Rethinking Rotation Invariance with Point Cloud Registration

Jianhui Yu, Chaoyi Zhang, Weidong Cai

School of Computer Science, University of Sydney, Australia
{jianhui.yu, chaoyi.zhang, tom.cai}@sydney.edu.au

Abstract

Recent investigations on rotation invariance for 3D point clouds have been devoted to devising rotation-invariant feature descriptors or learning canonical spaces where objects are semantically aligned. Examinations of learning frameworks for invariance have seldom been looked into. In this work, we review rotation invariance in terms of point cloud registration and propose an effective framework for rotation invariance learning via three sequential stages, namely rotation-invariant shape encoding, aligned feature integration, and deep feature registration. We first encode shape descriptors constructed with respect to reference frames defined over different scales, e.g., local patches and global topology, to generate rotation-invariant latent shape codes. Within the integration stage, we propose Aligned Integration Transformer to produce a discriminative feature representation by integrating point-wise self- and cross-relations established within the shape codes. Meanwhile, we adopt rigid transformations between reference frames to align the shape codes for feature consistency across different scales. Finally, the deep integrated feature is registered to both rotation-invariant shape codes to maximize feature similarities, such that rotation invariance of the integrated feature is preserved and shared semantic information is implicitly extracted from shape codes. Experimental results on 3D shape classification, part segmentation, and retrieval tasks prove the feasibility of our work. Our project page is released at: https://rotation3d.github.io/.

Introduction

Point cloud analysis has recently drawn much interest from researchers. As a common form of 3D representation, the growing presence of point cloud data is encouraging the development of many deep learning methods (Qi et al. 2017a; Guo et al. 2021; Zhang et al. 2021), showing great success for well-aligned point clouds on different tasks. However, it is difficult to directly apply 3D models to real data as raw 3D objects are normally captured at different viewing angles, resulting in unaligned data samples, which inevitably impact the deep learning models which are sensitive to rotations. Therefore, rotation invariance becomes an important research topic in the 3D domain.

To achieve rotation invariance, a straightforward way is to augment training data with massive rotations which, however, requires a large memory capacity and exhibits limited generalization ability to unseen data (Kim, Park, and Han 2020). There are attempts to align 3D inputs to a conical pose (Jaderberg et al. 2015; Cohen et al. 2018), or to learn rotation robust features via equivariance (Deng et al. 2021; Luo et al. 2022), while these methods are not rigorously rotation-invariant and present noncompetitive performance on 3D shape analysis. To maintain consistent model behavior under random rotations, some methods (Zhang et al. 2019; Chen et al. 2019; Xu et al. 2021) follow Drost et al. (2010) to handcraft rotation-invariant point-pair features. Others (Zhang et al. 2020; Li et al. 2021a; Zhao et al. 2022) design robust features from equivariant orthonormal bases. Most of the mentioned works either manipulate model inputs or generate canonical spaces to achieve rotation invariance (RI). In this work, we review the problem of RI from a different aspect: robust point cloud registration (PCR). We find that PCR and RI share the *same* goal: PCR aligns low-dimensional point cloud features (*e.g.*, xyz) from the source domain to the target domain regardless of transformations,

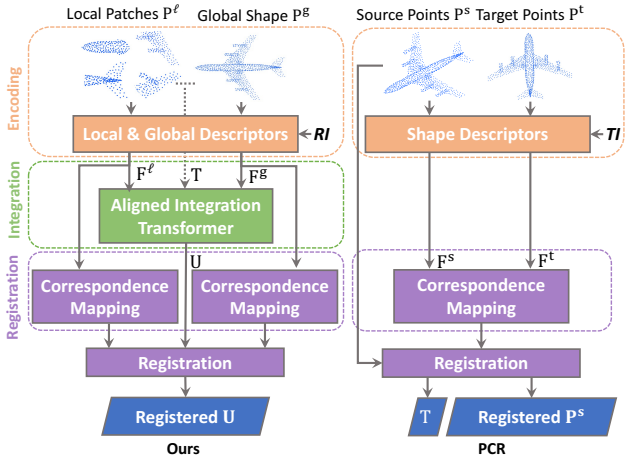

Figure 1: Frameworks of our design (left) and robust point cloud registration (right), where TI and RI are transformation invariance and rotation invariance, and $\mathbf{T}$ is the rigid transformation. The dotted line indicates the computation of $\mathbf{T}$ between reference frames.

while RI can be considered to align high-dimensional latent features to rotation-invariant features. Specifically, the goal of PCR is to explicitly align the source point cloud to the target, both representing the same 3D object, and for RI learning, we implicitly align the final feature representation of a 3D shape to a hidden feature of the same shape, which is universally rotation-invariant to any rotations.

Motivated by this finding, we propose our learning framework in Fig. 1 with three sequential stages, namely rotation-invariant shape encoding, aligned feature integration, and deep feature registration. Firstly, we **(a)** construct and feed point pairs with different scales as model inputs, where we consider local patches $\mathbf{P}^{\ell}$ with small number of points and global shape $\mathbf{P}^{g}$ with the whole 3D points. Hence, the final feature representation can be enriched by information from different scales. Low-level rotation-invariant descriptors are thus built on reference frames and encoded to generate latent shape codes $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$ following recent PCR work (Pan, Cai, and Liu 2022). Secondly, we **(b)** introduce a variant of transformer (Vaswani et al. 2017), Aligned Integration Transformer (AIT), to implicitly integrate information from both self- and cross-attention branches for effective feature integration. In this way, information encoded from different point scales is aggregated to represent the same 3D object. Moreover, we consider $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$ as *unaligned* since they are encoded from *unaligned* reference frames. To address the problem, we follow the evaluation technique proposed in PCR (Pan, Cai, and Liu 2022), where we use relative rotation information ($\mathbf{T}$) with learnable layers to align $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$ for feature consistency. Finally, to ensure RI of the integrated feature $\mathbf{U}$, we follow PCR to **(c)** examine the correspondence map of ($\mathbf{F}^{g}$, $\mathbf{U}$) and ($\mathbf{F}^{\ell}$, $\mathbf{U}$), such that the mutual information between a local patch of a 3D object and the whole 3D object is maximized, and RI is further ensured in the final geometric feature.

The contributions of our work are summarized as following three folds: (1) To our knowledge, we are the first in developing a PCR-cored representation learning framework towards effective RI studies on 3D point clouds. (2) We introduce Aligned Integration Transformer (AIT), a transformer-based architecture to conduct aligned feature integration for a comprehensive geometry study from both local and global scales. (3) We propose a registration loss to maintain rotation invariance and discover semantic knowledge shared in different parts of the input object. Moreover, the feasibility of our proposed framework is successfully demonstrated on various 3D tasks.

Related Work

Rotation Robust Feature Learning. Networks that are robust to rotations can be equivariant to rotations. Esteves et al. (2018) and Cohen et al. (2018) project 3D data into a spherical space for rotation equivariance and perform convolutions in terms of spherical harmonic bases. Some (Spezialetti et al. 2020; Sun et al. 2021) learn canonical spaces to unify the pose of point clouds. Recent works (Luo et al. 2022; Deng et al. 2021; Jing et al. 2020) vectorize the scalar activations and mapping SO(3) actions to a latent space for easy manipulations. Although these works present competitive results, they cannot be strictly rotation-invariant. Another way for rotation robustness is to learn rotation-invariant features. Handcrafted features can be rotation-invariant (Zhang et al. 2019; Chen et al. 2019; Chen and Cong 2022; Xu et al. 2021), but they normally ignore the global overview of 3D objects. Others use rotation-equivariant local reference frames (LRFs) (Zhang et al. 2020; Thomas 2020; Kim, Park, and Han 2020) or global reference frames (GRFs) (Li et al. 2021a) as model inputs based on principal component analysis (PCA). However, they may produce inconsistent features across different reference frames, which would limit the representational power. In contrast to abovementioned methods with rotation robust model inputs or modules, we examine the relation between RI and PCR and propose an effective framework.

3D Robust Point Cloud Registration. Given a pair of LiDAR scans, 3D PCR requires an optimal rigid transformation to best align the two scans. Despite the recent emerging of ICP-based methods (Besl and McKay 1992; Wang and Solomon 2019b), we follow robust correspondence-based approaches in our work (Deng, Birdal, and Ilic 2018; Yuan et al. 2020; Qin et al. 2022; Pan, Cai, and Liu 2022), where RI is widely used to mitigate the impact of geometric transformations during feature learning. Specifically, both Pan, Cai, and Liu (2022) and Qin et al. (2022) analyze the encoding of transformation-robust information and introduce a rotation-invariant module with contextual information into their registration pipeline. All these methods showing impressive results are closely related to rotation invariance. We hypothesize that the learning framework of RI can be similar to PCR, and we further prove in experiments that our network is feasible and able to achieve competitive performance on rotated point clouds.

Transformers in 3D Point Clouds. Transformers (Dosovitskiy et al. 2021; Liu et al. 2021) applied to 2D vision have shown great success, and they are gaining prominence in 3D point clouds. For example, Zhao et al. (2021) uses vectorized self-attention (Vaswani et al. 2017) and positional embedding for 3D modeling. Guo et al. (2021) proposes offset attention for noise-robust geometric representation learning. Cross-attention is widely employed for semantic information exchange (Qin et al. 2022; Yu et al. 2021a), where feature relations between the source and target domains are explored. Taking advantage of both, we design a simple yet effective feature integration module with self and cross relations. In addition, transformation-related embeddings are introduced for consistent feature learning.

Contrastive Learning with 3D Visual Correspondence. Based on visual correspondence, contrastive learning aims to train an embedding space where positive samples are pushed together whereas negative samples are separated away (He et al. 2020). The definition of positivity and negativity follows the visual correspondence maps, where pairs with high confidence scores are positive otherwise negative. Visual correspondence is important in 3D tasks, where semantic information extracted from matched point pairs improves the network's understanding on 3D geometric struc-

tures. For example, PointContrast (Xie et al. 2020) explores feature correspondence across multiple views of one 3D point cloud with InfoNCE loss (Van den Oord, Li, and Vinyals 2018), increasing the model performance for downstream tasks. Info3D (Sanghi 2020) and CrossPoint (Afham et al. 2022) minimize the semantic difference of point features under different poses. We follow the same idea by registering the deep features to rotation-invariant features at intermediate levels, increasing feature similarities in the embedding space to ensure rotation invariance.

Method

Given a 3D point cloud including N_{in} points with xyz coordinates $\mathbf{P} = \{p_i \in \mathbb{R}^3\}_{i=1}^{N_{in}}$, we aim to learn a shape encoder f that is invariant to 3D rotations: $f(\mathbf{P}) = f(\mathbf{RP})$, where $\mathbf{R} \in SO(3)$ and SO(3) is the rotation group. RI can be investigated and achieved through three stages, namely rotation-invariant shape encoding (Section), aligned feature integration (Section), and deep feature registration (Section).

Rotation-Invariant Shape Encoding

In this section, we first construct the input point pairs from local and global scales based on reference frames, following the idea of Pan, Cai, and Liu (2022) to obtain low-level rotation-invariant shape descriptors from LRFs and GRF directly. Then we obtain latent shape codes via two set abstraction layers as in PointNet++ (Qi et al. 2017b).

Rotation Invariance for Local Patches. To construct rotation-invariant features on LRFs, we hope to construct an orthonormal basis for each LRF as $p \in \mathbb{R}^{3\times3}$. Given a point p_i and its neighbor $p_j \in \mathcal{N}(p_i)$, we choose $\overrightarrow{x_i}^\ell = \overrightarrow{p_m p_i} / \|\overrightarrow{p_m p_i}\|_2$, where p_m is the barycenter of the local geometry and $\|\cdot\|_2$ is L2-norm. We then define $\overrightarrow{z_i}^\ell$ following Tombari, Salti, and Stefano (2010) to have the same direction as an eigenvector, which corresponds to the smallest eigenvalue via eigenvalue decomposition (EVD):

$$\mathbf{\Sigma}_i^\ell = \sum_{j=1}^{|\mathcal{N}(p_i)|} \alpha_j \left(\overrightarrow{p_i p_j}\right)\left(\overrightarrow{p_i p_j}\right)^\top,\ \alpha_j = \frac{d - \|\overrightarrow{p_i p_j}\|_2}{\sum_{j=1}^{|\mathcal{N}(p_i)|} d - \|\overrightarrow{p_i p_j}\|_2}, \tag{1}$$

where α_j is a weight parameter, allowing nearby p_j to have large contribution to the covariance matrix, and d is the maximum distance between p_i and p_j. Finally, we define $\overrightarrow{y_i}^\ell$ as $\overrightarrow{z_i}^\ell \times \overrightarrow{x_i}^\ell$. RI is introduced to p_i with respect to its neighbor p_j as $p_{ij}^\ell = \overrightarrow{p_i p_j}^\top \mathbf{M}_i^\ell$. Proofs of the equivariance of $\mathbf{M}_i^\ell$ and invariance of p_{ij}^ℓ are shown in the supplementary material. The latent shape code $\mathbf{F}^\ell \in \mathbb{R}^{N\times C}$ is obtained via PointNet++ and max-pooling.

Rotation Invariance for Global Shape. We apply PCA as a practical tool to obtain RI in a global scale. Similar to Eq. 1, PCA is performed by $\frac{1}{N_0}\sum_{i=1}^{N_0}(\overrightarrow{p_m p_i})(\overrightarrow{p_m p_i})^\top = \mathbf{U}^g \mathbf{\Lambda}^g \mathbf{U}^{g\top}$, where p_m is the barycenter of $\mathbf{P}$, $\mathbf{U}^g = [\overrightarrow{u_1}^g, \overrightarrow{u_2}^g, \overrightarrow{u_3}^g]$ and $\mathbf{\Lambda}^g = \mathrm{diag}(\lambda_1^g, \lambda_2^g, \lambda_3^g)$ are eigenvector and eigenvalue matrices. We take $\mathbf{U}^g$ as the orthonormal basis $\mathbf{M}^g = [\overrightarrow{x}^g, \overrightarrow{y}^g, \overrightarrow{z}^g]$ for GRF. By transforming point p_i with $\mathbf{U}^g$, the shape pose is canonicalized as $p_i^g = p_i \mathbf{M}^g$. Proof of the RI of p_i^g is omitted for its simplicity, and $\mathbf{F}^g \in \mathbb{R}^{N\times C}$ is obtained following PointNet++.

Sign Ambiguity. EVD introduces sign ambiguity for eigenvectors, which negatively impacts the model performance (Bro, Acar, and Kolda 2008). The description of sign ambiguity states that for a random eigenvector $\vec{u}$, $\vec{u}$ and $\vec{u}'$, with $\vec{u}'$ having an opposite direction to $\vec{u}$, are both acceptable solutions to EVD. To tackle this issue, we simply force $\overrightarrow{z_i}^\ell$ of LRF to follow the direction of $\overrightarrow{op_i}$, with o being the origin of the world coordinate. We disambiguate basis vectors in $\mathbf{M}^g$ by computing an inner product with $\overrightarrow{p_m p_i}, \forall i \in N_0$. Taking $\overrightarrow{x}^g$ for example, its direction is conditioned on the following term:

$$\overrightarrow{x}^g = \begin{cases} \overrightarrow{x}^g, & \text{if } S_x \geq \frac{N_0}{2} \\ \overrightarrow{x}'^g, & \text{otherwise} \end{cases}, \quad S_x = \sum_{i=1}^{N_0} \mathbb{1}[\langle \overrightarrow{x}^g, \overrightarrow{p_m p_i} \rangle], \tag{2}$$

where $\langle\cdot,\cdot\rangle$ is the inner product, $\mathbb{1}[\cdot]$ is a binary indicator that returns 1 if the input argument is positive, otherwise 0. S_x denotes the number of points where $\overrightarrow{x}^g$ and $\overrightarrow{p_m p_i}$ point to the same direction. The same rule is applied to disambiguate $\overrightarrow{y}^g$ and $\overrightarrow{z}^g$ by S_y and S_z. Besides, as mentioned in Li et al. (2021a), $\mathbf{M}^g$ might be non-rotational (e.g., reflection). To ensure $\mathbf{M}^g$ a valid rotation, we simply reverse the direction of the basis vector whose S value is the smallest. More analyses on sign ambiguity are in the supplementary material.

Aligned Feature Integration

Transformer has been widely used in 3D domain to capture long-range dependencies (Yu et al. 2021b). In this section, we introduce Aligned Integration Transformer (AIT), an effective transformer to align latent shape codes with relative rotation angles and integrate information via attention-based integration (Cheng et al. 2021). Within each AIT module, we first apply Intra-frame Aligned Self-attention on $\mathbf{F}^\ell$ and we do not encode $\mathbf{F}^g$, which is treated as supplementary information to assist local geometry learning with the global shape overview. We discuss that encoding $\mathbf{F}^g$ via self-attention can increase model overfitting, thus lowering the model performance. We will validate our discussion in Section . Inter-frame Aligned Cross-attention is applied on both $\mathbf{F}^\ell$ and $\mathbf{F}^g$, and we use Attention-based Feature Integration module for information Aggregation.

Preliminary: Offset Attention. AIT utilizes offset attention (Guo et al. 2021) for noise robustness. In the following, we use subscripts *sa* and *ca* to denote implementations related to self- and cross-attention, respectively. We first review offset attention as follows:

$$\mathbf{F} = \boldsymbol{\phi}(\mathbf{F}_{oa}) + \mathbf{F}_{in},\ \mathbf{F}_{oa} = \mathbf{F}_{in} - \|\,\mathrm{SM}(\mathbf{A})\|_1 \mathbf{v},\ \mathbf{A} = \mathbf{q}\mathbf{k}^\top, \tag{3}$$

where $\mathbf{q} = \mathbf{F}_{in}\mathbf{W}_q$, $\mathbf{k} = \mathbf{F}_{in}\mathbf{W}_k \in \mathbb{R}^{N\times d}$, and $\mathbf{v} = \mathbf{F}_{in}\mathbf{W}_v \in \mathbb{R}^{N\times C}$ are query, key, and value embeddings, and $\mathbf{W}_q$, $\mathbf{W}_k \in \mathbb{R}^{C\times d}$, $\mathbf{W}_v \in \mathbb{R}^{C\times C}$ are the corresponding projection matrices. $\|\cdot\|_1$ is L1-norm and $\boldsymbol{\phi}$ denotes a multi-layer perceptron (MLP) and $\mathrm{SM}(\cdot)$ is softmax operation. $\mathbf{F}_{oa}$ is offset attention-related feature and $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the attention logits.

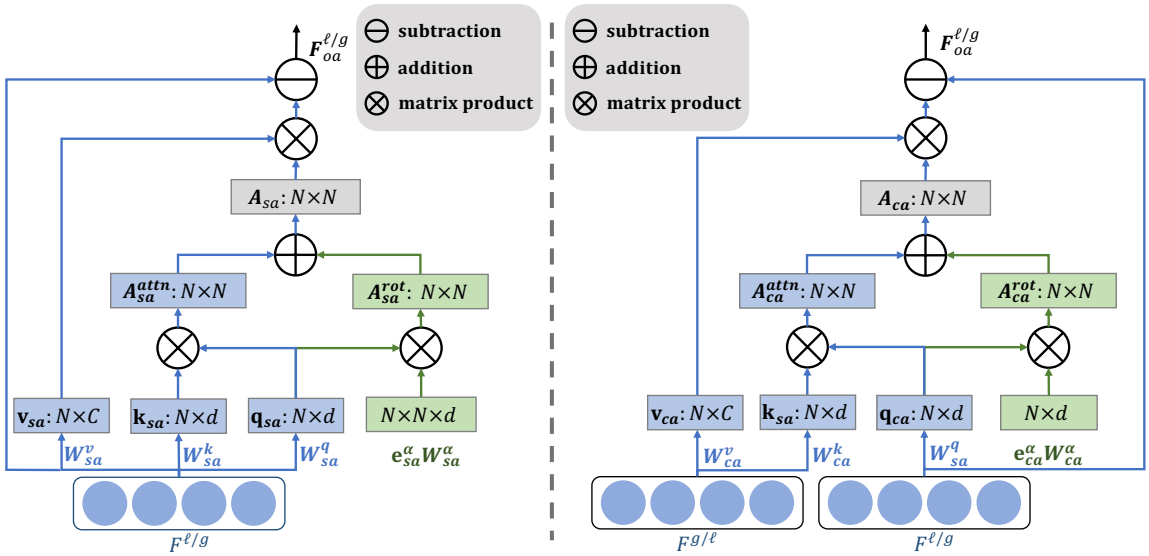

Figure 2: Illustrations of (a) Intra-frame Aligned Self-attention and (b) Inter-frame Aligned Cross-attention modules. Note that we only present processes for computing $\mathbf{F}_{oa}$ in both modules.

Intra-frame Aligned Self-attention. Point-wise features of $\mathbf{F}^{\ell}$ are encoded from *unaligned* LRFs, so direct implementation of self-attention on $\mathbf{F}^{\ell}$ can cause feature inconsistency during integration. To solve this problem, rigid transformations between distinct LRFs are considered, which are explicitly encoded and injected into point-wise relation learning process. We begin by understanding the transformation between two LRFs. For any pair of local orthonormal bases $\mathbf{M}_i^{\ell}$ and $\mathbf{M}_j^{\ell}$, a rotation can be easily derived $\Delta\mathbf{R}_{ji} = \mathbf{M}_i^{\ell}\mathbf{M}_j^{\ell\top}$ and translation is defined as $\Delta\mathbf{t}_{ji} = o_i^{\ell} - o_j^{\ell}$, where $o_{i/j}^{\ell}$ indicates the origin. In our work, the translation part is intentionally ignored, where we show in the supplementary material that by keeping both rotation and translation information, the model performance decreases.

Although $\Delta\mathbf{R}_{ji}$ is invariant to rotations, we do not directly project it into the embedding space, as it is sensitive to the order of matrix product: $\Delta\mathbf{R}_{ji} \neq \Delta\mathbf{R}_{ij}$, giving inconsistent rotation information when the product order is not maintained. To address this issue, we construct our embedding via the relative rotation angle $\Delta\alpha_{ji}$ between $\mathbf{M}_i^{\ell}$ and $\mathbf{M}_j^{\ell}$, which is normally used in most PCR works (Yew and Lee 2020; Pan, Cai, and Liu 2022) for evaluations. The relative rotation angle $\Delta\alpha_{ji}$ is computed as:

$$\Delta\alpha_{ji} = \arccos\left(\frac{\text{Trace}\left(\Delta\mathbf{R}_{ji}\right) - 1}{2}\right)\frac{180}{\pi} \in [0, \pi], \quad (4)$$

where it is easy to see that $\Delta\alpha_{ji} = \Delta\alpha_{ij}$. We further apply sinusoidal functions on $\Delta\alpha_{ji}$ to generate N^2 pairs of angular embeddings $\mathbf{e}^{\alpha} \in \mathbb{R}^{N\times N\times d}$ for all N points as:

$$e_{i,j,2k}^{\alpha} = \sin\left(\frac{\Delta\alpha_{ji}/t_{\alpha}}{10000^{2k/d}}\right),\ e_{i,j,2k+1}^{\alpha} = \cos\left(\frac{\Delta\alpha_{ji}/t_{\alpha}}{10000^{2k/d}}\right), \quad (5)$$

where t_{α} controls the sensitivity to angle variations.

Finally, we inject $\mathbf{e}^{\alpha}$ into offset attention and learn intra-frame aligned feature $\mathbf{F}_{\text{IAS}}^{\ell}$ via self-attention as follows:

$$\begin{gathered}\mathbf{F}_{\text{IAS}}^{\ell} = \phi\left(\mathbf{F}_{oa}^{\ell}\right) + \mathbf{F}^{\ell},\ \mathbf{F}_{oa}^{\ell} = \mathbf{F}^{\ell} - \|\,\text{SM}(\mathbf{A}_{sa})\|_1\mathbf{v}_{sa}, \\ \mathbf{A}_{sa} = \mathbf{A}_{sa}^{attn} + \mathbf{A}_{sa}^{rot}, \\ \mathbf{A}_{sa}^{attn} = \mathbf{q}_{sa}\mathbf{k}_{sa}^{\top},\ \mathbf{A}_{sa}^{rot} = \mathbf{q}_{sa}(\mathbf{e}_{sa}^{\alpha}\mathbf{W}_{sa}^{\alpha})^{\top},\end{gathered} \quad (6)$$

where $\mathbf{q}_{sa}/\mathbf{k}_{sa}/\mathbf{v}_{sa} = \mathbf{F}^{\ell}\mathbf{W}_{sa}^{\mathbf{q}}/\mathbf{F}^{l}\mathbf{W}_{sa}^{\mathbf{k}}/\mathbf{F}^{l}\mathbf{W}_{sa}^{\mathbf{v}}$, $\mathbf{W}_{sa}^{\alpha} \in \mathbb{R}^{d\times d}$ is a linear projection to refine the learning of $\mathbf{e}_{sa}^{\alpha}$, and $\mathbf{A}_{sa}$ is the attention logits. The same process can be performed for $\mathbf{F}^{g}$ by swapping the index ℓ and g. Detailed illustrations are shown in Fig. 2 (a).

Inter-frame Aligned Cross-attention. Semantic information exchange between $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$ in the feature space is implemented efficiently by cross-attention (Chen, Fan, and Panda 2021). Since $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$ are learned from different coordinate systems, inter-frame transformations should be considered for cross-consistency between $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$. An illustration of the cross-attention module is shown in Fig. 2 (b), which indicates that the computation of inter-frame aligned feature $\mathbf{F}_{\text{IAC}}^{\ell}$ via cross-attention follows a similar way as Eq. 6 by replacing all subscripts sa by ca. As illustrated in Fig. 2 (b), $\mathbf{A}_{ca}$ is cross-attention logits containing point-wise cross-relations over point features defined across local and global scales. $\mathbf{e}_{ca}^{\alpha} \in \mathbb{R}^{N\times d}$ is computed via Eq. 4 and Eq. 5 in terms of the transformation between $\mathbf{M}_i^{\ell}$ and $\mathbf{M}^{g}$. To this end, the geometric features learned between local and global reference frames can be aligned given $\mathbf{e}_{ca}^{\alpha}$, leading to a consistent feature representation.

Attention-based Feature Integration. Instead of simply adding the information from both $\mathbf{F}^{\ell}$ and $\mathbf{F}^{g}$, we integrate

information by incrementing attention logits. Specifically, we apply self-attention on $\mathbf{F}^\ell$ with attention logits $\mathbf{A}_{sa}$ and cross-attention between $\mathbf{F}^\ell$ and $\mathbf{F}^g$ with attention logits $\mathbf{A}_{ca}$. We combine $\mathbf{A}_{sa}$ and $\mathbf{A}_{ca}$ via addition, so that encoded information of all point pairs from a local domain can be enriched by the global context of the whole shape. Illustration is shown in the supplementary material. The whole process is formulated as follows:

$$\begin{aligned} \mathbf{U} &= \boldsymbol{\phi}\left(\mathbf{F}_{oa}\right) + \mathbf{F}^\ell, \\ \mathbf{F}_{oa} &= \mathbf{F}^\ell - \|\,\mathrm{SM}(\mathbf{A}_{sa} + \mathbf{A}_{ca})\|_1(\mathbf{v}_{sa} + \mathbf{v}_{ca}). \end{aligned} \tag{7}$$

Hence, intra-frame point relations can be compensated by inter-frame information communication in a local-to-global manner, which enriches the geometric representations.

Deep Feature Registration

Correspondence mapping (Wang and Solomon 2019a; Pan, Cai, and Liu 2022) plays an important role in PCR, and we discuss that it is also critical for achieving RI in our design. Specifically, although $\mathbf{F}^\ell$ and $\mathbf{F}^g$ are both rotation-invariant by theory, different point sampling methods and the sign ambiguity will cause the final feature not strictly rotation-invariant. To solve this issue, we first examine the correspondence map:

$$m\left(\mathcal{X}, \mathcal{Y}\right) = \frac{\exp\left(\Phi_1(\mathcal{Y})\Phi_2(\mathcal{X})^\top / t\right)}{\sum_{j=1}^{N} \exp\left(\Phi_1(\mathcal{Y})\Phi_2(\boldsymbol{x}_j)^\top / t\right)}, \tag{8}$$

where Φ_1 and Φ_2 are MLPs that project latent embeddings $\mathcal{X}$ and $\mathcal{Y}$ to a shared space, and t controls the variation sensitivity. It can be seen from Eq. 8 that the mapping function m reveals feature similarities in the latent space, and it is also an essential part for 3D point-level contrastive learning in PointContrast (Xie et al. 2020) for the design of InfoNCE losses (Van den Oord, Li, and Vinyals 2018), which have been proven to be equivalent to maximize the mutual information. Based on this observation, we propose a registration loss function $\mathcal{L}_r = \mathcal{L}_r^\ell + \mathcal{L}_r^g$, where $\mathcal{L}_r^\ell$ and $\mathcal{L}_r^g$ represent the registration loss of ($\mathbf{F}^\ell$,$\mathbf{U}$) and ($\mathbf{F}^g$,$\mathbf{U}$). Mathematically, $\mathcal{L}_r^\ell$ is defined as follows:

$$\mathcal{L}_r^\ell = -\sum_{(i,j)\in M} \log \frac{\exp\left(\Phi_1(\mathbf{U}_j)\Phi_2(\mathbf{f}_i^\ell)^\top / t\right)}{\sum_{(\cdot,k)\in M} \exp\left(\Phi_1(\mathbf{U}_k)\Phi_2(\mathbf{f}_i^\ell)^\top / t\right)}. \tag{9}$$

The same rule is followed to compute $\mathcal{L}_r^g$. Although we follow the core idea of PointContrast, we differ from it in that PointContrast defines positive samples based on feature correspondences computed at the same layer level, while our positive samples are defined across layers.

The intuition for the loss design is that the 3D shape is forced to learn about its local region as it has to distinguish it from other parts of different objects. Moreover, we would like to maximize the mutual information between different poses of the 3D shape, as features encoded from different poses should represent the same object, which is very useful in achieving RI in SO(3). Moreover, the mutual information between $\mathbf{F}^\ell$ and $\mathbf{F}^g$ is implicitly maximized, such that shared semantic information about geometric structures can be learned, leading to a more geometrically accurate and discriminative representation. More details about $\mathcal{L}_r^\ell$ can be found in the supplementary material.

Rotation Sensitive	z/z	z/SO(3)	SO(3)/SO(3)
PointNet (Qi et al. 2017a)	89.2	16.2	75.5
PoinNet++ (Qi et al. 2017b)	89.3	28.6	85.0
PCT (Guo et al. 2021)	90.3	37.2	88.5
Rotation Robust	z/z	z/SO(3)	SO(3)/SO(3)
SFCNN (Rao, Lu, and Zhou 2019)	**91.4**	84.8	90.1
RIConv (Zhang et al. 2019)	86.5	86.4	86.4
ClusterNet (Chen et al. 2019)	87.1	87.1	87.1
PR-InvNet (Yu et al. 2020)	89.2	89.2	89.2
RI-GCN (Kim, Park, and Han 2020)	89.5	89.5	89.5
GCAConv (Zhang et al. 2020)	89.0	89.1	89.2
RI-Framework (Li et al. 2021b)	89.4	89.4	89.3
VN-DGCNN (Deng et al. 2021)	89.5	89.5	90.2
SGMNet (Xu et al. 2021)	90.0	90.0	90.0
Li et al. (2021a)	90.2	90.2	90.2
OrientedMP (Luo et al. 2022)	88.4	88.4	88.9
ELGANet (Gu et al. 2022)	90.3	90.3	90.3
Ours	91.0	**91.0**	**91.0**

Table 1: Classification results on ModelNet40. All methods take raw points of 1024×3 as inputs.

Experiments

We evaluate our model on 3D shape classification, part segmentation, and retrieval tasks under rotations, and extensive experiments are conducted to analyze the network design. Detailed model architectures for the three tasks are shown in the supplementary material. We follow (Esteves et al. 2018) for evaluation: training and testing the network under z-axis (z/z); training under z-axis and testing under arbitrary rotations (z/SO(3)); and training and testing under arbitrary rotations (SO(3)/SO(3)).

3D Object Classification

Synthetic Dataset. We first examine the model performance on the synthetic ModelNet40 (Wu et al. 2015) dataset. We sample 1024 points from each data with only xyz coordinates as input features. Hyper-parameters for training follow the same as (Guo et al. 2021), except that points are downsampled in the order of (1024, 512, 128) with feature dimensions of (3, 128, 256). We report and compare our model performance with state-of-the-art (SoTA) methods in Table 1. Both rotation sensitive and robust methods achieve great performance under z/z. However, the former could not generalize well to unseen rotations. Rotation robust methods like SFCNN (Rao, Lu, and Zhou 2019) achieve competitive results under z/z, but their performance is not consistent on z/SO(3) and SO(3)/SO(3) due to the imperfect projection from points to voxels when using spherical solutions. We outperform the recent proposed methods (Luo et al. 2022; Xu et al. 2021; Deng et al. 2021) and achieve an accuracy of 91.0%, proving the superiority of our framework on classification.

Real Dataset. Experiments are also conducted on a real-scanned dataset. ScanObjectNN (Uy et al. 2019) is a commonly used benchmark to explore the robustness to noisy and deformed 3D objects with non-uniform surface density, which includes 2,902 incomplete point clouds in 15 classes.

Method	z/SO(3)	SO(3)/SO(3)
PointNet (Qi et al. 2017a)	16.7	54.7
PointNet++ (Qi et al. 2017b)	15.0	47.4
PCT (Guo et al. 2021)	28.5	45.8
RIConv (Zhang et al. 2019)	78.4	78.1
RI-GCN (Kim, Park, and Han 2020)	80.5	80.6
GCAConv (Zhang et al. 2020)	80.1	80.3
RI-Framework (Li et al. 2021b)	79.8	79.9
LGR-Net (Zhao et al. 2022)	81.2	81.4
VN-DGCNN (Deng et al. 2021)	79.8	80.3
OrientedMP (Luo et al. 2022)	76.7	77.2
Ours	**86.6**	**86.3**

Table 2: Classification results on ScanObjectNN OBJ_BG.

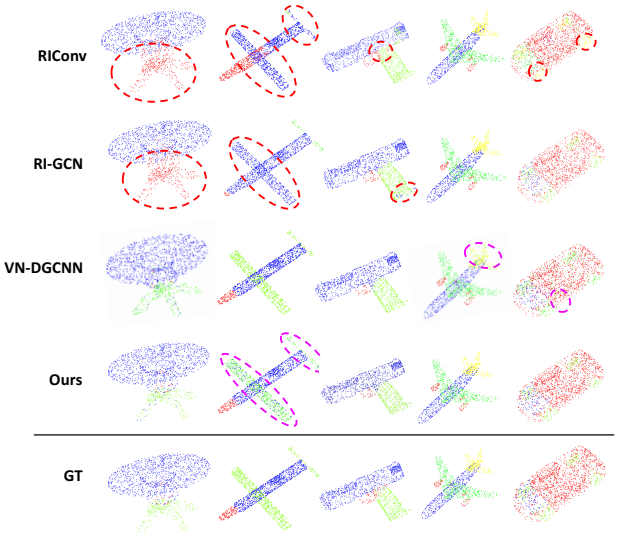

Figure 3: Segmentation comparisons on ShapeNetPart, where ground truth (GT) samples are shown for reference. Red dotted circles indicate obvious failures on certain classes, and purple circles denote the slight difference between our design and VN-DGCNN.

We use *OBJ_BG* subset with the background noise and sample 1,024 points under z/SO(3) and SO(3)/SO(3). Table 2 shows that our model achieves the highest results with excellent consistency with random rotations.

3D Part Segmentation

Shape part segmentation is a more challenging task than object classification. We use ShapeNetPart (Yi et al. 2016) for evaluation, where we sample 2048 points with xyz coordinates as model inputs. The training strategy is the same as the classification task except that the training epoch number is 300. Part-averaged IoU (mIoU) is reported in Table 3, and detailed per-class mIoU values are shown in the supplementary material. Representative methods such as PointNet++ and PCT are vulnerable to rotations. Rotation robust methods present competitive results under z/SO(3), where we achieve the second best result of 80.3%. We give more details of comparison between VN-DGCNN (Deng et al. 2021) and our work in the supplementary material, where our method performs better than VN-DGCNN for several classes. Moreover, qualitative results shown in Fig. 3 present that we can achieve visually better results than VN-DGCNN in certain classes such as the airplane and car. More qualitative results are shown in the supplementary material.

Method	z/SO(3)	SO(3)/SO(3)
PointNet (Qi et al. 2017a)	38.0	62.3
PointNet++ (Qi et al. 2017b)	48.3	76.7
PCT (Guo et al. 2021)	38.5	75.2
RIConv (Zhang et al. 2019)	75.3	75.5
RI-GCN (Kim, Park, and Han 2020)	77.2	77.3
RI-Framework (Li et al. 2021b)	79.2	79.4
LGR-Net (Zhao et al. 2022)	80.0	80.1
VN-DGCNN (Deng et al. 2021)	**81.4**	**81.4**
OrientedMP (Luo et al. 2022)	80.1	<u>80.9</u>
Ours	<u>80.3</u>	80.4

Table 3: Segmentation results on ShapeNetPart. The second best results are underlined.

Method	micro mAP	macro mAP	Score
Spherical CNN (Esteves et al. 2018)	0.685	0.444	0.565
SFCNN (Rao, Lu, and Zhou 2019)	0.705	0.483	0.594
GCAConv (Zhang et al. 2020)	0.708	0.490	0.599
RI-Framework (Li et al. 2021b)	0.707	**0.510**	0.609
Ours	**0.715**	**0.510**	**0.613**

Table 4: Comparisons of SoTA methods on the 3D shape retrieval task.

3D Shape Retrieval

We further conduct 3D shape retrieval experiments on ShapeNetCore55 (Chang et al. 2015), which contains two categories of datasets: normal and perturbed. We only use the perturbed part to validate our model performance under rotations. We combine the training and validation sets and validate our method on the testing set following the training policy of (Esteves et al. 2018). Experimental results are reported in Table 4, where the final score is the average value of micro and macro mean average of precision (mAP) as in (Savva et al. 2017). Similar to the classification task, our method achieves SoTA performance.

Ablation Study

Effectiveness of Transformer Designs. We examine the effectiveness of our transformer design by conducting classification experiments under z/SO(3). We first ablate one or both of the angular embeddings and report the results in Table 5 (models A, B, and C). Model B performs better than model C by 0.4%, which validates our design of feature integration where $\mathbf{M}_i^\ell$ is used as the main source of information. When both angular embeddings are applied, the best result is achieved (*i.e.*, 91.0%). Moreover, we validate our discussion in Section by comparing models D and E. We

Model	$\mathbf{e}^{\alpha}_{sa}$	$\mathbf{e}^{\alpha}_{ca}$	$\mathbf{F}^{g*}$	$\mathbf{A}_{sa}+\mathbf{A}_{ca}$	$\mathcal{L}^{\ell}_{r}$	$\mathcal{L}^{g}_{r}$	Acc.
A				✓	✓	✓	90.0
B	✓			✓	✓	✓	90.6
C		✓		✓	✓	✓	90.2
D	✓	✓	✓	✓	✓	✓	90.2
E	✓	✓			✓	✓	90.4
F	✓	✓		✓			90.0
G	✓	✓		✓	✓		90.2
H	✓	✓		✓		✓	90.6
Ours	✓	✓		✓	✓	✓	**91.0**

Table 5: Module analysis of AIT and loss functions. $\mathbf{F}^{g*}$ means encoding $\mathbf{F}^{g}$ via Intra-frame Aligned Self-attention.

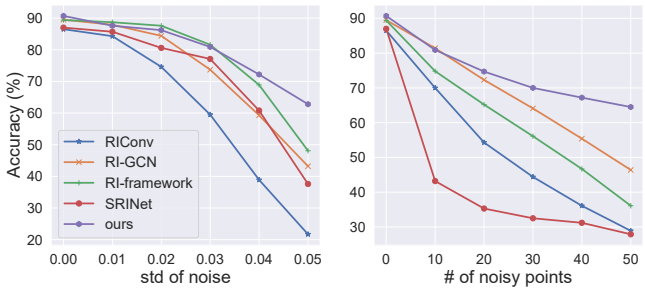

Figure 4: Left: Results on Gaussian noise of zero mean and variant standard deviation values. Right: Results on different numbers of noisy points.

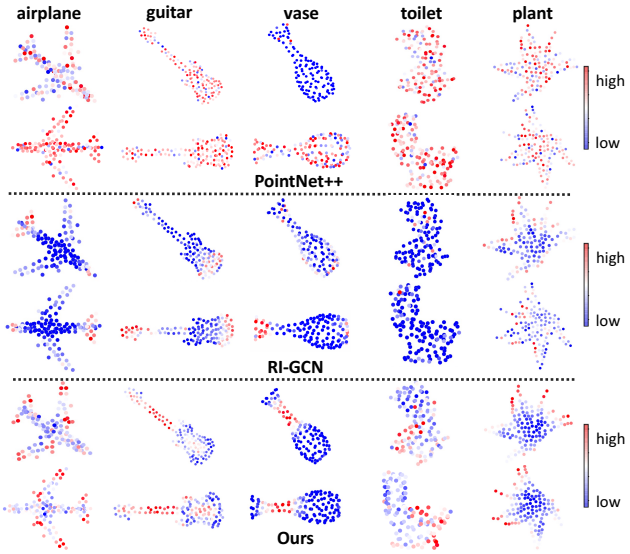

Figure 5: Network attention on PointNet++, RI-GCN and our model.

demonstrate in model D that when encoding $\mathbf{F}^{g}$ in the same way as $\mathbf{F}^{\ell}$, the model performance decreases, which indicates that encoding $\mathbf{F}^{g}$ via self-attention will increase the model overfitting. More analyses can be found in the supplementary material. Finally, we examine the effectiveness of our attention logits-based integration scheme by comparing our model with the conventional method (model E), which applies self- and cross-attention sequentially and repeatedly. We observe that our result is better than model E by 0.6%, indicating that our design is more effective.

Registration Loss. We sequentially ablate $\mathcal{L}^{g}_{r}$ and $\mathcal{L}^{\ell}_{r}$ (models F, G, and H) to check the effectiveness of our registration loss deign. Results in Table 5 demonstrate that we can still achieve a satisfactory result of 90.0% without feature registration. Individual application of $\mathcal{L}^{g}_{r}$ and $\mathcal{L}^{\ell}_{r}$ shows the improvement when forcing the final representation to be close to rotation-invariant features. Moreover, it can be seen that model H performs better than model G, which indicates that intermediate features learned from the global scale are important for shape classification. The best model performance is hence achieved by applying both losses.

Noise Robustness. In real-world applications, raw point clouds contain noisy signals. We conduct experiments to present the model robustness to noise under z/SO(3). Two experiments are conducted: (1) We sample and add Gaussian noise of zero mean and varying standard deviations $\mathcal{N}(0, \sigma^2)$ to the input data; (2) We add outliers sampled from a unit sphere to each object. As shown in Fig. 4 (left), we achieve on par results to RI-Framework when std is low, while we perform better while std increases, indicating that our model is robust against high levels of noise. Besides, as the number of noisy points increases, most methods are heavily affected while we can still achieve good results.

Visualization of Rotation Invariance. We further examine RI of learned features. Specifically, we use Grad-CAM (Selvaraju et al. 2017) to check how the model pays attention to different parts of data samples under different rotations. Results are reported in Fig. 5 with correspondence between gradients and colors shown on the right. RI-GCN presents a good result, but its behavior is not consistent over some classes (*e.g.*, vase and plant) and it does not pay attention to regions that are critical for classification (see toilet), showing inferior performance to ours. PointNet++ shows no resistance to rotations, while our method exhibits a consistent gradient distribution over different parts with random rotations, indicating our network is not affected by rotations.

Conclusion

In this work, we rethink and investigate the close relation between rotation invariance and point cloud registration, based on which we propose a PCR-cored learning framework with three stages. With a pair of rotation-invariant shape descriptors constructed from local and global scales, a comprehensive learning and feature integration module is proposed, Aligned Integration Transformer, to simultaneously effectively align and integrate shape codes via self- and cross-attentions. To further preserve rotation invariance in the final feature representation, a registration loss is proposed to align it with intermediate features, where shared semantic knowledge of geometric parts is also extracted. Extensive experiments demonstrated the superiority and robustness of our designs. In future work, we will examine efficient methods for invariance learning on large-scale point clouds.

References

Afham, M.; Dissanayake, I.; Dissanayake, D.; Dharmasiri, A.; Thilakarathna, K.; and Rodrigo, R. 2022. Crosspoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding. In *CVPR*.

Besl, P. J.; and McKay, N. D. 1992. Method for registration of 3-D shapes. In *Sensor fusion*.

Bro, R.; Acar, E.; and Kolda, T. G. 2008. Resolving the sign ambiguity in the singular value decomposition. In *JoC*.

Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. 2015. Shapenet: An information-rich 3D model repository. In *arXiv:1512.03012*.

Chen, C.; Li, G.; Xu, R.; Chen, T.; Wang, M.; and Lin, L. 2019. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. In *CVPR*.

Chen, C.-F. R.; Fan, Q.; and Panda, R. 2021. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *ICCV*.

Chen, R.; and Cong, Y. 2022. The Devil is in the Pose: Ambiguity-free 3D Rotation-invariant Learning via Pose-aware Convolution. In *CVPR*.

Cheng, R.; Razani, R.; Taghavi, E.; Li, E.; and Liu, B. 2021. (AF)2-S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *CVPR*.

Cohen, T. S.; Geiger, M.; Köhler, J.; and Welling, M. 2018. Spherical CNNs. In *ICLR*.

Deng, C.; Litany, O.; Duan, Y.; Poulenard, A.; Tagliasacchi, A.; and Guibas, L. J. 2021. Vector neurons: A general framework for SO (3)-equivariant networks. In *ICCV*.

Deng, H.; Birdal, T.; and Ilic, S. 2018. PPFNet: Global context aware local features for robust 3d point matching. In *CVPR*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

Drost, B.; Ulrich, M.; Navab, N.; and Ilic, S. 2010. Model globally, match locally: Efficient and robust 3D object recognition. In *CVPR*.

Esteves, C.; Allen-Blanchette, C.; Makadia, A.; and Daniilidis, K. 2018. Learning SO(3) equivariant representations with spherical CNNs. In *ECCV*.

Gu, R.; Wu, Q.; Li, Y.; Kang, W.; Ng, W.; and Wang, Z. 2022. Enhanced local and global learning for rotation-invariant point cloud representation. In *MultiMedia*.

Guo, M.-H.; Cai, J.-X.; Liu, Z.-N.; Mu, T.-J.; Martin, R. R.; and Hu, S.-M. 2021. PCT: Point cloud transformer. In *CVM*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*.

Jaderberg, M.; Simonyan, K.; Zisserman, A.; et al. 2015. Spatial transformer networks. In *NeurIPS*.

Jing, B.; Eismann, S.; Suriana, P.; Townshend, R. J.; and Dror, R. 2020. Learning from protein structure with geometric vector perceptrons. In *ICLR*.

Kim, S.; Park, J.; and Han, B. 2020. Rotation-Invariant Local-to-Global Representation Learning for 3D Point Cloud. In *NeurIPS*.

Li, F.; Fujiwara, K.; Okura, F.; and Matsushita, Y. 2021a. A Closer Look at Rotation-Invariant Deep Point Cloud Analysis. In *ICCV*.

Li, X.; Li, R.; Chen, G.; Fu, C.-W.; Cohen-Or, D.; and Heng, P.-A. 2021b. A rotation-invariant framework for deep point cloud analysis. In *TVCG*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*.

Luo, S.; Li, J.; Guan, J.; Su, Y.; Cheng, C.; Peng, J.; and Ma, J. 2022. Equivariant Point Cloud Analysis via Learning Orientations for Message Passing. In *CVPR*.

Pan, L.; Cai, Z.; and Liu, Z. 2022. Robust Partial-to-Partial Point Cloud Registration in a Full Range. In *IJCV*.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*.

Qin, Z.; Yu, H.; Wang, C.; Guo, Y.; Peng, Y.; and Xu, K. 2022. Geometric Transformer for Fast and Robust Point Cloud Registration. In *CVPR*.

Rao, Y.; Lu, J.; and Zhou, J. 2019. Spherical fractal convolutional neural networks for point cloud recognition. In *CVPR*.

Sanghi, A. 2020. Info3d: Representation learning on 3D objects using mutual information maximization and contrastive learning. In *ECCV*.

Savva, M.; Yu, F.; Su, H.; Kanezaki, A.; Furuya, T.; Ohbuchi, R.; Zhou, Z.; Yu, R.; Bai, S.; Bai, X.; et al. 2017. Large-scale 3D shape retrieval from ShapeNet Core55: SHREC'17 track. In *workshop of 3DOR*.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*.

Spezialetti, R.; Stella, F.; Marcon, M.; Silva, L.; Salti, S.; and Di Stefano, L. 2020. Learning to orient surfaces by self-supervised spherical cnns. In *NeurIPS*.

Sun, W.; Tagliasacchi, A.; Deng, B.; Sabour, S.; Yazdani, S.; Hinton, G. E.; and Yi, K. M. 2021. Canonical Capsules: Self-Supervised Capsules in Canonical Pose. In *NeurIPS*.

Thomas, H. 2020. Rotation-Invariant Point Convolution With Multiple Equivariant Alignments. In *3DV*.

Tombari, F.; Salti, S.; and Stefano, L. D. 2010. Unique signatures of histograms for local surface description. In *ECCV*.

Uy, M. A.; Pham, Q.-H.; Hua, B.-S.; Nguyen, D. T.; and Yeung, S.-K. 2019. Revisiting Point Cloud Classification: A New Benchmark Dataset and Classification Model on Real-World Data. In *ICCV*.

Van den Oord, A.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. In *arXiv:1807.03748*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*.

Wang, Y.; and Solomon, J. M. 2019a. Deep closest point: Learning representations for point cloud registration. In *ICCV*.

Wang, Y.; and Solomon, J. M. 2019b. Prnet: Self-supervised learning for partial-to-partial registration. In *NeurIPS*.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3D shapenets: A deep representation for volumetric shapes. In *CVPR*.

Xie, S.; Gu, J.; Guo, D.; Qi, C. R.; Guibas, L.; and Litany, O. 2020. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *ECCV*.

Xu, J.; Tang, X.; Zhu, Y.; Sun, J.; and Pu, S. 2021. SGM-Net: Learning Rotation-Invariant Point Cloud Representations via Sorted Gram Matrix. In *ICCV*.

Yew, Z. J.; and Lee, G. H. 2020. Rpm-net: Robust point matching using learned features. In *CVPR*.

Yi, L.; Kim, V. G.; Ceylan, D.; Shen, I.-C.; Yan, M.; Su, H.; Lu, C.; Huang, Q.; Sheffer, A.; and Guibas, L. 2016. A scalable active framework for region annotation in 3D shape collections. In *ACM ToG*.

Yu, H.; Li, F.; Saleh, M.; Busam, B.; and Ilic, S. 2021a. CoFiNet: Reliable Coarse-to-fine Correspondences for Robust PointCloud Registration. In *NeurIPS*.

Yu, J.; Zhang, C.; Wang, H.; Zhang, D.; Song, Y.; Xiang, T.; Liu, D.; and Cai, W. 2021b. 3D Medical Point Transformer: Introducing Convolution to Attention Networks for Medical Point Cloud Analysis. In *arXiv:2112.04863*.

Yu, R.; Wei, X.; Tombari, F.; and Sun, J. 2020. Deep Positional and Relational Feature Learning for Rotation-Invariant Point Cloud Analysis. In *ECCV*.

Yuan, W.; Eckart, B.; Kim, K.; Jampani, V.; Fox, D.; and Kautz, J. 2020. Deepgmr: Learning latent gaussian mixture models for registration. In *ECCV*.

Zhang, C.; Yu, J.; Song, Y.; and Cai, W. 2021. Exploiting Edge-Oriented Reasoning for 3D Point-Based Scene Graph Analysis. In *CVPR*.

Zhang, Z.; Hua, B.-S.; Chen, W.; Tian, Y.; and Yeung, S.-K. 2020. Global context aware convolutions for 3D point cloud understanding. In *3DV*.

Zhang, Z.; Hua, B.-S.; Rosen, D. W.; and Yeung, S.-K. 2019. Rotation invariant convolutions for 3D point clouds deep learning. In *3DV*.

Zhao, C.; Yang, J.; Xiong, X.; Zhu, A.; Cao, Z.; and Li, X. 2022. Rotation invariant point cloud analysis: Where local geometry meets global topology. In *Pattern Recognition*.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *ICCV*.