently, many deep learning-based methods [3], [7], [8] make the attempt to learn descriptors in a rotation-invariant fashion. As a pioneer, PPF-FoldNet [7] encodes PPF patches into embeddings, from which a FoldingNet [28] decoder reconstructs the input. Correspondences are extracted from the rotation-invariant embeddings for registration. Different from PPF-FoldNet [7] that learns from handcrafted LRF-free descriptors, 3DSN [8] leverages LRF, which transforms local patches around interest points to defined canonical representations, to enhance the robustness of learned descriptors against rotations. Similarly, SpinNet [3] and Graphite [11], [12] align local patches according to the defined axes before learning descriptors from them. However, all those methods are limited by their locality, i.e., their descriptors are only learned from the local region where their rotational invariance is defined. Those descriptors are blind to the global context and are therefore less distinctive. Without relying on rotation-invariant handcrafted features, YOHO [4] leverages an icosahedral group to learn a group of rotation-equivariant descriptors for each point. Rotating the input point cloud will permute the descriptors within the group, and rotational invariance is achieved by max-pooling over the group. However, its rotational equivariance is fragile in practice, as the finite rotation group cannot span the infinite rotation space. Additionally, expanding a single descriptor to a group damages efficiency. In object-centric registration, recent methods [23], [29], [30] strengthen the rotational invariance in their learned descriptors by concatenating rotation-invariant descriptors, e.g., PPF [23], with their rotation-variant input. However, as shown in Table I, the registration performance of those methods still drops severely when facing large rotations [30].

In the task of object-centric point cloud classification, there are also works [31], [32], [33] focusing on describing the whole shape as a rotation-invariant descriptor. Although they could generate a shape descriptor to globally depict the shape information, these methods are not globally-aware for learning node/point descriptors. Taking [32] as an example, it leverages graph convolutional networks (GCNs) to expand the receptive fields to larger areas that still remain local. In the task of point cloud registration, the model needs to decode dense point-level descriptors from node descriptors for matching. As the node descriptors are not globally-aware, the point descriptors are also blind to global context. To this end, we propose to adopt the Transformer [24] architecture with the global positions depicted by PFFs for incorporating global awareness to all the node descriptors. As a result, RIGA significantly outperforms [32] when applied to the task of point cloud registration, as demonstrated in Tables IV, V, and VI.

B. Globally-Aware Descriptors

PPF, as an example, has been made semi-global before the widespread of deep neural networks for different tasks [23], [34], [35], [36]. With the widespread of deep neural networks, PPFNet [7] makes the first attempt to incorporate learned global context into their learned descriptors. However, their descriptors are rotation-variant in nature, as the absolute coordinates and PPF features are concatenated as input. Moreover,
III. METHOD

A. Problem Statement

We aim at recovering the rigid transformation $T = \{ R \in SO(3), t \in \mathbb{R}^3 \}$ that best aligns two partially-overlapping point clouds $\mathcal{X} = \{ x_1, \ldots, x_N \}$ and $\mathcal{Y} = \{ y_1, \ldots, y_M \}$. We follow the paradigm of those correspondence-based models [2], [3], [6], [9], [10], [13], [14], [38], [39], where transformation is solved naively leveraging a max-pooling operator for global awareness largely neglects global information beyond each local patch. Predator [2] leverages attention [24] mechanism in a point cloud registration method to strengthen their descriptors with learned global context. Global information is incorporated from the same and the opposite frame, by interleaving Edge Conv-based [37] self-attention modules and Transformer-based [24] cross-attention modules, respectively. Similarly, Yu et al. [14] interleave Transformer-based [24] self- and cross-attention modules for learning globally-aware descriptors. Such a paradigm is also leveraged in the most recent works [13], [38], [39] for incorporating global awareness into local descriptors. However, these methods ignore the inherent rotational invariance of their learned descriptors. As a result, rotational invariance is learned through data augmentation during training, which is intricate for large rotations and adds significant capacity requirements to the deep model.

Fig. 3. Method Overview. Point cloud $\mathcal{X}$ and $\mathcal{Y}$ are processed in the same way, and we only explain for $\mathcal{X}$ hereafter. (1) Local and global PPF signatures are computed for each node $x_i$, which is sparsely sampled from $\mathcal{X}$. Local geometry and global structures are encoded into descriptors $g^{x_i}$ and $s^{x_i}$ by PointNet [15] $\Psi_g$ and $\Psi_s$, respectively. (2) $s^{x_i}$ joins $g^{x_i}$ with global 3D structures via element-wise addition, yielding a globally-informed descriptor $d^{x_i}$. A stack of $K$ attention blocks is leveraged, where intra- and inter-frame geometric context is globally incorporated, resulting in a globally-aware descriptor $d^{x_i \prime}$. (3) Descriptor $d^{x_i \prime}$ of every point $x_i \in \mathcal{X}$ is obtained via interpolation. Node correspondence set $\mathcal{C}$ is retrieved in the Node Matching Module (Fig. 5(a)). In the Matching Refinement Module (Fig. 5(b)), point correspondence set $\hat{\mathcal{C}}$ is extracted according to $\mathcal{C}$ and point descriptors. All the descriptors are invariant to rotations by design.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
and \( n_0 \) of each point \( x_0 \in \Omega_0^{X^*} \) are estimated [40], and the local PPF signature of \( x_i^* \) is represented as a set of PPFs:

\[
S_l(x_i^*|\Omega_i^{X^*}) = \{ξ(x, n, x_i, n_i) | x, n \in \Omega_i^{X^*} \},
\]

with each PPF defined as:

\[
ξ(x, n, x_i, n_i) = (\|d\|_2, \angle(n, d), \angle(n_i, d), \angle(n_i, n_i)),
\]

where \( d \) represents the vector between \( x_i^* \) and \( x_0 \), and \( \angle \) computes the angle between two vectors \( v_1 \) and \( v_2 \), following the way in [6], [34]:

\[
\angle(v_1, v_2) = \tan^{-1}(\|v_1 \times v_2\|_2, v_1 \cdot v_2).
\]

Then, we leverage PointNet [15] to project each local PPF signature to a \( c \)-dimension local geometric descriptor:

\[
g_i^{X^*} = \Psi_g(S_l(x_i^*|\Omega_i^{X^*})), \quad 1 \leq i \leq N^t,
\]

where \( \Psi_g \) stands for a PointNet [15] model shared across all the support areas, and \( c \) is the dimension of learned local descriptors. As a result, each support area is described by a rotation-invariant geometric descriptor of length \( c \).

C. Learning Rotation-Invariant Descriptors From Global 3D Structures

The learned geometric descriptor \( g_i^{X^*} \), defined in (6), is conditioned only on its support area \( \Omega_i^{X^*} \). Consequently, it lacks awareness of the global context and is less distinctive for correspondence search. We consider this the main reason why existing rotation-invariant methods [3], [4], [6], [8] fail to compete with rotation-variant but globally-aware approaches [2], [13], [14]. To address this issue, we propose to enrich local descriptors with global structural cues learned from our global PPF signatures that are invariant to rotations by design.

The design of global PPF signatures is inspired by the handcrafted PPF which is widely used for describing local geometry. For each node \( x_i^* \) with normal \( n_i^* \) (\( 1 \leq i \leq N^0 \)), we compute the structural relationship of every other node \( x_j^* \in X^* \) w.r.t. it (see Fig. 4(b)) by:

\[
S_g(x_i^*|X^*) = \{ξ(x, n, x_i, n_j) | x, n \in X^* \},
\]

which we define as the global PPF signature of node \( x_i^* \). Similar to the conventional PPF, the obtained global PPF signatures are rotation-invariant by design. However, the global PPF signatures are unordered as well. Besides, as the global PPF signatures are conditioned on the whole scene represented by sparse nodes, they can be sensitive to partial overlap, i.e., although some nodes can be occluded in \( x_j^* \), they still contribute to the structural awareness of \( x_i^* \). Therefore, we further leverage a second PointNet [15] architecture \( \Psi_s \) to address both issues simultaneously. The network \( \Psi_s \) projects each global PPF signature to a \( c \)-dimension structural descriptor. This successfully eliminates the inherent unordered property of the global PPF signatures and provides more robustness against partial overlap in real scenes. We denote the obtained structural descriptors as:

\[
s_i^{X^*} = \Psi_s(S_g(x_i^*|X^*) \in \mathbb{R}^c, \quad 1 \leq i \leq N^t.
\]

Each global structural descriptor \( s_i^{X^*} \) will be used to inform its corresponding local geometric descriptor \( g_i^{X^*} \) with global structural information from 3D space.

D. Rotation-Invariant Global Information

1) Incorporating Global Information From 3D Structures:

Following the examples of [2], [14], [41], we interleave self- and cross-attention for intra- and inter-frame global context, respectively. However, the standard attention [24] lacks the awareness of global 3D structures, as it is based purely on the similarity of learned geometry. To this end, we inform each learned local geometric descriptor \( g_i^{X^*} (1 \leq i \leq N^t) \) and \( g_j^{Y^*} (1 \leq j \leq M^t) \) with global structural cues encoded in corresponding global structural descriptor \( s_i^{X^*} \) and \( s_j^{Y^*} \), respectively. The obtained globally-informed descriptors are calculated as (1):

\[
(1) \ d_i^{X^*} = g_i^{X^*} \oplus s_i^{X^*} \quad \text{and} \quad (2) \ d_j^{Y^*} = g_j^{Y^*} \oplus s_j^{Y^*}, \quad \text{where} \ \oplus \ \text{is the element-wise addition.}
\]

2) Global Intra-Frame Aggregation of Geometric Context:

A stack of \( K \) attention blocks operates on globally-informed descriptors to exchange learned geometric information among nodes. Each attention block has an intra-frame module followed by an inter-frame module.

Taking node \( x_i^* \in X^* \) as an example, we detail the computation of the intra-frame module inside the \( l \)-th (\( 1 \leq l \leq K \)) attention block hereafter. Learnable matrices \( W_q \), \( W_k \), and \( W_v \) are introduced to linearly project \( (l-1)d_i^{X^*} \) to \( \text{query, key, and value with:} \)

\[
(1) \ q_i^{X^*} = W_q (l-1) d_i^{X^*}, \\
(2) \ k_i^{X^*} = W_k (l-1) d_i^{X^*}, \\
(3) \ v_i^{X^*} = W_v (l-1) d_i^{X^*},
\]

respectively, where \( q_i^{X^*} \) and \( k_i^{X^*} \) are used for retrieving similar nodes, and \( v_i^{X^*} \) encodes the context for aggregation.

The attention [24] is defined on a node set \( S \in \{X^*, y^*\} \):

\[
(1) \ a_i^{X^*+S} = \text{softmax} \left( \left[ (1) a_1, (2) a_2, \ldots, (|S|) a_{|S|} \right]^T / \sqrt{|c|} \right) \in \mathbb{R}^{|S|},
\]

where \( (i) a_i \) is calculated as \( (i) a_i = (i) q_i^{X^*} (l-1) k_j^{X^*} (1 \leq j \leq |S|) \), and \| \cdot \| \) denotes the set cardinality. The message \( (i) m_i^{X^*+S} \in \mathbb{R}^c \) which flows from set \( S \) to node \( x_i^* \in X^* \), is...
calculated as:

\[
(i)\ m_i^{\mathcal{V}\rightarrow\mathcal{S}} = [(i)\ v_1^S, (i)\ v_2^S, \ldots, (i)\ v_{|\mathcal{S}|}^S], (i)\ a_i^{\mathcal{V}\rightarrow\mathcal{S}} \in \mathbb{R}^c.
\]  

(11)

We globally aggregate the intra-frame learned geometry with:

\[
(i)\ \tilde{d}_i^\mathcal{V} = ((i-1)\ d_i^\mathcal{V} + \text{MLP}\left((i-1)\ d_i^\mathcal{V}, m_i^{\mathcal{V}\rightarrow\mathcal{S}} \right)),
\]

(12)

where MLP is a multilayer perceptron with \(\mathcal{S} = \mathcal{X}'\). For node \(y_j' \in \mathcal{Y}'\), \((i)\ \tilde{d}_j^\mathcal{Y}\) is calculated in the same way according to (12), with but \(\mathcal{S} = \mathcal{Y}'\).

3) Global Inter-Frame Fusion of Geometric Context: For the \(l\)th \(1 \leq l \leq K\) attention block, the inter-frame module takes as input the output of the intra-frame module, i.e., \((i)\ \tilde{d}_i^\mathcal{V}\) and \((i)\ \tilde{d}_j^\mathcal{Y}\). Taking node \(x_i' \in \mathcal{X}'\) as an example, similar to (9), \((i)\ \tilde{d}_i^\mathcal{V}\) is linearly projected by learnable matrices \((i)\ W_q, (i)\ W_k\), and \((i)\ W_v \in \mathbb{R}^{c \times c} : \)

\[
\begin{align*}
(i)\ \tilde{d}_i^\mathcal{V} & = ((i)\ \tilde{d}_i^\mathcal{V}, (i)\ m_i^{\mathcal{V}\rightarrow\mathcal{S}}), \\
(i)\ \tilde{d}_i^\mathcal{V} & = (i)\ W_q (i)\ d_i^\mathcal{V}, \\
(i)\ \tilde{d}_i^\mathcal{V} & = (i)\ W_k (i)\ d_i^\mathcal{V}, \\
(i)\ \tilde{d}_i^\mathcal{V} & = (i)\ W_v (i)\ d_i^\mathcal{V},
\end{align*}
\]

(13)

upon which \((i)\ \tilde{d}_i^\mathcal{V}\) and \((i)\ \tilde{m}_i^{\mathcal{V}\rightarrow\mathcal{S}}\) are computed following (10) and (11), respectively, with \(\mathcal{S} = \mathcal{Y}'\). Finally, the geometric context from the opposite frame, i.e., the node set \(\mathcal{Y}'\), is fused to node \(x_i'\):

\[
(i)\ d_i^\mathcal{V} = ((i)\ \tilde{d}_i^\mathcal{V} + \text{MLP}\left((i)\ \tilde{d}_i^\mathcal{V}, m_i^{\mathcal{V}\rightarrow\mathcal{S}} \right)),
\]

(14)

with \(\mathcal{S} = \mathcal{Y}'\). For node \(y_j' \in \mathcal{Y}'\), \((i)\ \tilde{d}_j^\mathcal{Y}\) is calculated in the same way according to (14), but with \(\mathcal{S} = \mathcal{X}'\).

Since all the operations are performed in feature space, the rotation-invariance of \((i)\ d_i^\mathcal{X}\) remains in all \((i)\ \tilde{d}_i^\mathcal{X}\) and \((i)\ d_i^\mathcal{V}\) with \(1 \leq l \leq K\). As a result, the obtained globally-aware descriptor \(d_i^{\mathcal{V}} := (K)\ d_i^\mathcal{V}\) is rotation-invariant by design. Similarly, globally-aware descriptor \(d_j^{\mathcal{Y}} := (K)\ d_j^\mathcal{Y}\) is also rotation-invariant for each \(y_j' \in \mathcal{Y}\).

E. Rotation-Invariant Dense Description

Until here, we have successfully incorporated global awareness into learned local descriptors of nodes without sacrificing the inherent rotational invariance. The aforementioned repetatability issue of sparsely sampled nodes, however, still remains. To address this issue, we leverage the coarse-to-fine strategy proposed in [14], where nodes are first matched according to the overlap ratios of their vicinities, and point correspondences are then extracted from the vicinities of matched nodes. As the first step, dense point descriptors are generated via interpolation. For each point \(x_u \in \mathcal{X}\), we find its \(k\)-nearest neighbor nodes in \(\mathcal{X}'\) according to their euclidean distance. The descriptor \(d_u^\mathcal{X}\) of point \(x_u\) can be interpolated as:

\[
d_u^\mathcal{X} = \sum_{i=1}^{k} w_i^u \cdot d_i^\mathcal{X}, \quad \text{with } w_i^u = \frac{1/d_i^u}{\sum_{l=1}^{k} 1/d_l^u},
\]

(15)

where \(d_i^u\) depicts the euclidean distance of point \(x_u\) to its \(l\)th nearest node in geometry space. Point descriptor \(d_i^\mathcal{Y}\) of \(y_v \in \mathcal{Y}\) is calculated in the same way. As the interpolation coefficients are only related to euclidean distance, the obtained point descriptors remain invariant to rotations.

F. Coarse-to-Fine Correspondence Extraction

The coarse-to-fine mechanism [14] is leveraged to extract correspondences from our obtained node and point descriptors. We first project \(d_i^\mathcal{X}\) and \(d_u^\mathcal{Y}\) by using two individual multilayer perceptrons (MLP), which provides \(d_i^{\mathcal{X}'}\) and \(d_u^{\mathcal{Y}'}\). On the coarse level, as shown in Fig. 5(a), the similarity between node \(x_i' \in \mathcal{X}'\) and \(y_j' \in \mathcal{Y}'\) is calculated as \(1/||d_i^{\mathcal{X}'} - d_j^{\mathcal{Y}'}||^2\). As the following step, Top-K node correspondences with the highest similarity values are sampled, resulting in the node correspondence set \(\mathcal{C} \subseteq [C] \) correspondences. In “Grouping” of Fig. 5(b), vicinities \((V_i^{\mathcal{X}'}, V_j^{\mathcal{Y}'})\) of coarse correspondence \(C_l := (x_i', y_j') \in \mathcal{C}\) are collected by the point-to-node assignment [14], [43], i.e., assigning points to their nearest nodes in geometry space. For node \(x_i'\), its vicinity \(V_i^{\mathcal{X}'}\) and the associated descriptor group \(D_i^{\mathcal{X}'}\) can be defined as:

\[
\begin{align*}
V_i^{\mathcal{X}'} & = \{x_u \in \mathcal{X}': ||x_u - x_i'||_2 < ||x_u - x_j'||_2, \forall j \neq i\}, \\
D_i^{\mathcal{X}'} & = \{d_u^{\mathcal{X}} | d_u^{\mathcal{X}} \leftrightarrow x_u \in V_i^{\mathcal{X}'}\},
\end{align*}
\]

(16)

where \(d_u^{\mathcal{X}} \leftrightarrow x_u\) denotes that \(d_u^{\mathcal{X}}\) is the descriptor associated to point \(x_u\). \(V_j^{\mathcal{Y}'}\) and \(D_j^{\mathcal{Y}'}\) are defined in the same way for nodes \(y_j' \in \mathcal{Y}'\). Finally, we present the similarity of \((D_i^{\mathcal{X}'} , D_j^{\mathcal{Y}'})\) as a matrix \(S_l \in \mathbb{R}^{|D_i^{\mathcal{X}'}| \times |D_j^{\mathcal{Y}'}|}\), where each entry is calculated as \(S_{l_{uv}} = (d_u^{\mathcal{X}'})^T \cdot d_v^{\mathcal{Y}'}\), with \(d_u^{\mathcal{X}'} \in D_i^{\mathcal{X}'}\) and \(d_v^{\mathcal{Y}'} \in D_j^{\mathcal{Y}'}\). To deal with partial overlap, we follow the slack idea [41] and augment \(S_l\) with an additional row and an additional column filled with the same learnable parameter \(\alpha\). In “Sinkhorn” of Fig. 5(b), each augmented similarity matrix is normalized to a confidence matrix \(\hat{Z}_l \in \mathbb{R}^{(D_i^{\mathcal{X}'} + 1) \times (D_j^{\mathcal{Y}'} + 1)}\), which is a non-negative matrix with every row and every column summing to 1, with the Sinkhorn [42] algorithm. From \(\hat{Z}_l\) we extract the point correspondence set \(\hat{C}_l\) as the maximum confidence individually for each row and column. The union of all \(\hat{C}_l\) (\(1 \leq l \leq |\mathcal{C}|\)) constructs the final point correspondence set \(\hat{C}\), which we use for registration.

G. Loss Functions

The total loss function \(L = L_c + \lambda L_f\) consists of a coarse-level matching loss \(L_c\) and a fine-scale correspondence refinement loss \(L_f\). \(\lambda \in \mathbb{R}\) is the hyper-parameter used to balance the two terms.

1) Coarse-Level Loss for Node Matching: Following [14], our coarse-level loss is defined according to the overlap ratios of the vicinities \((V_i^{\mathcal{X}'}, V_j^{\mathcal{Y}'})\) of each node correspondence \((x_i', y_j')\). Given vicinities \((V_i^{\mathcal{X}'}, V_j^{\mathcal{Y}'})\) of node correspondence \((x_i', y_j')\), the number of visible points in one vicinity w.r.t. the other vicinity
is defined as:

\[ n^i_j = \sum_{x_u \in V^i} 1(\exists y_v \in V^j \text{s.t.} \|T(x_u) - y_v\|_2 < \tau_p), \quad (17) \]

and

\[ n^i_j = \sum_{y_v \in V^j} 1(\exists x_u \in V^i \text{s.t.} \|T(x_u) - y_v\|_2 < \tau_p), \quad (18) \]

denote \(Y^i_X\) and \(Y^j_Y\), respectively, where \(\tau_p \in \mathbb{R}\) is the distance threshold for correspondence decision. The overlap ratio between \(Y^i_X\) and \(Y^j_Y\) is further defined as

\[ r^i_j = \frac{n^i_j}{n^i + n^i_j}. \]

Similar to [2], [10], [39], we use Circle Loss [44], a variant of Triplet Loss [45], to guide the learning of node descriptors. For a node \(x^i_x\) from \(X^i\), we sample a positive set \(C^i_p\) composed of nodes \(y^j_y\) from \(Y^j\) s.t. \(T(Y^i_X\) overlaps with \(Y^j_Y\), and a negative set \(C^i_n\) consisting of nodes \(y^j_y\) from \(Y^j\) s.t. \(T(Y^i_X)\) and \(Y^j_Y\) share no overlap, where \(T(Y^i_X)\) denote \(Y^i_X\) transformed by the ground truth transformation \(T\). The loss function on \(X^i\) can be defined upon \(n\) nodes \(x^i_x\) sampled from \(X^i\) as:

\[ \mathcal{L}^i_c = \frac{1}{n} \sum_{i=1}^n \log \left[ \frac{1}{1 + \sum_{j=0}^{n} e^{r^i_j \beta^i_p (d^i_j - \Delta_p)} + \sum_{j'=0}^{n} e^{r^i_j \beta^i_n (\Delta_n - d^i_j')}} \right], \quad (19) \]

where \(r^i_j\) is the overlap ratio between \(Y^i_X\) and \(Y^j_Y\), and \(d^i_j = \|d^i_j - d^i_j\|_2\) denotes the euclidean distance of nodes \(x^i_x\) and \(y^j_y\) in learned feature space. \(\Delta_p\) and \(\Delta_n\) are the positive and negative margins, which are set to 0.1 and 1.4 in practice, respectively.

Furthermore, \(\beta^i_p = \gamma(d^i_j - \Delta_p)\) and \(\beta^i_n = \gamma(\Delta_n - d^i_j)\) are the weights determined for each sample individually, with the same hyper-parameter \(\gamma \in \mathbb{R}\). We can similarly define the loss \(\mathcal{L}^i_c\) and write the total coarse-level loss as \(\mathcal{L}_c = \frac{1}{2}(\mathcal{L}^i_c + \mathcal{L}^j_c)\).

2) Fine-Level Loss for Correspondence Refinement: After getting the coarse correspondence set \(C\), we adopt a negative log-likelihood loss [41] to guide the correspondence refinement procedure. For node correspondence \(C^i := (x^i_x, y^j_y) \in C\), as mentioned before, we compute its confidence matrix \(\hat{Z}_i \in \mathbb{R}^{|D^i|+1}\) augmented with a slack row and slack column for no correspondence. The ground truth point correspondence set between \(Y^i_X\) and \(Y^j_Y\) is denoted as \(\mathcal{M}_i\), while the sets of unmatched points in vicinity \(Y^i_X\) and \(Y^j_Y\) are represented as \(\mathcal{I}_i\) and \(\mathcal{J}_i\), respectively. The ground truth point correspondence set between \(Y^i_X\) and \(Y^j_Y\) is defined as:

\[ \mathcal{M}_i = \{ (x_u \in V^i, y_v \in V^j) ||T(x_u) - y_v\|_2 < \tau_p \}. \quad (20) \]

The set of occluded points in one vicinity w.r.t. the other one is defined as:

\[ \mathcal{I}_i = \{ x_u \in V^i | \exists y_v \in V^j \text{s.t.} \|T(x_u) - y_v\|_2 < \tau_p \} \]

and

\[ \mathcal{J}_i = \{ y_v \in V^j | \exists x_u \in V^i \text{s.t.} \|T(x_u) - y_v\|_2 < \tau_p \} \]

Finally, the correspondence refinement loss of \(C^i\) reads as:

\[ \mathcal{L}_i^c = - \sum_{(x_u, y_v) \in \mathcal{M}_i} \log \hat{Z}^{u,v}_{i} + \sum_{x_u \notin \mathcal{I}_i} \log \hat{Z}^{u,v}_{i} + 1 \]

\[ - \sum_{y_v \notin \mathcal{J}_i} \log \hat{Z}^{u,v}_{i} + 1 \]  

where \(\hat{Z}^{u,v}_{i}\) denotes the entry of \(\hat{Z}\) on the \(u\)th row and \(v\)th column. The total loss is averaged across the whole node correspondence set \(C\) as \(\mathcal{L}_f = \frac{1}{|C|} \sum_{i=0}^{n} \mathcal{L}_i^c\).

IV. RESULTS

We evaluate RIGA on both synthetic object dataset ModelNet40 [46] and real scene benchmarks, including 3DMatch [47] and 3DLoMatch [2]. RANSAC [1] is leveraged to estimate transformation based on putative correspondences. Qualitative results can be found in Fig. 7. We also illustrate failed cases from 3DLoMatch in Fig. 8. For discussing the shortcomings of RIGA, additional experiments are conducted on KITTI [48], which consists of real-scanned outdoor scene, to demonstrate how the quality of estimated normals affects the performance. Moreover, we compare RIGA to the state-of-the-art methods
Fig. 6. Detailed Architecture of Components. In attention modules, “Multi-head” stands for the multi-head mechanism [24], where \( q, k, \) and \( v \in \mathbb{R}^{in} \) are first reshaped to \((\text{head}, \text{in}/\text{head})\), and attention is then computed separately for each head channel from corresponding \( q \) and \( k \). Value \( v \) in each head channel is fused independently according to the attention computed for the same head. The fused values with shape \((\text{head}, \text{in}/\text{head})\) are reshaped back to \((\text{in}, 1)\), which is finally projected to message \( m \in \mathbb{R}^{in} \).

Table I: Results on ModelNet40 under the “Unseen” and “Noise” Settings

| Methods    | \#dim | Unseen & Noise |
|------------|-------|----------------|
| RPRNet [51]| 1024  | \[0.45\]^* | \[0.18\]^* |
| IDAM [53]  | 32    | -              | -            |
| RGM [54]   | 1024  | 6.47^*         | 0.061        | 0.081 |
| DCP [55]   | 1024  | 10.58^*        | 0.072        | 0.109 |
| DeepGMR [56]| 128  | 17.67^*        | 0.067        | 0.131 |
| RPMNet [29]| 96    | 3.80^*         | 0.022        | 0.032 |
| GCMNet [30]| 128  | 1.32^*         | 0.008        | 0.100 |
| RIGA       | 32    | 2.30^*         | <0.0001      | 0.41^* |

Best performance is highlighted in bold while the second-best is marked with an underline. In “Unseen”, 20 categories are used for training and the rest 20 for testing. In “Noise”, all the categories are split into training and testing. Gaussian noise sampled from \( N(0, 0.01) \) and clipped to \([-0.05, 0.05]\) is added to individual points in both training and testing. In \([0, 45]^\ast\), rotations along each axis are randomly sampled from \([0, 45]^\ast\) and translations are sampled from \([-0.5, 0.5]\). Rotations are enlarged to \(180^\circ\) in \([0, 180]^\ast\).

Table II: Results on ModelNet40 under the “Unseen” Setting

| Methods    | \#dim | \[0.45\]^* | \[0.18\]^* |
|------------|-------|------------|------------|
| PRNet [51]| 1024  | 6.47^*     | 0.061      |
| IDAM [53]  | 32    | 9.50^*     | 0.051      |
| RGM [54]   | 1024  | 2.62^*     | 0.025      |
| DCP [55]   | 1024  | 10.58^*    | 0.072      |
| DeepGMR [56]| 128  | 17.67^*    | 0.067      |
| RPMNet [29]| 96    | 3.80^*     | 0.022      |
| GCMNet [30]| 128  | 1.32^*     | 0.008      |
| RIGA       | 32    | 2.30^*     | <0.0001    |

Table III: Modified Chamfer Distance (MCD) on ModelNet40

| Methods    | \[0.45\]^* | \[0.18\]^* |
|------------|------------|------------|
| GCMNet [30]| 0.0025     | 0.0045     |
| CoFiNet [14]| 0.0007   | 0.0007     |
| RIGA       | 0.0017     | 0.0019     |

Best performance is highlighted in bold.

A. Implementation Details

1) Detailed Architecture: The detailed architecture of each component leveraged in RIGA can be found in Fig. 6. PointNets [15] \( \Psi_g \) and \( \Psi_s \) are two individual models with the same architecture (input dimension \( \text{in} = 4 \), project dimension \( \text{proj} = 64 \) and output dimension \( \text{out} = 256 \)), as shown in the leftmost column in Fig. 6. Each attention block has an intra-frame module and an inter-frame module, both with the architecture of the “Attention Module” shown in Fig. 6. Differently, for intra-frame modules, \( d_q, d_k, \) and \( d_v \) are all from the same frame, while in inter-frame modules, \( d_k \) and \( d_v \) are from the opposite frame. \( \text{MLP}_c \) and \( \text{MLP}_f \) in Fig. 5 have the same MLP architecture shown in the rightmost column of Fig. 6, with an input dimension list of \([256, 128, 64, 32]\).
and for 20 epochs on KITTI, all with $\lambda = 1$ to balance different loss functions. We leverage an Adam optimizer [50] with an initial learning rate of 1e-4, which is exponentially decayed by 0.05 after each epoch. On ModelNet40, we sparsely sample $N'_i = M'_i = 256$ nodes from each point cloud pair, with a radius $r = 0.2$ m to construct support areas, within which the number of points is truncated to 64. On 3DMatch/3DLoMatch, $N'_i$ and $M'_i$ are both set to 512, with $r = 0.3$ m and 512 points within each support area. On KITTI, $N'_i$ and $M'_i$ are set to 512, with $r = 5.0$ m and 128 points within each support area. Besides, the number of points in vicinity $V$ is truncated to 32, 128, and 128 on ModelNet40, 3DMatch/3DLoMatch, and KITTI, respectively. On all datasets, the dimension of intermediate descriptors $g$, $s$, and $d$ is set to 256, while that of descriptors $\hat{d}$, from which correspondences are hierarchically extracted, is set to 32. The number of neighbor points used for feature interpolation is set to $k = 3$. We use 100 iterations for Sinkhorn [42] algorithm. The number of attention blocks is set to $K = 6$, and the attention mechanism is implemented with 4 heads. During training, 256 node pairs that overlap under ground truth transformation are sampled as the node correspondence set $C$. During testing, 256 node correspondences with the highest similarity scores are selected for the consecutive refinement.

### B. Synthetic Object Dataset: ModelNet40

1) **Dataset:** ModelNet40 [46] consists of 12,311 CAD models of objects from 40 different categories. We follow the setting of [51], where 9,833 shapes are used for training, and the rest 2,468 for testing. For each model, 1,024 points are randomly sampled from its surface. For simulating the partial overlap from scanning, 768 points nearest to a randomly selected viewpoint in the space are resampled from the 1,024 points, which serves as the input point cloud. Following [30], instead of using the ground truth normals, we estimate them using Open3D [52]. To demonstrate the significance of being rotation-invariant, we follow [30] to enlarge the rotations of objects to a maximum of 180$^\circ$. As the rotations are generated by adopting Rodrigues’ rotation formula on a random rotation axis together with a random angle, the rotation angles within $[0, 180^\circ]$ cover the full range of 360$^\circ$.

2) **Metrics:** We use 4 widely-adopted metrics [29], [30]: (1) Relative Rotation Error (RRE) that evaluates the error between estimated and ground truth rotation matrices; (2) Relative Translation Error (RTE) that measures the error between estimated and ground truth translation vectors; (3) Root-Mean-Square Error (RMSE) which calculates the residual error between correspondences from the same point cloud, separately transformed by the estimated and ground truth transformation; (4) Modified Chamfer Distance (MCD) that measures the Chamfer distance between a partial frame and the clean whole shape of the other frame. Please refer to the Appendix, available online, for the detailed definition.

### TABLE IV

| 3DMatch | Origin | Rotated | 3DLoMatch | Origin | Rotated |
|---------|--------|---------|-----------|--------|---------|
| 3DSN [8] | 36.0 | 11.4 | -$\infty$ | $|$ | |
| FCGF [9] | 56.8 | 19.4 | 17.3 | $|$ | $|$ |
| DeformNet [10] | 39.0 | 17.2 | 12.1 | $|$ | $|$ |
| RC-GCN [32] | 31.2 | 12.0 | 7.5 | $|$ | $|$ |
| SpinNet [3] | 48.5 | 25.7 | 18.2 | $|$ | $|$ |
| Predictor [2] | 58.0 | 24.7 | 22.4 | $|$ | $|$ |
| YOHO [4] | 64.4 | 25.9 | 23.2 | $|$ | $|$ |
| CoNet [14] | 49.9 | 24.4 | 21.5 | $|$ | $|$ |
| Lepard [13] | 58.8 | 28.4 | 28.4 | $|$ | $|$ |
| RIGA | 57.3 | 27.7 | 17.4 | $|$ | $|$ |
| $|$ | 65.9 | 32.1 | 32.1 | $|$ | $|$ |
| **Feature Matching Recall(%) $\dagger$** | & | & | & | & |
| 3DSN [8] | 95.0 | 63.6 | & | & |
| FCGF [9] | 97.4 | 76.6 | 73.6 | $|$ | $|$ |
| DeformNet [10] | 95.6 | 76.7 | 63.3 | $|$ | $|$ |
| RC-GCN [32] | 90.8 | 79.1 | 70.9 | $|$ | $|$ |
| SpinNet [3] | 97.4 | 80.3 | 75.4 | $|$ | $|$ |
| Predictor [2] | 96.6 | 76.8 | 73.7 | $|$ | $|$ |
| YOHO [4] | 98.2 | 79.8 | 78.6 | $|$ | $|$ |
| CoNet [14] | 95.1 | 80.1 | 78.7 | $|$ | $|$ |
| Lepard [13] | 98.0 | 81.3 | 79.0 | $|$ | $|$ |
| RIGA | 97.9 | 79.6 | 74.3 | 6.6 | $|$ |
| **Registration Recall(%) $\dagger$** | & | & | & | & |
| 3DSN [8] | 79.4 | 33.0 | & | & |
| FCGF [9] | 85.1 | 40.3 | 53.4 | $|$ | $|$ |
| DeformNet [10] | 81.6 | 37.2 | 53.5 | $|$ | $|$ |
| RC-GCN [32] | 74.9 | 41.0 | 41.9 | $|$ | $|$ |
| SpinNet [3] | 88.6 | 58.2 | 41.4 | $|$ | $|$ |
| Predictor [2] | 89.0 | 59.8 | 58.4 | $|$ | $|$ |
| YOHO [4] | 90.8 | 62.5 | 64.6 | $|$ | $|$ |
| CoNet [14] | 95.9 | 67.5 | 62.5 | $|$ | $|$ |
| Lepard [13] | 92.7 | 64.9 | 64.0 | $|$ | $|$ |
| RIGA | 87.2 | 56.8 | 56.8 | 0 | $|$ |
| **Best performance is highlighted in bold while the second best is marked with an underline. In column “Rotated”, every point cloud pair is evaluated with # Samples=5,000’ in Tab. 5 and Tab. 6, and each point cloud is rotated individually with random rotations up to 360$^\circ$ along each axis. Our method significantly outperforms state-of-the-art methods on the rotated benchmarks.** | & | & | & | & |
Comparisons to the State-of-the-Art: We compare RIGA with 9 state-of-the-art baselines, including 7 direct registration methods and 2 correspondence-based approaches (Predator [2] and CoFiNet [14]). The detailed results are shown in Tables I and II. From the second column that lists the dimension of descriptors used for correspondence search, it can be noticed that RIGA uses the most compact descriptors among all the methods. In Table I, under the “Unseen” setting, RIGA surpasses all the other methods with rotations in the range of [0, 45°]. With a maximum rotation of 180°, it achieves on-par performance with GMCNet [30] and outperforms others. Under the “Noise” setting, when Gaussian noise is added, although RIGA stays comparable with GMCNet [30] with rotations in [0, 45°], it outperforms all the baselines on all the metrics by a large margin with rotations enlarged to 180°. Table II demonstrates the results under a more challenging setting that combines the “Unseen” and “Noise” settings in Table I. In this case, RIGA still achieves the state-of-the-art performance and significantly surpasses all the others under rotations up to 180°. Notably, in both tables, all the methods except for RIGA degenerate significantly, which shows the superiority of the inherent rotational invariance of RIGA.

Although direct registration methods are specifically tuned with good performance on object-level data as pointed out in [2],
RIGA could compete with them and even performs significantly better than them on data with Gaussian noise and large rotations. More importantly, RIGA also achieves the state-of-the-art performance on scene-level benchmarks [2], [47], while most direct registration methods fail to work there according to [2]. Moreover, since the symmetry exists in some categories of the ModelNet40 data, we follow [29] to use the MCD metric to evaluate CoFiNet [14] that is the preliminary version of this paper, GMCNet [30] that benefits from its rotation-robust features (not fully rotation-invariant), and RIGA. Results can be found in Table III, where RIGA outperforms baselines with a significant margin and also has the strongest robustness against rotations among all the methods. This experiment again confirms the superiority of guaranteeing the rotation invariance by model design.

C. Real Scene Benchmarks: 3DMatch and 3DLoMatch

1) Datasets: 3DMatch [47] collects 62 scenes, where 46 scenes are used for training, 8 for validation, and the rest 8 for testing. We use the processed data and split in [2], and evaluate RIGA on both 3DMatch [47] (>30% overlap) and 3DLoMatch [2] (10% ~ 30% overlap) protocols. Additionally, we also follow [3], [7] to test on benchmarks with enlarged rotations to demonstrate the superiority of the inherent rotational invariance of our descriptors.

2) Metrics: We follow [2], [14] and use 3 metrics for evaluation: (1) Inlier Ratio (IR), which is the fraction of putative correspondences whose residual error is lower than a threshold $\tau_2 = 0.1 \text{ m}$ under the ground truth transformation, and (2) Feature Matching Recall (FMR) that counts the fraction of point cloud pairs whose Inlier Ratio is larger than a threshold $\tau_1 = 5\%$, and (3) Registration recall (RR) that stands for the fraction of point cloud pairs whose RMSE between the estimated and ground truth transformation is smaller than a threshold $\tau_3 = 0.2 \text{ m}$. Please refer to the Appendix, available online, for details.

3) Comparisons to the State-of-the-Art: In Table V, we compare RIGA with 9 baseline methods. Specifically, 3DSN [8], SpinNet [3], and YOHO [4] are rotation-invariant approaches without global awareness. Predator [2], CoFiNet [14], and Lepard [13] are globally-aware algorithms that are variant to rotations. Specially, we also include the comparisons to RI-GCN [32] which is a rotation-invariant method proposed for point cloud classification with receptive fields enlarged by graph convolutional networks (GCN). For a fair comparison, we use the coarse-to-fine matching strategy same to RIGA to extract correspondences from RI-GCN descriptors. We validate our method on both original and rotated benchmarks. For IR, RIGA significantly outperforms all the baselines on original 3DMatch and 3DLoMatch, which indicates RIGA learns more distinctive descriptors and extracts more reliable correspondences. When the benchmarks are further rotated, our superiority over others becomes more significant, which demonstrates the advantage of our rotational invariance by design. Notably, with larger rotations, only the performance of SpinNet [3], YOHO [4], and RIGA remains stable, which further proves the superiority of inherent rotational invariance over the learned one. For FMR, we perform the best on rotated data. When rotations are enlarged, especially on 3DLoMatch, the performance of all the methods except for RIGA, RI-GCN [32], and SpinNet [3] drops sharply. The performance drop of YOHO further demonstrates the aforementioned drawback of achieving rotational invariance via equivariance. Moreover, due to the lack of global awareness, SpinNet [3] falls behind Predator [2], CoFiNet [14], Lepard [13], and RIGA in terms of FMR, which supports the significance of being globally-aware. Finally, for RR, we perform on-par with CoFiNet [14] and Lepard [13] on original datasets, but again show our excellence when rotations are enlarged. Specially, the behavior of RegTr [38] should be further noticed. Different to all the other baselines that extract correspondences by matching descriptors, RegTr proposes to directly regress the corresponding coordinates. As it outputs the corresponding xyz coordinates that are sensitive to both rotations and translations, when rotations are enlarged on the testing set, the performance of RegTr drops sharply on all the metrics (RR even achieves 0), which indicates its high sensitivity to large rotations.

4) Detailed Results With Different Numbers of Samples: In Tables V, VI and Fig. 9, we follow [2], [14] to show the performance with different numbers of sampled points/correspondences. Lepard [13] and RegTr [38] are excluded in this experiment, as the number of correspondences is fixed by their default settings. The IR of RI-GCN [32],
CoFiNet [14], and RIGA increases when the number of samples decreases. This is because methods with the coarse-to-fine matching mechanism implicitly consider all the potential correspondences and sample the most confident ones for registration, while methods relying on uniform sub-sampling or keypoint detection only extract correspondences from sparsely-sampled nodes, whose repeatability is hard to guarantee especially with fewer samples. When the sample number is decreased from 5,000 to 250, all the other metrics of CoFiNet and RIGA remain stable, while those of the others usually drop significantly, which further proves the excellence of the coarse-to-fine mechanism against fewer samples.

5) Scene-Wise Results on 3DMatch and 3DLoMatch: Following [2], [14], we further detail the performance of RIGA with scene-wise results and 2 metrics (RRE and RT) that have been used for the evaluation on ModelNet40) in Table VII. It can be observed that for RRE and RT, RIGA performs the second best among all the methods (slightly worse than RegTr [38]). Nevertheless, it should be noticed that RegTr’s good performance on the original data comes at the cost of losing the robustness against rotations indicated by the detrimental performance on the rotated data demonstrated in Table IV. Hence, the detailed scene-wise results confirm the superiority of RIGA for scene-level registration.

D. Ablation Study

We ablate different parts of RIGA, including (1) Local Description, (2) Global Description and (3) Attention Blocks to assess the importance of each individual component. We use 3DLoMatch, together with its rotated versions for ablation study.

Table VI: Quantitative Results on Rotated 3DMatch and 3DLoMatch with Different Numbers of Samples

| # Samples | 3DMatch | 3DLoMatch |
|-----------|---------|-----------|
| 500       | 250     | 500       |
|           | InterRatio(%) | InterRatio(%) |
| FC2F [9]  | 94.3    | 47.1      | 42.5          |
| D3Feat [10]| 37.7  | 37.0       | 36.0          |
| RI-GCN [32]| 50.7  | 41.3       | 43.2          |
| SpinNet [3] | 18.4  | 18.7       | 18.4          |
| Predator [2] | 52.8  | 35.4       | 25.0          |
| YOFO [4]  | 61.1    | 60.4       | 51.3          |
| CoFiNet [14]| 46.6  | 48.2       | 49.0          |
| RIGA      | 46.5    | 49.8       | 70.7          |

Best performance is highlighted in bold while the second best is marked with an underline. Each point cloud is rotated individually with random rotations up to 360° along each axis.

Detailed results are found in Table VIII for 3DLoMatch and Rotated 3DLoMatch. Moreover, as an extension of CoFiNet [14], we also ablate on the matching strategies to demonstrate the generalizability of RIGA descriptors when combined with other matching strategies. The detailed comparisons can be found in Table IX. To compare with the concurrent pipeline of GeoTransformer [39] which uses a rotation-invariant global Transformer, we replace our global aggregation part with their global Transformer to evaluate the significance of our remaining pipeline design choices. We show related results in Table X.

1) Local Description: In the ablation of (1) Local Description, we replace our local PPF-based geometric description with two rotation-variant variants: (a) xyz - learning local descriptors from the raw 3D coordinates of all the points in the support area around each node; and (b) relative xyz - learning descriptors from relative 3D coordinates of points w.r.t. the central node of the support area. In both cases, the performance drops compared to the baseline RIGA, which indicates the power of our PPF signature-based geometric description. Moreover, we observe a more significant drop in performance in terms of IR and FMR when facing larger rotations which further demonstrates the importance of rotational invariance. Similarly to [6], [29], we also concatenate PPF signatures with coordinates of points for local description in (c) and (d). This results in a better performance than the variants with only 3D coordinates but still perform slightly worse than the baseline RIGA. Thanks to the global awareness in RIGA, it is unnecessary to supplement PPF with global coordinates, as in (c), to incorporate global contexts. Pure local geometry which is rotation-invariant already promises good performance.

2) Global Description: We first ablate (2) Global Description by removing structural descriptors learned from our proposed global PPF signatures. As shown in (a), this significantly damages the performance especially in terms of IR, which proves the importance of informing local descriptors with global structural cues. To further prove the significance of our rotation-invariant structural description, we replace the structural descriptors in baseline RIGA with (b) xyz - learning global positional descriptors from the raw 3D coordinates of each node, and (c) relative xyz - learning global positional descriptors from the relative position of each node w.r.t. the other nodes in the same frame. Moreover, we also follow [24] to learn descriptors from node coordinates projected by sinusoidal functions [5] in (d). The decreased performance of all the variants further confirms the superiority of our design of encoding structural descriptors from global PPF signatures.

3) Attention Blocks: To emphasize the importance of global awareness, we ablate RIGA in terms of the (3) Attention Blocks and compare all the variants to the original design that has 6 attention blocks (K = 6) consisting of both the self- and cross-attention mechanisms. As shown in (a), we first justify the necessity of using the cross-attention mechanism by replacing it with the self-attention. Results indicate that the model benefits a lot from being aware of the cross-frame context. Then we ablate on the number of leveraged attention blocks for a better understanding of the role that the attention mechanism plays in RIGA. In (b), we remove all the attention blocks (K = 0)
and only use the globally-informed descriptors, which leads to a sharp decrease of the performance. This proves the significance of global awareness obtained from learned global contexts. When we increase the number of attention blocks to (c) $K = 1$ and (d) $K = 3$, the performance increases correspondingly, though it does not reach the baseline performance with $K = 6$. This observation indicates that stronger global awareness improves the overall performance. However, when we keep including more and more Attention Blocks in (e) $K = 10$, the performance only stays on-par with RIGA baseline, indicating that using 6 Attention Blocks is a proper option with good performance.

4) Matching Strategy: As an extension work of CoFiNet [14] whose core contribution is the coarse-to-fine matching strategy, we conduct ablation studies on the way to generate correspondences from descriptors to demonstrate the superiority of the coarse-to-fine matching as well as the generalizability of RIGA descriptors when combined with other matching strategies. Results are demonstrated in Table IX, where we combine RIGA descriptors with two other matching strategies (rand for matching randomly-sampled points and detect for matching keypoints detected by the strategy used in Predator [2]). We follow the evaluation criterion used in [2] where the mutual correspondences are used to compute IR and FMR, while the non-mutual ones are leveraged for RR computation. RIGA descriptors consistently outperform Predator descriptors when combined with the same matching strategy, which illustrates the significance as well as the generalizability of our descriptors.

| Ablation Part | Models | 3DMatch (IR)↑ | 3DMatch (FMR)↑ | 3DMatch (RR)↑ | 3DMatch (Rotated) (IR)↑ | 3DMatch (Rotated) (FMR)↑ | 3DMatch (Rotated) (RR)↑ |
|--------------|--------|---------------|---------------|---------------|------------------|------------------|------------------|

*PPF (baseline)

(a) xyz

20.8(–11.3)

77.5−7.6

56.0−9.1

20.2(–11.9)

76.2−8.3

57.4(−9.5)

(b) relative xyz

25.7(–6.4)

79.4−5.5

58.5−6.6

24.9(–7.2)

79.9−4.6

59.9(−7.0)

(c) xyz + PPF

31.1(1.0)

85.1(0.0)

65.4(0.2)

31.1(1.0)

83.6(0.9)

66.5(0.4)

(d) relative xyz + PPF

30.7(1.4)

84.3(1.2)

63.2(2.0)

30.7(1.4)

83.1(1.4)

64.2(0.7)

*PPF (baseline)

(a) none

13.8(–18.3)

75.4−9.7

61.1(−4.0)

13.9(–18.2)

76.0(−8.5)

66.0(−9.9)

(b) relative xyz

18.4(–13.5)

81.3−3.8

65.1(4.0)

18.5(–13.4)

80.5(−4.7)

65.8(−1.1)

(c) xyz + sinusoidal [24]

15.1(–17.0)

77.5−7.6

62.8(2.3)

14.8(–17.3)

75.7(−8.8)

65.2(−1.7)

(3) Attention Blocks

(a) K=0 w/o/cross

16.3(−15.8)

77.8−8.0

61.3(−3.3)

16.2(−15.9)

77.8−8.1

65.1(−1.7)

(b) K=0

10.2(−21.9)

60.8−24.3

50.0(−15.1)

10.3(−21.8)

60.2(−24.3)

53.6(−13.3)

(c) K=1 + cross

16.6(−15.5)

77.1−8.0

63.0(2.1)

16.6(−15.5)

77.8(−6.7)

66.1(−0.8)

(d) K=3 − cross

24.9(−7.2)

82.2(2.9)

65.1(4.0)

25.0(−7.1)

82.4(2.1)

66.6(−0.3)

(e) K=10 − cross

32.5(0.4)

83.6(1.5)

63.6(1.5)

32.4(0.3)

83.4(1.1)

66.8(−0.1)

In the brackets are the changes compared to baseline RIGA that starts with *.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Also, the superiority of the coarse-to-fine matching strategy in comparison with others is well-demonstrated (see the comparisons in terms of FMR and RR between RIGA-rand and RIGA in Table V). Moreover, the correspondences generated from RIGA descriptors are invariant to rotations, regardless of the matching strategies.

5) Global Aggregation: To evaluate our global aggregation design (global PPF signature + Transformer), as well as to make a fair comparison with GeoTransformer [39], whose core contribution is a rotation-invariant Transformer for global context aggregation, we design an ablation study that uses the global transformer proposed in [39] inside the RIGA pipeline. As shown in Table X, in terms of all the metrics, the original RIGA achieves on-par performance with RIGA with the global Transformer from [39], which confirms the significance of our design of aggregating global context for learning more discriminative descriptors.

6) Loss Functions: In the design of RIGA, we use different loss functions for guiding the training of different levels of matching, i.e., Circle Loss [44] for coarse matching and the negative log-likelihood loss [41] for fine matching following [39]. This is inconsistent with the original design of CoFiNet [14] that uses the negative log-likelihood loss for both levels, since the two levels of matching play different roles. The coarse-level matching should cover a wider range of overlapping patches and provide a better initialization for the consecutive refinement, while the fine-level matching should accurately determine the point correspondences such that they can be used for downstream tasks, e.g., point cloud registration. To support this claim, we further conduct experiments by using same loss functions (Circle Loss or the negative log-likelihood loss on both levels). Related results can be found in Table XI. Compared to the original design of RIGA, using Circle Loss on both levels leads to a significant performance drop, as on the fine-matching level, Circle loss cannot deal with occlusion, which introduces a plenty of outlier correspondences. Staying consistent with CoFiNet achieves on par performance on 3DMatch but leads to a performance drop in the more challenging low overlapping scenarios, which justifies the loss design of RIGA.

E. Robustness Against Poor Normal Estimation

We conclude the major shortcoming of RIGA as the side effects brought by the poor normal estimation in certain scenarios, as our inherent rotational invariance is affected by the quality of the estimated normals. For a better understanding of this part, we further conduct extensive experiments on KITTI [48], which consists of 11 outdoor sequences scanned by LiDAR, to prove the robustness of our RIGA descriptors against poor normal estimation. We follow [2] for data processing and training. The major challenge of KITTI comes from its varying point density which could lead to a poor normal estimation. The estimated normals of both indoor and outdoor scenarios are visualized in Fig. 10 to show the poor normal estimation for outdoor scenes compared to indoor ones. Under this circumstance, as shown in Table XII, although RIGA is affected by the poor normal quality,
it still performs on par with those state-of-the-art methods in terms of three different metrics.

V. CONCLUSION

In this paper, we introduce RIGA with a ViT architecture that learns both rotation-invariant and globally-aware descriptors, upon which correspondences are established in a coarse-to-fine manner for point cloud registration. We learn from rotation-invariant PPFs for encoding local geometry and further introduce global PPF signatures to encode a node-specific structural description of the whole scene. The structural descriptors learned from global PPF signatures strengthen local descriptors with the global 3D structures in a rotation-invariant fashion. The distinctiveness of descriptors is further enhanced in the consecutive attention blocks with the learned geometric context across the whole scene. The coarse-to-fine mechanism is further leveraged to establish reliable correspondences upon our powerful RIGA descriptors. Experimental results confirm the effectiveness of our approach on both object and scene-level data. We hope our work can inspire more research looking toward the joint rotational invariance and distinctiveness of descriptors in point cloud registration.

REFERENCES

[1] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” Commun. ACM, vol. 24, pp. 381–395, 1981.
[2] S. Huang, Z. Gojcic, M. Usvyatsov, A. Wieser, and K. Schindler, “PREDATOR: Registration of 3D point clouds with low overlap,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 4265–4274.
[3] S. Ao, Q. Hu, B. Yang, A. Markham, and Y. Guo, “SpinNet: Learning a general surface descriptor for 3D point cloud registration,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 11748–11757.
[4] H. Wang, Y. Liu, Z. Dong, W. Wang, and B. Yang, “You only hypothesize once: Point cloud registration with rotation-equivariant descriptors,” 2021, arXiv:2109.00182.
[5] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, pp. 2579–2605, 2008.
[6] H. Deng, T. Birdal, and S. Ilic, “PPFNet: Global context aware local features for robust 3D point matching,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 195–205.
[7] H. Deng, T. Birdal, and S. Ilic, “PPF-FoldNet: Unsupervised learning of rotation invariant 3D local descriptors,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 620–638.
[8] Z. Gojcic, C. Zhou, J. D. Wegner, and A. Wieser, “The perfect match: 3D point cloud matching with smoothed densities,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 5540–5549.
[9] C. Choy, J. Park, and V. Koltun, “Fully convolutional geometric features,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 8957–8965.
[10] X. Bai, Z. Luo, L. Zhou, H. Fu, L. Quan, and C.-L. Tai, “D3Feat: Joint learning of dense detection and description of 3D local features,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 5540–5549.
[11] M. Saleh, S. Dehghani, B. Busam, N. Navab, and F. Tombari, “Graphite: Graph-induced feature extraction for point cloud registration,” in Proc. Int. Conf. 3D Vis., 2020, pp. 241–251.
[12] M. Saleh, S.-C. Wu, L. Cosmo, N. Navab, B. Busam, and F. Tombari, “Bending graphs: Hierarchical shape matching using gated optimal transport,” 2022, arXiv:2202.01537.
[13] Y. Li and T. Harada, “Lepard: Learning partial point cloud matching in rigid and deformable scenes,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2022, pp. 5544–5554.
[14] H. Yu, F. Li, M. Saleh, B. Busam, and S. Ilic, “CoFlNet: Reliable coarse-to-fine correspondences for robust poincloud registration,” in Proc. Int. Conf. Neural Inf. Process. Sys., 2021, pp. 23872–23884.
[15] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 77–85.
Zheng Qin received the BE and ME degrees in computer science and technology from the National University of Defense Technology (NUDT), China, in 2016 and 2018, respectively, where he is currently working toward the PhD degree. His research interests focus on 3D vision, including point cloud registration, pose estimation, and 3D representation learning.

Mahdi Saleh received the master’s degree in biomedical computing from the Technical University of Munich, Germany. He is currently working toward the PhD degree with the Chair for Computer Aided Medical Procedures and Augmented Reality (CAMP), TU Munich. He previously did his internship with Meta (Facebook) Reality Labs in Redmond, WA and Sausalito, CA. His research focus is on hierarchical 3D representation and 6D object pose estimation.

Ivan Shugurov received the bachelor’s degree in software engineering from the Higher School of Economics. He is currently working toward the PhD degree with the Chair for Computer Aided Medical Procedures / TU Munich and at Siemens Corporate Technology under supervision of PD. Dr. Slobodan Ilic. Subsequently, he studied informatics with TU Munich. In continuation to his Master’s thesis on “Precise 6 DoF pose estimation with dense correspondences”, which was presented at ICCV 2019, he now focuses his research on multimodal 2D/3D object detection and pose estimation.

Kai Wang received the BEng degree in computer science from Zhejiang University, in 2016, and the PhD degree in computer science from the University of New South Wales, in 2020. He is currently an Assistant Professor with the Antai College of Economics and Management, Shanghai Jiao Tong University. His research interests lie in complex data analytics and data management.

Benjamin Busam received the graduate (distinction) degree from TU Munich, in 2014, and studied mathematics with École Polytechnique, Palaiseau, France and with the University of Melbourne, Australia. He is a research scientist with the Technical University of Munich where he coordinates the Computer Vision activities at CAMP. Formerly head of research with FRAMOS, he led the 3D Computer Vision Team, Huawei Research, London from 2018 to 2020. He received the Young Professional Award 2015 from the European Machine Vision Association (EMVA) and Outstanding Reviewer Awards at 3DV both in 2020 and 2021. He now focuses on visual 3D perception, sensor fusion, and collaborative robotics.

Slobodan Ilic received the PhD degree from EPFL, Switzerland, in 2005 under supervision of Pascal Fua. He is currently senior key expert research scientist with Siemens Corporate Technology in Munich, Perlach. He is also a visiting researcher and lecturer with Computer Science Department, TUM and closely works with the CAMP chair. From 2009 until end of 2013, he was leading the Computer Vision Group of CAMP, TUM, and before that he was a senior researcher with Deutsche Telekom Laboratories in Berlin. His research interests include: 3D reconstruction, deformable surface modelling and tracking, real-time object detection and tracking, human pose estimation, and semantic segmentation.