Deep Hierarchical Rotation Invariance Learning with Exact Geometry Feature Representation for Point Cloud Classification

Jianjie Lin, Markus Rickert, and Alois Knoll

Abstract— **Rotation invariance is a crucial property for 3D object classification, which is still a challenging task. State-of-the-art deep learning-based works require a massive amount of data augmentation to tackle this problem. This is however inefficient and classification accuracy suffers a sharp drop in experiments with arbitrary rotations. We introduce a new descriptor that can globally and locally capture the surface geometry properties and is based on a combination of spherical harmonics energy and point feature representation. The proposed descriptor is proven to fulfill the rotation-invariant property. A limited bandwidth spherical harmonics energy descriptor globally describes a 3D shape and its rotation-invariant property is proven by utilizing the properties of a Wigner D-matrix, while the point feature representation captures the local features with a KNN to build the connection to its neighborhood. We propose a new network structure by extending PointNet++ with several adaptations that can hierarchically and efficiently exploit local rotation-invariant features. Extensive experimental results show that our proposed method dramatically outperforms most state-of-the-art approaches on standard rotation-augmented 3D object classification benchmarks as well as in robustness experiments on point perturbation, point density, and partial point clouds.**

I. Introduction

Convolutional neural networks (CNN) [1] have shown tremendous success in image processing due to their translation-invariant capability of detecting local patterns regardless of their position in the image and their ability to process regular data, such as image grids or 3D voxels. However, the more challenging rotation-invariant property is still missing in the designed structure [2]. Data augmentation is a common approach to address this issue. The infinite property of the rotation group howver makes this approach less efficient and comes with a high computational cost. A big neural network with rotation-augmented data is required to generalize the data set. In 3D, geometric irregular data formats such as point clouds increase the difficulty of handling the rotation transformation, while irregular data formats suffer from a permutation problem $N!$. To address this issue and to inherit the benefits of convolutional networks, which can process regular data formats, previous work such as [3], [4] voxelized geometric shapes. [5], [3], [6] proposed a rotation-equivariant network with newly designed spherical convolutional operators. However, the voxelization of 3D geometry induces a trade-off between resolution and computational cost. The pioneering work PointNet used a spatial transformation network to learn an affine transformation, which still did not fulfill the requirement. Inspired by CNNs, which use different receptive fields to aggregate the local features, DGCNN used a dynamic k-nearest neighbors algorithm (KNN) to exploit local information. However, its classification results still suffer a sharp drop in rotation experiments.

For alleviating the issue, we introduce two different rotation-invariant features (RIF). The first one is spherical harmonics [7], which transform the Cartesian pose to the spectral domain by using a non-commutative Fourier analysis methods and are related to the power spectrum in the perspective of signal processing. The second feature can locally describe the geometry relationship by creating a Darboux frame at each object point with a KNN-graph. This geometry point feature is also utilized in the point feature histogram [8] and fast point feature histogram [9]. The rotation-invariant feature aims at separating the rotated point cloud and the network so that the input space is invariant to arbitrary rotation perturbation. Furthermore, we design a new network structure that can hierarchically extract the local features by applying the farthest point sampling strategy. The proposed network structure is composed of RIMapping, PF Abstraction, and Classification blocks. In the RIMapping block, rotation-invariant features are fed to a feature transformation network, which maps the lower level feature to a high level embedding space. Two consequent abstraction layers work on these high-level embedding features. For further exploiting the local geometry information, a fully connected point feature graph is built on each cluster and the resultant features are fed to a point feature transformation. Afterward, a global abstraction layer can aggregate all previous embedding features together to obtain a global feature. The Classification block is a standard fully connected network to classify the objects. We evaluated our proposed network on ModelNet40 with different experimental settings and achieve or exceed most state-of-the-art approaches.

Our primary contributions are two-fold: a) we introduce a novel geometry rotation-invariant feature descriptor, which can globally and locally represent a 3D shape. b) a new rotation-invariant classification network structure is designed, which can efficiently exploit local geometric features.

II. Related Work

With recent good results from deep learning in image-based recognition, 3D visual recognition has also received more attention and rapid development. It benefits from deep learning in extracting and learning geometric features more

Jianjie Lin, Markus Rickert, Alois Knoll are with Robotics, Artificial Intelligence and Real-Time Systems, Department of Informatics, Technische Universität München, Munich, Germany jianjie.lin@tum.de {rickert,knoll}@in.tum.de

efficiently, but the recognition of 3D geometry differs from image-based recognition in many factors. One main aspect are the representation formats, where 3D geometry uses various methods such as point cloud-based representation, implicit surfaces based representation, or volumetric-based representations. These different formats lead to different learning methods. In contrast, the imaged-based representation is interpreted in regular data, where the conventional CNN is designed to handle such regular data. The permutation $N!$ is a common problem in the irregular data format. Based on these observations, previous work seeks to utilize benefits from conventional CNNs by voxelizing the 3D geometric shape [4], [10], [11], [12] or by using multi-view images [11], [13], [14]. However, the trade-off between the resolution and computational cost makes generalization impossible. Most 3D convolutional neural networks sacrifice high resolution to obtain fast calculations to build upon a shallow network. To alleviate the negative impact of accuracy due to resolution, an Octet [15] is proposed by hierarchically partitioning the space using a set of unbalanced octrees to exploit sparse input data.

In contrast to a volumetric representation, PointNet [16] is the first work that directly feeds the point cloud into a set of shared MLP networks and uses the max pool operator to extract global features. It shows a significant improvement in the perspective of 3D shape reasoning and computational cost. PointNet, however, does not extract local information. Follow-up work such as PointNet++ [17] progressively aggregated local features using the farthest point sampling strategy. Moreover, DGCNN [18] introduced a dynamic KNN to build a local graph and aggregated the edge features to obtain a better feature representation. A point-based neural network satisfies many properties, e.g., permutation invariance with a shared MLP and max pool operator and translation equivariance with a relu operator [5]. This network is shown to solve many classical problems such as classification, part segmentation, and instance segmentation. The rotation-invariant property is however still missing in the designed structures. PointNet applies a spatial transformer network [19] to predict an affine transformation matrix. Other work attempts to augment the data set by generating a lot of SO(3) combinations. However, SO(3) is infinite, and data augmentation wastes computational resources and cannot guarantee effectiveness. To alleviate this issue, previous work proposed a rotation-equivalence network structure. [20], [5], [3] designed a spherical-based convolutional operator utilizing the properties of spherical harmonics. [21] proposed tensor field networks, which map point clouds to point clouds under the constraint of SE(3) equivariance by utilizing a spherical harmonics filter. Spherical representations for 3D data are not novel and have been used for retrieval tasks before the deep learning era [7], [22].

Spherical-based CNNs were initially designed for voxelized shapes and suffered a loss of geometric information, as there is no bijection between $\mathbb{R}^3$ and 2-dimensional sphere S^2 [23] as mentioned above. Instead of proposing a new convolutional operator, [23] introduced rigorously rotation-invariant (RRI) features by transforming the point from Cartesian space into an embedding space and showed a good improvement in experiments. However, the RRI features focus only on the local feature using the same dynamic KNN as DGCNN.

III. Geometric RIF Descriptor

Given a set of transformations $\boldsymbol{T}_{g_i} : \mathcal{V} \to \mathcal{V}$ for $g_i \in \mathrm{SO}(3)$, a rotation-invariant function $\phi(\cdot)$ has the property

$$\phi(\boldsymbol{T}_{g_1}\boldsymbol{q}) = \phi(\boldsymbol{T}_{g_2}\boldsymbol{q}), \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^3$ is a point in the Cartesian coordinate system. Pioneering works in processing point clouds are PointNet and DGCNN, where EdgeConv from DGCNN and mini-PointNet from PointNet++ utilize the edge feature represented as an implicit geometry feature by considering geometric constraints between points. The edge features $\boldsymbol{x}_i - \boldsymbol{x}_j$ and pose point $\boldsymbol{x}_i$ do not satisfy the property described in (1). Furthermore, edge features under a dynamic KNN can only represent the local geometric context for point clouds in the embedding space. For alleviating this issue, two rotation-invariant descriptors will be introduced, that globally (spherical harmonics descriptor) and locally (point feature descriptor) represent the geometry shape.

A. RI Spherical Harmonics (SH) Descriptor

Definition: Spherical harmonics define an orthonormal basis over the sphere, with the parameterization

$$(x, y, z) = (\sin(\theta)\cos(\varphi), \sin(\theta)\sin(\varphi), \cos(\theta)), \tag{2}$$

where (x, y, z) is a location defined on a unit sphere with co-latitude θ and longitude φ and the orthonormal basis function given by Rodrigues' formula can be described as

$$Y_l^m(\theta, \varphi) = K_l^m P_l^m(\cos\theta) e^{im\varphi}, \tag{3}$$

with the normalized constant variable K_l^m and the associated Legendre polynomials P_l^m. The parameters l and m are the spherical harmonic degree and order, respectively. Furthermore, the order should satisfy the constraint $-l \leq m \leq l$. The real spherical harmonics are sometimes known as tesseral spherical harmonics. These functions have the same orthonormality properties as the complex ones above. The harmonics with $m > 0$ are said to be of cosine type and those with $m < 0$ of sine type. The reason for this can be seen by writing the functions in terms of the Legendre polynomials P_l^m with Condon-Shortley phase convention as

$$Y_{lm} = \begin{cases} (-1)^m \sqrt{2} K_{|m|}^l P_l^{|m|}(\cos\theta)\sin(|m|\varphi) & \text{if } m < 0 \\ \sqrt{\frac{2l+1}{4\pi}} P_l^m(\cos\theta) & \text{if } m = 0 \\ (-1)^m \sqrt{2} K_m^l P_l^m(\cos\theta)\cos(m\varphi) & \text{if } m > 0 \end{cases} \tag{4}$$

Moreover, for any rotation matrix $\boldsymbol{R} \in \mathrm{SO}(3)$, the rotated SH $Y_l^m(\boldsymbol{R}\cdot)$ can be expressed as a linear combination of other SHs of the same degree l

$$Y_l^m(\boldsymbol{R}\cdot) = \sum_{m'=-l}^{l} \left[D_{\boldsymbol{R}}^{(l)}[m, m'] \right]^* Y_l^{m'}, \tag{5}$$

where $\boldsymbol{D}_{\boldsymbol{R}}^{(l)}[m, m'] \in \mathbb{C}^{(2l+1)\times(2l+1)}$ is called the Wigner D-matrix. Note that the Wigner matrices $\boldsymbol{D}^{l}$ are all orthonormal and irreducible representations of SO(3) [24], which considers them as *smallest* representations possible. In accordance with the unitary of $\boldsymbol{D}_{\boldsymbol{R}}^{(l)}$, the energy within a subspace is preserved. Therefore, for any given vector $\boldsymbol{c} \in \mathbb{C}^{2l+1}$, the Wigner D-matrix shows a norm preservation property [25], [26] as $\left\| \boldsymbol{D}_{\boldsymbol{R}}^{(l)} \boldsymbol{c} \right\| = \|\boldsymbol{c}\|$. The theory of spherical harmonics says that any spherical function $\mathrm{f}(\theta, \varphi)$ is decomposed as the sum of its harmonics:

$$f(\theta, \varphi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} a_{lm} Y_l^m(\theta, \varphi), \tag{6}$$

with the coefficient a_l^m. Equation (6) can be seen as a kind of Fourier series on the sphere.

*1) **Information loss for a limited bandwidth**:* Since we cannot solve $l \to \infty$, we limit the band l to a constant degree $n_{\mathrm{sh,deg}}$. The information loss is defined as

$$\text{Loss} = \left\| \sum_{l=0}^{n_{\mathrm{sh,deg}}} f_l - \sum_{l=0}^{\infty} f_l \right\|_2 . \tag{7}$$

Furthermore, the numerical solution of coefficients a_l^m can be approximated by using the Monte Carlo integration approach.

$$a_l^m = \frac{4\pi}{n_{\mathrm{sh,deg}}} \sum_{j=0}^{n_{\mathrm{sh,deg}}} f_j\left(\theta_m, \phi_m\right) Y_{l,j}^m\left(\theta_m, \phi_m\right) \tag{8}$$

*2) **SH energy descriptor**:* Polygonal-based surface representations are typically described as Cartesian coordinates (x, y, z). For spherical harmonics, the surfaces are represented by $\mathrm{f}(\theta, \phi)$, therefore the mesh must be transformed into spherical polar coordinates (r, θ, ϕ) about the origin. In this case we define $\mathrm{f}(\theta, \phi) = r$ [27] with the energy spectrum descriptor of spherical harmonics [7]

$$\mathcal{X}_{sh}(f) = \left\{ \left\| f_0(\theta, \varphi) \right\|, \left\| f_1(\theta, \varphi) \right\|, \ldots \right\} \tag{9}$$

with the frequency components

$$f_l = \left[a_{l,-l} Y_l^{-l}, a_{l,-l+1} Y_l^{-l+1}. \cdots, a_{l,l} Y_l^{l} \right] . \tag{10}$$

Utilizing the norm preservation property of Wigner D-matrices [7], [20], [26], we can prove that $\|f_l\|$ is a rotation-invariant descriptor.

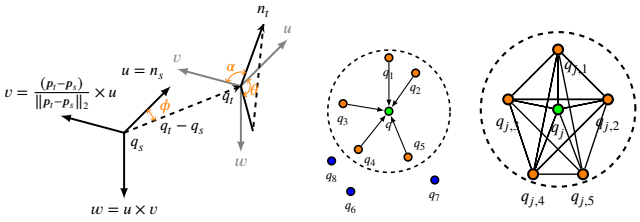
B. RI Point Feature (PF) Descriptors

We employ point feature representations to encode the neighborhood's geometrical properties, which provides an overall point density and pose invariant multi-value feature. The surface normal [28] is estimated by using PCA on the k-neighborhood. Furthermore, for each pair $\boldsymbol{p}_s$ and $\boldsymbol{q}_t$ with $\boldsymbol{q}_t \in \mathcal{N}(\boldsymbol{p}_s)$, Darboux frame at $\langle \boldsymbol{p}_s, \boldsymbol{n}_s \rangle$ is defined as

$$\mathbf{u} = \boldsymbol{n}_s, \quad \mathbf{v} = \frac{(p_t - p_s)}{\|p_t - p_s\|_2} \times \mathbf{u}, \quad \mathbf{w} = \mathbf{u} \times \mathbf{v}. \tag{11}$$

The point features descriptor [29] is described as a quadruplet $\langle \alpha, \phi, \theta, d_{\mathrm{st}} \rangle$ with

$$\begin{aligned} d_{\mathrm{st}} &= \|\boldsymbol{p}_t - \boldsymbol{p}_s\|_2 , & \alpha &= \mathbf{v} \cdot \boldsymbol{n}_t , \\ \phi &= \mathbf{u} \cdot \frac{(p_t - p_s)}{d_{st}} , & \theta &= \operatorname{atan2}\left(\mathbf{w} \cdot \boldsymbol{n}_t, \mathbf{u} \cdot \boldsymbol{n}_t\right) . \end{aligned} \tag{12}$$



(a) (b) (c)

Fig. 1: (a) illustrates the Darboux frame between a point pair. (b) calculates the simplified point features of a source point $\boldsymbol{q}$ with a KNN graph for each point pair. A fully connect point feature (PF) graph (c) is built on a given source point $\boldsymbol{q}_i$.

Furthermore, we augment the distance $r_s = \|\boldsymbol{p}_s\|_2$ to the point feature, therefore, we get a quintuple feature descriptor.

*1) **Proof of rotation-invariant property**:* Given a point set $S = \left\{ \boldsymbol{p}_i \mid \boldsymbol{p}_i \in \mathbb{R}^n \right\}_{i=0}^{N-1}$. It is obvious that the L^2 norm is a rotation-invariant operator $\mathbb{R}^n$ to $\mathbb{R}$ due to norm preservation: $\|\boldsymbol{R}\boldsymbol{x}\|_2^2 = \|\boldsymbol{x}\|_2^2$. It can be easily extended that the inner product $\langle \cdot, \cdot \rangle$ between two arbitrary points preserves the rotation-invariant property. In addition, the cross product has the property $\boldsymbol{R}\mathbf{a} \times \boldsymbol{R}\mathbf{b} = \boldsymbol{R}(\mathbf{a} \times \mathbf{b})$ under proper rotations $\boldsymbol{R}$. We define the Darboux frame at $\boldsymbol{q}_i$ as a triple tuple: $\mathcal{O}_i = \langle \boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i \rangle$. By applying a rotation matrix to the point set, we can get $\boldsymbol{q}_j = \boldsymbol{R}\boldsymbol{q}_i$ with the corresponding Darboux frame $\mathcal{O}_j = \langle \boldsymbol{u}_j, \boldsymbol{v}_j, \boldsymbol{w}_j \rangle$. As a result, we can conclude that $\mathcal{O}_j = \boldsymbol{R}\mathcal{O}_i = \langle \boldsymbol{R}\boldsymbol{u}_i, \boldsymbol{R}\boldsymbol{v}_i, \boldsymbol{R}\boldsymbol{w}_i \rangle$. The PF descriptor is proven to be rotation-invariant:

$$d_{st,j} = \left\| \boldsymbol{p}_{t,j} - \boldsymbol{p}_{s,j} \right\|_2 = \left\| \boldsymbol{R}\boldsymbol{p}_{t,i} - \boldsymbol{R}\boldsymbol{p}_{s,i} \right\|_2 = d_{st,i} = d_{st} \tag{13}$$

$$\alpha_j = \left\langle \boldsymbol{v}_j, \boldsymbol{n}_{t,j} \right\rangle = \left\langle \boldsymbol{R}\boldsymbol{v}_i, \boldsymbol{R}\boldsymbol{n}_{t,i} \right\rangle = \left\langle \boldsymbol{v}_i, \boldsymbol{n}_{t,i} \right\rangle = \alpha_i \tag{14}$$

$$\phi_j = \left\langle \boldsymbol{u}_j, \frac{(\boldsymbol{p}_{t,j} - \boldsymbol{p}_{s,j})}{d_{st}} \right\rangle = \left\langle \boldsymbol{R}\boldsymbol{u}_i, \boldsymbol{R}\frac{(\boldsymbol{p}_{t,i} - \boldsymbol{p}_{s,i})}{d_{st}} \right\rangle = \phi_i \tag{15}$$

$$\begin{aligned} \theta_j &= \operatorname{atan2}\left(\left\langle \boldsymbol{w}_j, \boldsymbol{n}_{t,j} \right\rangle, \left\langle \boldsymbol{u}_j, \boldsymbol{n}_{t,j} \right\rangle \right) \\ &= \operatorname{atan2}\left(\left\langle \boldsymbol{R}\boldsymbol{w}_i, \boldsymbol{R}\boldsymbol{n}_{t,i} \right\rangle, \left\langle \boldsymbol{R}\boldsymbol{u}_i, \boldsymbol{R}\boldsymbol{n}_{t,i} \right\rangle \right) = \theta_i \end{aligned} \tag{16}$$

C. Geometry RIF-Descriptor

The SH-energy and PF descriptors are shown to be rotation-invariant descriptors. The SH-energy descriptor focuses on capturing the global features of the 3D shape and the PF descriptor aims at describing the local features. It is straightforward to concatenate both descriptors and this results in the rotation-invariant feature (RIF)-descriptor at $\boldsymbol{q}_i$

$$\mathcal{X}_{rif,i} = \left[(\mathcal{X}_{sh,i}, \mathcal{X}_{pf,i,0}), \quad \ldots, \quad (\mathcal{X}_{sh,i}, \mathcal{X}_{pf,i,k}) \right]^{\mathrm{T}}, \tag{17}$$

where $\mathcal{X}_{rif,i} \in \mathbb{R}^{k \times (n_{\mathrm{sh}} + n_{\mathrm{pf}})}$ with point feature descriptor

$$\mathcal{X}_{pf,i,j} = [d_{i,j}, \alpha_{i,j}, \phi_{i,j}, \theta_{i,j}, r_{s,i}] \in \mathbb{R}^{n_{\mathrm{pf}}}, \tag{18}$$

with $n_{\mathrm{pf}} = 5$ and the spherical harmonics energy descriptor

$$\mathcal{X}_{sh,i} = \left[\left\| f_{0,i} \right\|, , \ldots, \left\| f_{n_{\mathrm{sh,deg}},i} \right\| \right] \in \mathbb{R}^{n_{\mathrm{sh}}}, n_{\mathrm{sh}} = n_{\mathrm{sh,deg}} + 1 . \tag{19}$$

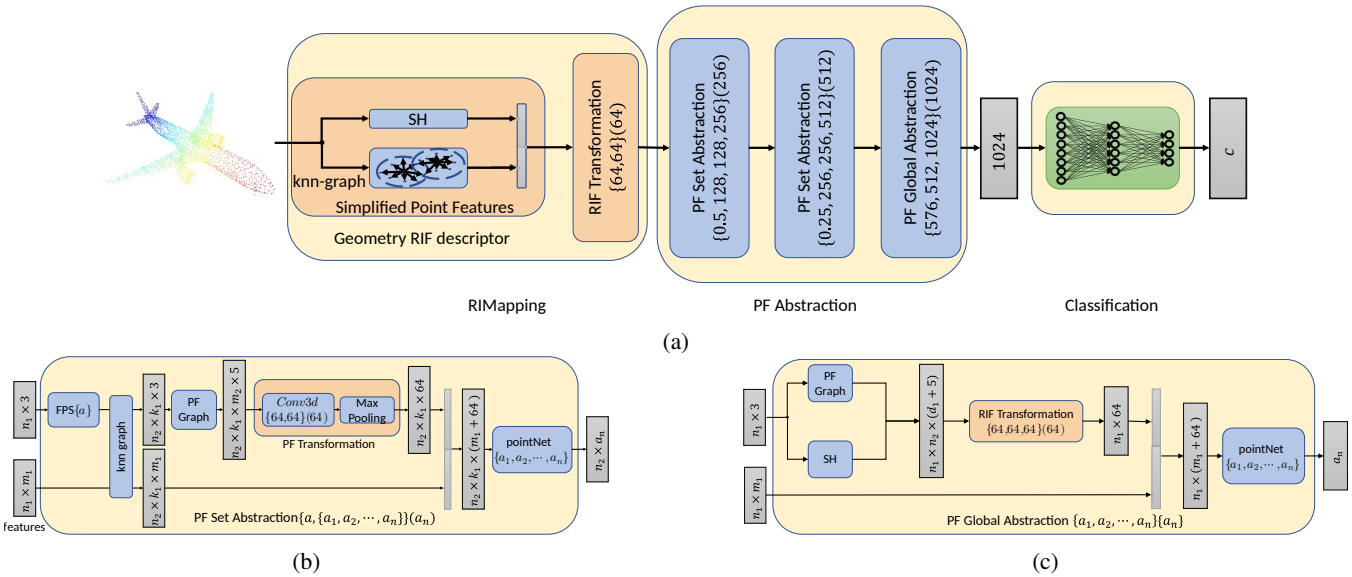


Fig. 2: The model architecture (a) consists of the RIMapping, PF Abstraction, and Classification blocks. In the RIMapping block, a spherical harmonics energy descriptor and simplified point features (SPF) at each point are computed to form a Geometry RIF descriptor, which is fed to a RIF Transformation to extract a high-level feature. In the PF Abstraction Block, we have two PF Set Abstraction layers (b) together with a PF Global Abstraction layer (c) to obtain a final global feature, which is used in a fully-connected Classification network.

IV. Network Architecture

The proposed rotation-invariant network consists of three blocks: RIMapping, PF Abstraction, and Classification, where the latter is a feed-forward network. Our main contributions are on the design of the RIMapping and PF Abstraction blocks.

A. Pre-Alignment with PCA

In an experiment on rotation invariance, we transfer a point cloud set $\boldsymbol{q}$ with an arbitrary rotation matrix $\boldsymbol{R}$, resulting in a rotated clone $\boldsymbol{q}^1 = \boldsymbol{R}\boldsymbol{q}$. According to [22], we pre-align each input $\boldsymbol{q}$ based on the PCA to its principle axes, which are indicated as orthonormal coordinates $\boldsymbol{R}_0$, with the formula of $\boldsymbol{q}_1 = \boldsymbol{R}_0^{\mathrm{T}}\boldsymbol{q}$. It can be shown that the PCA alignment for a rotated clone $\boldsymbol{q}^1$ with corresponding orthonormal coordinates $\boldsymbol{R}_1 = \boldsymbol{R}\boldsymbol{R}_0$ leads to $\boldsymbol{q}_2 = \boldsymbol{R}_1^{\mathrm{T}}\boldsymbol{q}^1 = \boldsymbol{R}_0^{\mathrm{T}}\boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{q} = \boldsymbol{R}_0^{\mathrm{T}}\boldsymbol{q} = \boldsymbol{q}_1$. Therefore, pre-alignment can reduce the impact of the rotation matrix on the network.

B. RIMapping Block

The RIMapping block consists of the Geometry RIF descriptor and RIF Transformation. For acquiring the rotation-invariant features, we leverage the spherical harmonics and point feature representation with a KNN to enrich geometric features for the point cloud, which is represented as a rotation-invariant descriptor with a size of $\mathbb{R}^{n\times k_1\times(n_{\mathrm{sh}}+n_{\mathrm{pf}})}$ and k_1-neighborhood that can globally and locally manifest a 3D shape. This descriptor provides low-level geometric clues for high-level geometric feature learning, realized with a RIF Transformation. The RIF Transformation layer utilizes a mini PointNet (without input and feature transformation), consisting of a set of shared Conv2d layers with kernel size equal to 1, to extract a global feature employing a max-pooling operator. The output of RIF Transformation is indicated as embedding feature with a size of $\mathbb{R}^{n\times a_n}$. The PF Transformation has the same structure as the RIF Transformation, apart from the different input sizes. Both Transformations intend to aggregate the local details by calculating a weighted average of neighboring features through a shared local fully-connected layer.

C. PF Abstraction Block

*1) **PF Set Abstraction**:* The extracted information from the RIMapping block is still insufficient for the precise classification task, as max-pooling can only describe an outline and some local details could be omitted. To address the problem, we propose the PF Set Abstraction Layer to hierarchically exploit the local features, which consists of the sampling layer, grouping layer, and PointNet layer. PointNet++ inspires PF Set Abstraction. However, there are several significant adaptations inside the grouping layer. In the sampling layer, we use iterative farthest point sampling (FPS) to obtain n_2 points, indicated as $\boldsymbol{P}_i, i \in [0, \cdots, n_2]$. Each point $\boldsymbol{P}_i$ is the center of a local region C_i. In the sequence, a KNN graph is built at point $\boldsymbol{P}_i$ to obtain k_1-neighborhood, indicated as $[\boldsymbol{P}_{0,i}, \boldsymbol{P}_{1,i}, \cdots, \boldsymbol{P}_{k_1,i}]$, with $i \in [0, \cdots, n_2]$. In contrast to PointNet++, which combines the point with the feature from the last layer and works as input for the PointNet layer, we utilize a PF graph to convert a point to a rotation-invariant feature, where PF graph is a fully connected graph (Fig. 1c) and built at each local region C_i. Then, we can get a feature with a size of $\mathbb{R}^{k_1-1\times n_{\mathrm{pf}}}$ for each point $\boldsymbol{P}_{n_k,i}$ in the region C_i. In the end, a new rotation-invariant point feature for all local regions is obtained, indicated as $\boldsymbol{f}_{PF} \in \mathbb{R}^{n_2\times k_1\times k_1-1\times n_{\mathrm{pf}}}$.

Sequentially, we apply the PF Transformation to extract a feature for each local region. We concatenate the previous feature at each center point $\boldsymbol{P}_i$ with the newly extracted feature to form a new feature representation and feed it to the PointNet layer.

*2) **PF Global Abstraction**:* The global abstraction is the successor layer to the PF Set Abstraction layer, which reduces the original input cloud to $\boldsymbol{X}_2 \in \mathbb{R}^{n_1 \times 3}$. We build a PF graph at the reduced point set and concatenate it with its spherical energy descriptor, which leads to a new representation with a size of $\mathbb{R}^{n_1 \times (n_1 - 1) \times (n_{\text{pf}} + n_{\text{sh}})}$. In the sequence, we apply the RIF Transformation for extracting a new feature representation. This new feature will concatenate with the feature from the PF Set Abstraction layer. In the end, a mini PointNet is applied to obtain a global feature.

V. Experiments

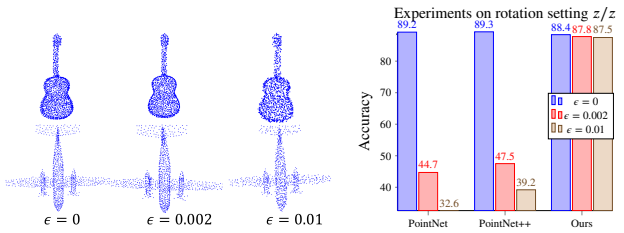
We evaluate our approach regarding rotation robustness and compare it with other state-of-the-art methods. We use ModelNet40 [30] as data set for validating the effectiveness of the proposed network structure. ModelNet40 consists of 40 categories in the form of CAD models (mostly human-made). We use the official split with 9843 shapes for training and 2468 for testing. We apply the farthest point sampling algorithm to obtain 1024 points on mesh faces according to the face area and then shift and normalize the point into a unit sphere with centroid on the origin. During training, we use Adam [31] for 200 epochs with an initial learning rate of 10^{-3}. The algorithm is implemented with PyTorch on Linux with one GeForce RTX 2080Ti GPU.

A. Evaluation of Rotation Robustness

For evaluating the property of rotation robustness, we multiply each point cloud from ModelNet40 with a randomly sampled rotation matrix. Based on the same principle as [20], we evaluate our model using three different settings: a) training and testing with azimuthal rotations (z/z), b) training and testing with arbitrary rotations (SO(3)/SO(3)), c) training with azimuth rotations while testing with arbitrary rotations (z/SO(3)). The results are presented in Table I. It can be seen that most networks exhibit a sharp drop in performance in the settings SO(3)/SO(3) and z/SO(3), in particular in the latter one. The network DGCNN [18] shows an outstanding performance in the setting z/z with an accuracy of 92.2 %, while it only achieves 21.1 % in z/SO(3). DGCNN applies the point directly for classification, which changes dramatically when applying a rotation matrix in SO(3). As mentioned before, that point cloud is not a rotation-invariant representation. This is also a common problem for the PointNet-based network. The spherical CNN [20] uses a spherical harmonics-based convolution layer by rasterizing the point cloud, which shows a significant improvement in the setting of z/SO(3). However, the difference between its best performance z/z and z/SO(3) is still significant with a value of 10.3 %. ClusterNet [23] uses the RRI representation together with a cluster abstraction to increase classification performance with a result of 87.1 % in each setting. Note that the result of ClusterNet is directly cited from [23], as the code is not available as open source. Table I demonstrates that our approach achieves the best performance in the settings z/SO(3) and SO(3)/SO(3) with $n_{\text{sh,deg}} = 20, 30$. The difference in the results between each setting is very small, approximately 0.2 %. Based on these observations, we can conclude that our proposed network shows the best performance regarding rotation robustness.

TABLE I: Comparison of rotation robustness on rotation-augmented ModelNet40 benchmark. Our proposed network shows the best performance in the settings z/SO(3) and SO(3)/SO(3). Note, values are given as a percentage.

Method	Input(size)	z/z	z/SO(3)	SO(3)/SO(3)
PointNet [16]	pc (1024×3)	89.2	14.7	83.6
PointNet++ [17]	pc (1024×3)	89.3	28.6	85.0
VoxNet [4]	voxel (30^3)	83.0	-	73.0
RotationNet 20x [14]	voxel (20×224^2)	92.4	20.0	80.0
SO-Net [32]	pc+normal (5000×6)	**92.6**	21.1	80.2
DGCNN [18]	pc (1024×3)	92.2	33.5	81.1
Spherical CNN [20]	voxel 2×64^2	88.9	78.6	86.9
ClusterNet [23]	pc (1024×3)	87.1	87.1	87.1
ours($n_{\text{sh,deg}}$=20)	pc(1024×3)	88.4	88.6	88.5
ours($n_{\text{sh,deg}}$=30)	pc(1024×3)	88.6	**88.7**	**88.8**



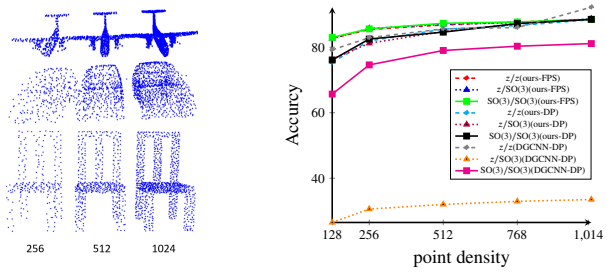
(a) (b)

Fig. 3: (a) shows the point cloud with different noise levels and (b) shows the comparison results against point perturbations.

B. Robustness Tests

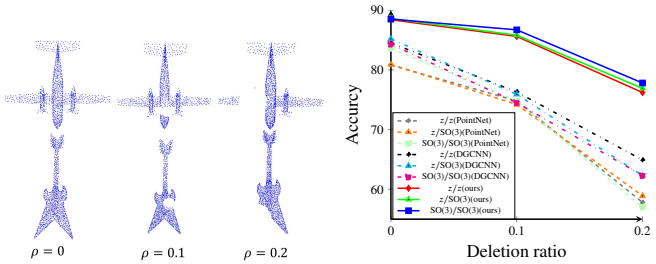
1) Evaluation of model against point cloud perturbation: For further evaluation of robustness against perturbation, we conducted experiments by adding perturbation at each point. Many studies have shown that deep learning-based networks can be fooled by using an adversarial attack. Following the same principle, we add a perturbation value to each point with $\|\delta\| < \epsilon$, where ϵ is set between 0.002 and 0.01, as shown in Fig. 3a. The results are listed in Fig. 3b. It can be seen that in these two different perturbation levels, our network with $n_{\text{sh,deg}} = 20$ is more robust under perturbation when compared to PointNet and PointNet++.

2) Evaluation of model against point cloud density: The point cloud density also plays an important role in the classification task. In this section, we downsample ModelNet40 to different point densities in the range of 1024 to 128 by using the farthest point sampling strategy (FPS) or random input droupout (DP). The downsampled point clouds are shown in Fig. 4a and the corresponding classification



(a) (b)

Fig. 4: (a) shows the downsampled point cloud and (b) illustrates the comparison results of different point densities in three rotation settings with FPS and DP.



(a) (b)

Fig. 5: (a) shows the partial point cloud with the deletion ratio ρ from 0.1 to 0.2 and (b) shows the results.

accuracy is illustrated in Fig. 4b. It is worth noting that our proposed network is very robust against point density changes in these three rotation setups, which decreases the classification accuracy from 88.6 % to 82.7 % by FPS and varies from 88.6 % to 75.4 % by DP. In comparison to DGCNN [18], the results vary from the 92.2 % to 79.2 % in z/z and from 33.5 % to 26.5 % in $z/\mathrm{SO}(3)$. Under the same point density, our model shows no significant change, which further verifies our model's robustness.

3) Evaluation of model against partial point cloud: In reality, we can only get a partial point cloud by using a single stationary camera. To evaluate the partial point cloud classification model, we train our model with a complete point cloud and test against a partial point cloud. The partial point cloud is obtained by first deleting the completed point cloud with a ratio ρ from 0.1 to 0.2 and then using iterative FPS (Fig. 5a). The results are illustrated in Fig. 5b. We compare our model with PointNet and DGCNN under three rotation settings. Training and testing data set are rotated with a PCA algorithm to reduce the impact of arbitrary rotation. From Fig. 5b, we can conclude that our model shows the best performance and far exceeds the other two classification models in all three experiments.

C. Ablation Studies

1) Analysis of architecture design: To evaluate our network architecture's effectiveness, we use PointNet as the baseline and connect our individually designed component to it. Note that we realign all data in this section with PCA (Section IV-A). In Table II, we can see that it shows a significant improvement when compared to the original PointNet in the setting of $z/\mathrm{SO}(3)$ and that the average accuracy rate has increased about 18 %. Furthermore, we analyzed the effectiveness of the RIMapping block by connecting it to PointNet. The results listed in Table II show that the accuracy in $z/\mathrm{SO}(3)$ and $\mathrm{SO}(3)/\mathrm{SO}(3)$ improved by 2.3 % and 3.7 %. Comparing our proposed network's worst performance shows that our PF Abstraction block helps in improving the final accuracy in all three settings. We also evaluated the effectiveness of the spherical harmonics energy descriptor. The results are listed in Table II. Without the spherical harmonics energy descriptor, the accuracy is worse when compared against our original design. However, it still shows better performance when compared to the PointNet-based network.

TABLE II: Effectiveness of designed network block.

Method	z/z	$z/\mathrm{SO}(3)$	$\mathrm{SO}(3)/\mathrm{SO}(3)$	mean
PointNet [16]	89.20	14.70	83.60	62.40
PCA+PointNet	80.90	80.90	80.80	80.84
RIMapping($n_{\mathrm{sh,deg}}$=20)+PointNet	82.00	83.20	84.50	83.23
ours(without SH)	85.40	85.70	86.20	85.77
ours($n_{\mathrm{sh,deg}}$=20)	88.40	88.60	88.50	88.50

TABLE III: Effectiveness of maximum degree $n_{\mathrm{sh,deg}}$.

$n_{\mathrm{sh,deg}}$	z/z	z/SO(3)	SO(3)/SO(3)
8	88.10	87.60	87.60
15	87.80	88.30	87.80
20	88.40	88.60	88.50
30	88.60	88.70	88.80

2) Effectiveness of maximum degree of spherical harmonics $n_{\mathrm{sh,deg}}$: The spherical harmonics descriptor is an essential aspect of our network. Based on the information loss, a higher degree of spherical harmonics leads to a smaller information loss. However, it will also increase the computational complexity to $\mathcal{O}(n^2)$. For evaluating the effectiveness of the $n_{\mathrm{sh,deg}}$, we vary the degree. The results are listed in Table III and it can be seen that the higher $n_{\mathrm{sh,deg}}$, the better the final classification accuracy.

VI. Conclusion

We presented a rotation-invariant point cloud-based neural network, which utilizes a global spherical harmonics feature and a local points feature to achieve rotation-invariant properties. Furthermore, a new neural network structure is designed, inspired by PointNet++, but with several adaptations such as PF graph and PF Transformation. The network is applied to 3D object classification, but can be extended to part segmentation and instance segmentation. Via several experiments, we have shown that our network can deal with arbitrary input orientations and achieve competitive performance compared to other state-of-the-art approaches on the ModelNet40 data set. Furthermore, our network shows robustness against point perturbations, point density, and partial point cloud.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] T. S. Cohen and M. Welling, "Transformation properties of learned visual representations," *Proceedings of the International Conference on Learning Representations*, 2015.

[3] T. S. Cohen, M. Geiger, J. Khler, and M. Welling, "Spherical CNNs," in *Proceedings of the International Conference on Learning Representations*, 2018.

[4] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 922–928.

[5] R. Kondor, Z. Lin, and S. Trivedi, "Clebsch-Gordan Nets: a fully fourier space spherical convolutional neural network," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 117–10 126.

[6] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Harmonic networks: Deep translation and rotation equivariance," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5028–5037.

[7] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proceedings of the Eurographics Symposiumon on Geometry Processing*, 2003, pp. 156–164.

[8] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Persistent point feature histograms for 3D point clouds," in *Proceedings of the International Conference on Intellligent Autonomous Systems*, 2008.

[9] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 3212–3217.

[10] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.

[11] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5648–5656.

[12] T. Yu, J. Meng, and J. Yuan, "Multi-view harmonized bilinear network for 3D object recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 186–194.

[13] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 945–953.

[14] A. Kanezaki, Y. Matsushita, and Y. Nishida, "RotationNet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5010–5019.

[15] G. Riegler, A. Osman Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3577–3586.

[16] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.

[17] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in Neural Information Processing Systems*, 2017, pp. 5099–5108.

[18] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Transactions on Graphics*, vol. 38, no. 5, pp. 1–12, 2019.

[19] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 2017–2025.

[20] C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis, "Learning SO(3) equivariant representations with spherical CNNs," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 52–68.

[21] N. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley, "Tensor field networks: Rotation-and translation-equivariant neural networks for 3D point clouds," *arXiv preprint arXiv:1802.08219*, 2018.

[22] M. Kazhdan and T. Funkhouser, "Harmonic 3D shape matching," in *ACM SIGGRAPH Conference Abstracts and Applications*, 2002, pp. 191–191.

[23] C. Chen, G. Li, R. Xu, T. Chen, M. Wang, and L. Lin, "ClusterNet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4994–5002.

[24] R. Gilmore, *Lie Groups, Physics, and Geometry: An Introduction for Physicists, Engineers and Chemists*. Cambridge University Press, 2008.

[25] V. Andrearczyk, V. Oreiller, J. Fageot, X. Montet, and A. Depeursinge, "Solid Spherical Energy (SSE) CNNs for efficient 3D medical image analysis," *Proceedings of the Irish Machine Vision and Image Processing Conference*, pp. 37–44, 2019.

[26] M. Reisert and H. Burkhardt, "Using irreducible group representations for invariant 3D shape description," in *Proceedings of the Joint Pattern Recognition Symposium*. Springer, 2006, pp. 132–141.

[27] C. R. Nortje, W. O. Ward, B. P. Neuman, and L. Bai, "Spherical harmonics for surface parametrisation and remeshing," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[28] R. B. Rusu, "Semantic 3D object maps for everyday manipulation in human living environments," *KI-Künstliche Intelligenz*, vol. 24, no. 4, pp. 345–348, 2010.

[29] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1–4.

[30] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1912–1920.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] J. Li, B. M. Chen, and G. Hee Lee, "SO-Net: Self-organizing network for point cloud analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9397–9406.