Equivariant Point Network for 3D Point Cloud Analysis

Haiwei Chen[1,2], Shichen Liu[1,2], Weikai Chen[2,3], Hao Li[4], Randall Hill[1,2]
[1]University of Southern California
[2]USC Institute for Creative Technologies
[3]Tencent Game AI Research Center
[4]Pinscreen
{chenh,lshichen,hill}@ict.usc.edu chenwk891@gmail.com hao@hao-li.com

Abstract

Features that are equivariant to a larger group of symmetries have been shown to be more discriminative and powerful in recent studies [5, 48, 6]. However, higher-order equivariant features often come with an exponentially-growing computational cost. Furthermore, it remains relatively less explored how rotation-equivariant features can be leveraged to tackle 3D shape alignment tasks. While many past approaches have been based on either non-equivariant or invariant descriptors to align 3D shapes, we argue that such tasks may benefit greatly from an equivariant framework. In this paper, we propose an effective and practical SE(3) (3D translation and rotation) equivariant network for point cloud analysis that addresses both problems. First, we present SE(3) separable point convolution, a novel framework that breaks down the 6D convolution into two separable convolutional operators alternatively performed in the 3D Euclidean and SO(3) spaces respectively. This significantly reduces the computational cost without compromising the performance. Second, we introduce an attention layer to effectively harness the expressiveness of the equivariant features. While jointly trained with the network, the attention layer implicitly derives the intrinsic local frame in the feature space and generates attention vectors that can be integrated with different alignment tasks. We evaluate our approach through extensive studies and visual interpretations. The empirical results demonstrate that our proposed model outperforms strong baselines in a variety of benchmarks. Code is available at https://github.com/nintendops/EPN_PointCloud.

1. Introduction

The success of 2D CNNs stems in large part from the ability of exploiting the translational symmetries via weight sharing and translation equivariance. Recent trends strive to duplicate this success to 3D domain in order to shed new light on the 3D learning tasks. With the 3D scanning technology being the mainstream manner of measuring the real world, point cloud arises naturally as one of the most prominent 3D representations. Yet, despite its simple and unified structure, it remains a nuisance to extend the CNN architecture to analyzing point clouds. In addition, the group of transformations in 3D data is more complex compared to 2D images, as 3D entities are often transformed by arbitrary 3D translations and 3D rotations when observed. Although group-invariant operators could render identical

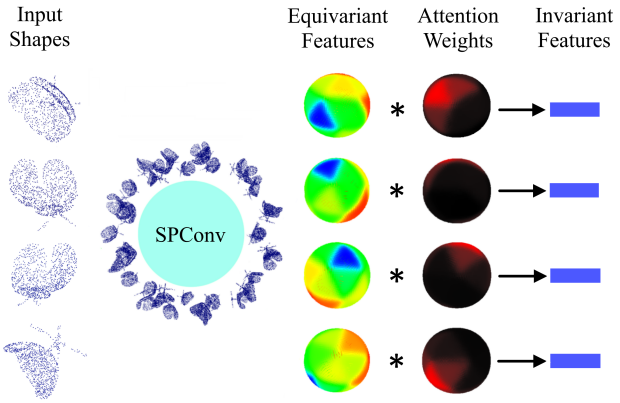


Figure 1: The core of our network is a convolution operator on point clouds, termed SE(3) separable point convolution (SPConv), that consumes features defined in the SE(3) space and outputs per-point features that are SE(3) equivariant. When the output feature is spatially pooled over the Euclidean space, it becomes SO(3) equivariant, as visualized above by projecting onto the spherical domain. Our method also supports a faithful conversion from the equivariant feature to its invariant counterpart by using a novel attentive fusing mechanism. Thereby, we offer a general framework that can generate equivariant or invariant point feature depending on the nature of downstream applications.

features even under different group transforms, it fails to distinguish distinct instances with internal symmetries (e.g. the counterparts of "6" and "9" in 3D scenarios regarding rotational symmetry). In contrast, equivariant features are much more expressive thanks to their ability to retain information about the input group transform on the feature maps throughout the neural layers. As a result, it could be very beneficial for point cloud features to be equivariant to the SE(3) group of transformations while being invariant to point permutations.

Despite the importance of deriving SE(3)-equivariant features for point clouds, progress in this regard remains highly sparse. The main obstacles arise in two aspects. First, the cost of computing convolutions between 6-dimensional functions over the entire SE(3) spaces is prohibitive especially in the presence of bulky 3D raw scans. Second, it remains challenging to fully harness the expressiveness of equivariant features without losing important structural information at a low computational cost. In particular, matching any two group-equivariant features is the prerequisite of many applications like correspondence computation, pose estimation, etc. One common practice is to compute the best relative group transformation that maximizes the similarity of the input features when the transformation is applied. This typically requires solving a PnP optimization which is quite costly considering the high dimensionality of the features. Another option is to fuse the equivariant features into invariant ones via pooling operation and directly compare the invariant features to obtain similarity. However, we argue that the naive pooling operations will inevitably discard useful features and damage the equivariant structure of the feature.

In this paper, we strive to address both of the problems by introducing an effective and practical framework for learning SE(3)-equivariant features of point clouds. In particular, inspired by the spirit of "going wider" in the Inception module [42], we first propose *SE(3) separable convolution*, a novel paradigm that breaks down the naive 6D convolution into two separable convolutional operators alternatively performed in the 3D Euclidean and SO(3) spaces. Due to the non-commutative and non-compact nature of SE(3) group, it is non-trivial to factorize SE(3) convolution into two separable sub-operators. We achieve this goal by first lifting the input points to the homogeneous space. We then take advantage of the finite rotation groups such as the icosahedral and aggregate spatially-convoluted features as functions on the rotation groups that are processed via group convolution. The proposed SE(3) separable convolution significantly reduces the computational cost of a SE(3) convolution and leads to practical solutions that can be deployed in the commodity hardware.

Second, we present an attention mechanism specially tailored for fusing SE(3)-equivariant features. We observe that while the commonly used pooling operations, such as max or mean pooling, work well in translation equivariant networks like 2D CNNs, they are not best suited for fusing equivariant features in SO(3) groups. This is mostly due to the highly sparse and non-linear structure of SO(3) features which poses additional challenges for max/mean pooling to maintain its unique pattern without losing too much information. We introduce *group attentive pooling* (GA pooling) to adaptively fuse rotation-equivariant features into their invariant counterparts. Trained together with the network, the GA pooling layer implicitly learns an intrinsic local frame of the feature space and generates attention weights to guide the pooling of rotation-equivariant features.

Third, compared to invariant features, equivariant features preserves, rather than discards, spatial structure and therefore can be seen as a more discriminative representation. It is for this reason that translational equivariance has been the premise for convolutional approaches for detection and instance segmentation [16]. Similarly, through the attention mechanism, the equivariant framework can be utilized for inferring 3D rotations. We demonstrate in the experiments that this structure significantly outperforms a non-equivariant framework in a shape alignment task.

We validate our proposed framework on a variety of tasks. Experimental results show that our approach consistently outperforms strong baselines. We also perform ablation analysis and qualitative visualization to evaluate the effectiveness of each algorithmic component.

2. Related Work

Human-engineered Point Features. There is rich literature [19] on investigating local geometric descriptors. One mainstream strategy resorts to local shape context encoded by geometry histogram and its variants [15, 45, 40, 25]. The other line of research strives to achieve rotation-invariance in designing the 3D feature. In particular, local reference frame (LRF) [46, 20, 45, 53] is widely employed to transform the local neighborhood of the point to a canonical space where the point features are analyzed and compared. In contrast, several approaches [40, 3, 39] leverage intrinsically invariant features, e.g. point pair features, without requiring LRF estimation. Though significant progress has been made, the hand-crafted features often fail to deal with noisy and incomplete data.

Learning-based Point Descriptor. The seminal work on handling irregular structure of point cloud places the main emphasis on permutation-invariant functions [36]. Later works proposes shift equivariant hierarchical architectures with localized filters to align with the regular grid CNNs [38, 30, 33]. Explicit convolution kernels have also received tremendous attention in recent years. In particular, various kernel forms have been proposed, including voxel bins [22], polynomial functions [52] or linear func-

tions [18]. Other works consider different representations of point clouds, noticeably image projection [11, 23, 32] and voxels [35, 2, 37, 51]. We point interested readers to [21] for a comprehensive survey on point cloud convolution.

Rotation invariant point descriptors have been an active research area due to its importance to correspondence matching. While the features extracted by most of the above approaches are permutation-invariant, very few of them can achieve rotation-invariance. The Perfect Match [17] incorporates a local reference frame (LRF) to extract rotation-invariant features from the voxelized point cloud. Similarly, [55] proposes a capsule network that consumes a point cloud along with the estimated LRF to disentangle shape and pose information. By only taking point pair as input, PPF-FoldNet [9] can learn rotation-invariant descriptors using folding-based encoding of point pair features. However, invariant features may be limited in expressiveness as spatial information is discarded a priori.

Learning Rotation-equivariant Features. Since CNNs are sensitive to rotations, a rapidly growing body of work focus on investigating rotation-equivariant variants. Starting from the 2D domain, various approaches have been proposed to achieve rotation equivariance by applying multiple oriented filters [34], performing a log-polar transform of the input [13], replacing filters with circular harmonics [50] or rotating the filters [31, 48]. Cohen and Welling later extend the domain of 2D CNNs from translation to finite groups [5] and further to arbitrary compact groups [8].

When it comes to the domain of 3D rotation, previous efforts can be divided into spectral and non-spectral methods. In the spectral branch, generalized Fourier transform for S^2 and SO(3) underlies designs for rotation equivariant CNN. We would like to highlight two seminal works [6], [12] that define convolution operators respectively by spherical (S^2) correlation, and SO(3) correlation with circularly symmetric kernels. The works most relevant to our setting are extensions of the two spectral paradigms to the 3D spatial domain. A number of works extend spherical CNNs to 3D voxels grids [49, 47, 12, 24]. Yet, the research work on exploring the potential on point clouds remains sparse, with the exception of a concurrent work Tensor field network (TFN) [44], which achieves SE(3) equivariance on irregular point clouds. While [44] shares with us in the use of tensor-field representation, their proposed filters are products of radial function and spherical harmonics. We instead focus on a non-spectral, computationally efficient separable framework.

Our work finds inspiration from the non-spectral group equivariant approaches that have seen recent progress, extending from mathematical framework derived in [5, 7]. Specifically, [7] provides a general framework for the practical implementation of convolution on discretized rotation group, with icosahedral convolution as an examplar. Discrete group convolution characterizes many recent works on images [14, 28], spherical signal [41], voxel grid [49] and point cloud [29]. Most of these works focus only on rotational equivariance. We are the first in this branch to provide a unified, hierarchical framework for point cloud convolution that is equivariant to the space of SE(3).

3. Method

Overview. In this section, we first start with the preliminaries of SE(3) convolutions. We will then provide the detailed mathematical formulation of our approach: (1) the SE(3) separable convolution; and (2) attention mechanism for the equivariant features. The Lie group SE(3) is the group of rigid body transformations in three-dimensions:

$$\text{SE(3)} = \{A | A = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}, R \in \text{SO(3)}, t \in \mathbb{R}^3\}.$$

SE(3) is homeomorphic to $\mathbb{R}^3 \times$ SO(3). Therefore, a function that is equivariant to SE(3) must be equivariant to both 3D translation $t \in \mathbb{R}^3$ and 3D rotation $g \in$ SO(3). Given a spatial point x and a rotation g, let us first define the continuous feature representation in SE(3) as a function $\mathcal{F}(x_i, g_j) : \mathbb{R}^3 \times \text{SO(3)} \rightarrow \mathbb{R}^D$. Equivariance to SE(3) is expressed as satisfying $\forall A \in \text{SE(3)}, A(\mathcal{F} * h)(x, g) = (A\mathcal{F} * h)(x, g)$. The SE(3) equivariant continuous convolutional operator can be defined as

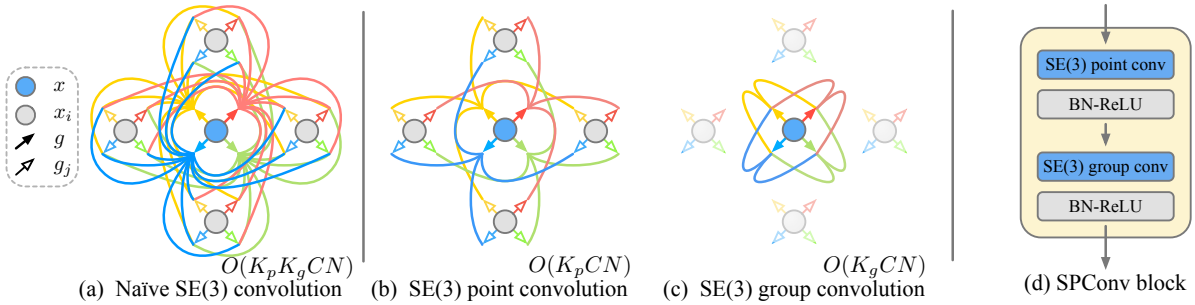
$$(\mathcal{F} * h)(x, g) = \int_{x_i \in \mathbb{R}^3} \int_{g_j \in \text{SO(3)}} \mathcal{F}(x_i, g_j) h(g^{-1}(x - x_i), g_j^{-1} g), \quad (1)$$

where h is a kernel $h(x, g) : \mathbb{R}^3 \times \text{SO(3)} \rightarrow \mathbb{R}^D$. The convolution is computed by translating and rotating the kernel and then computing a dot product with the input function $\mathcal{F}$. We prove that this convolution is equivariant to SE(3) in the supplementals.

Discretization. To discretize Eq. 1, we starts with discretizing the SE(3) space into a composition of a finite set of 3D spatial point $\mathcal{P} : \{x | x \in \mathbb{R}^3\}$ and a finite rotation group $G \subset$ SO(3). This leads to a discrete SE(3) feature mapping function $\mathcal{F}(x_i, g_j) : \mathcal{P} \times G \rightarrow \mathbb{R}^D$. The discrete convolutional operator in SE(3) is therefore:

$$(\mathcal{F} * h)(x, g) = \sum_{x_i \in \mathcal{P}} \sum_{g_j \in G} \mathcal{F}(x_i, g_j) h(g^{-1}(x - x_i), g_j^{-1} g). \quad (2)$$

We note that such discretization serves as a good approximation of the continuous formulation in Eq. 1, where the approximation error can be further mitigated by the rotation augmentation [1]. If we interpret $\mathcal{P}$ as a set of 3D displace-



(a) Naïve SE(3) convolution (b) SE(3) point convolution (c) SE(3) group convolution (d) SPConv block

Figure 2: Illustration of SPConv. Each arrow represents an element in the group and each edge represents a correlation needed to compute in the convolution operator. We propose to use two separable convolutions (b)(c) to achieve SE(3) equivariance. The computational cost is much lower than the naive 6D convolution (a). (d) shows the structure of a basic SPConv block.

ments, this leads to an equivalent definition:

$$(\mathcal{F} * h)(x, g) = \sum_{x_i \in \mathcal{P}} \sum_{g_j \in G} \mathcal{F}(g^{-1}(x - x_i), g_j^{-1} g) h(x_i, g_j) = \sum_{x'_i \in \mathcal{P}_g} \sum_{g_j \in G} \mathcal{F}(x - x'_i, g_j^{-1} g) h(g x'_i, g_j). \tag{3}$$

Without loss of generality, we assume the coordinate is expressed in the local frame of x and therefore $g^{-1}x = x$. In the second row of Eq. 3, the summation over the set $\mathcal{P}$ becomes a summation over a rotated set $\mathcal{P}_g : \{g^{-1}x | x \in \mathcal{P}\}$. When written this way, we can see that the kernel is parameterized by a set of translation offsets and rotation offsets under the reference frame given by g. We call the discrete set $\mathcal{P} \times G$ the domain of the kernel with a kernel size of $|\mathcal{P}| \times |G|$.

SE(3) Separable Convolution. A key issue with Eq. 3 is that the convolution is computed over a 6-dimensional space – a naive implementation would be computational prohibitive. Inspired by the core idea of separable convolution [4], we observe that the kernel h with a kernel size $|\mathcal{P}| \times |G|$ can be separated into two smaller kernels, denoting h_1 with a kernel size of $|\mathcal{P}| \times 1$ and h_2 with a kernel size of $1 \times |G|$. This divides the domain of the kernel to two smaller domains: $\mathcal{P} \times \{\mathrm{I}\}$ for h_1, and $\{\mathbf{0}\} \times G$ for h_2, where I is the identity matrix, and $\mathbf{0}$ is a zero displacement vector. From here, we are ready to separate Eq. 3 into two convolutions:

$$(\mathcal{F} * h_1)(x, g) = \sum_{x'_i \in \mathcal{P}_g} \mathcal{F}(x - x'_i, g) h_1(g x'_i, \mathrm{I}) \tag{4}$$

$$(\mathcal{F} * h_2)(x, g) = \sum_{g_j \in G} \mathcal{F}(x, g_j^{-1} g) h_2(\mathbf{0}, g_j) \tag{5}$$

We can see that h_1 is a kernel only varied by translation in the reference frame of g, and h_2 is a kernel only varied by the rotation g_j. In the following text, we simplify them to $h_1(g x'_i)$ and $h_2(g_j)$. The division here matches with the observation that the space SE(3) can be factorized into two spaces $\mathbb{R}^3$ and SO(3). Sequentially applying the two convolutions in Eq. (4-5) approximates the 6D convolution in Eq. 3 (Fig. 2(d)) while maintaining equivariance to SE(3) (proofs provided in the supplementary materials). The working principle here is similar to that of the Inception module [42] and its follow-up works [4], which have shown the promising property of separable convolutions in improving the network performance with reduced cost. We name the two consecutive convolutions as *SE(3) point convolution* and *SE(3) group convolution*, respectively, as shown in Fig. 2. We refer the combined convolutions as *SE(3) separable point convolution (SPConv)*. Formally, the original 6D convolution is approximated by: $\mathcal{F} * h \approx (\mathcal{F} * h_1) * h_2$.

A SE(3) equivariant convolutional network can be realized by consecutive blocks of SPConv. The network consumes the input $\mathcal{P}$ and produces a *SE(3) equivariant* feature for the point set. Since SPConv only takes functions defined on SE(3) as input, for each point in the input point set, we set $\mathcal{F}(x, g) = 1$ for each $g \in G$. In the following sections, we discuss in details the form of kernel and how it can be localized for each convolution module.

3.1. SE(3) point convolution

Our SE(3) point convolution layer aims at aggregating point spatial correlations locally under a rotation group element g. Let $\mathcal{N}_x = \{x_i \in \mathcal{P} \mid \|x - x_i\| \leqslant r\}$ be the set of neighbor points for point x, with a radius r, the SE(3) point convolution with localized kernel is:

$$(\mathcal{F} * h_1)(x, g) = \sum_{x'_i \in \mathcal{N}_{gx}} \mathcal{F}(x - x'_i, g) h_1(g x'_i), \tag{6}$$

where $\mathcal{N}_{gx} = \{g^{-1}(x - x_i) | x_i \in \mathcal{N}_x\}$ is the set of displacements to the neighbor points under a rotation g. h_1 is a kernel defined in a canonical neighbor space $\mathcal{B}_r^3$. Given that the convolution is computed as a spatial correlation under a rotation g, the form of the kernel can be naturally extended from any spatial kernel function. While our framework is general to support various spatial kernel definitions, we introduce two kernel formulations that are used in our implementation.

Explicit kernels. Given kernel size K, we can define a set of kernel points $\{\tilde{y}_k\}_K$ evenly distributed in $\mathcal{B}_r^3$. Each kernel point is associated with a kernel weight matrix $W_k \in \mathbb{R}^{D_{in} \times D_{out}}$, where D_{in} and D_{out} are the input and output channel, respectively. Let $\kappa(\cdot, \cdot)$ be the correlation function between two points, we have

$$h_1(x_i) = \sum_k^K \kappa(x_i, \tilde{y}_k) W_k. \tag{7}$$

The correlation function $\kappa(y, \tilde{y})$ can be either linear or Gaussian. For example, in the linear case described in [44], $\kappa(y, \tilde{y}) = \max(0, 1 - \frac{\|y - \tilde{y}\|}{\sigma})$, where σ adjusts the bandwidth.

Implicit kernels. The implicit formulation gives a function on point set that does not utilize parameterized kernels and is generally not considered a convolutional operation. Rather, spatial correlation is computed implicitly by concatenating the local frame coordinates of points to their corresponding features. In the SE(3) equivariant extension, the local coordinates are also composed by a corresponding rotation g. The implicit filter for the input signal $\mathcal{F}$ is:

$$\begin{aligned} h_1(\mathcal{F}(x, g)) &= \sum_{x_i \in \mathcal{N}_x} h_1(\mathcal{F}(x_i, g), g^{-1} x_i) \\ &= \sum_{x_i \in \mathcal{N}_x} \begin{bmatrix} \mathcal{F}(x_i, g) \\ g^{-1} x_i \end{bmatrix} W. \end{aligned} \tag{8}$$

We believe that other choices of kernel functions can be naturally extended from these two examples. In our implementation of the network, we use the explicit kernel formulation in all convolutional layers. The last layer before the output block of our network filters point features globally and therefore utilizes the implicit formulation, as it scales better to process a larger set of point features.

3.2. SE(3) group convolution

Given a discrete rotation group G, the SE(3) group convolution computes SO(3) correlation between the input signal and a kernel defined on the group domain.

We define a set of kernel rotation and their associate kernel weight matrices as $\mathcal{N}_g = \{g_j \in G\}_K$ and $\{W_j \in \mathbb{R}^{D_{in} \times D_{out}}\}_K$, with the kernel size $K = |\mathcal{N}_g|$. Thus the kernel is simply $h_2(g_j) = W_j$. Our SE(3) group convolution layer aggregates information from neighboring rotation signals within the group, which is given by

$$(\mathcal{F} * h_2)(x, g) = \sum_{g_j \in \mathcal{N}_g} \mathcal{F}(x, g_j^{-1} g) h_2(g_j). \tag{9}$$

In our implementation, the icosahedron group can be used as the discrete rotation group. The K neighbor rotations are a subset of the group that is smallest in the corresponding angle. The computation can be accelerated by pre-computing the permutation index and only performing constant-time query with an index layer at run time.

Complexity analysis. As illustrated in Fig. 2, by combining the two equivariance-preserving convolutions, we can achieve a similar effect with Eq. 2 at a significantly lower computational cost. In particular, suppose we divide the original number of kernels K into K_p and K_g, the number of kernels in the point and group convolution; $C = C_i C_o$ where C_i and C_o are the number of input and output channels, $N = N_p N_a$ is the product of the number of points and the number of SO(3) element in a rotation group. The naive 6D convolution requires a computational complexity of $O(K_p K_g C N)$. In contrast, the complexity of our approach is reduced to $O((K_p + K_g) C N)$, which could achieve orders-lower complexity compared to the naive solution especially with large K_p and K_g.

3.3. Shape Matching with Attention mechanism

In this section, we demonstrate how attention mechanism can be utilized to harness the power of equivariant feature. Given spatially pooled features that are equivariant to SO(3): $\mathcal{F}(g) : G \to \mathbb{R}^D$, we define a rotation-based attention $A : G \to \mathbb{R}$, $A(g) = \{a_g | \sum_{g \in G} a_g = 1\}$.

SO(3) Detection. Suppose a task requires the network to predict the pose $R \in$ SO(3) of an input shape. When the attention weight is used as a probability score, the equivariant network turns the pose estimation task into a SO(3) detection task, which is analogous to bounding box detection. Intuitively, each element from the discrete rotation group can be interpreted as an anchor. A two-branch network is used to classify whether the anchor is the "dominant rotation". Every anchor regresses a small rotational offset from its corresponding rotation. The multi-task loss for rotational regression is then given by:

$$\mathcal{L}(a, u, R, R^u) = \mathcal{L}_{cls}(a, u) + \lambda [u = 1] \mathcal{L}_2(R^u R^T) \tag{10}$$

where $a = \{a_g | g \in G\}$ are the predicted probabilities and R are the predicted relative rotations. $u = \{u_g | g \in G\}$ is the ground-truth label with $u_g = 1$ if g is the nearest rotation to the target ground truth rotation R_{GT}. $R^u = \{R_g^u | \forall g \in G, R_g^u g = R_{GT}\}$ is the ground truth relative rotation.

Group Attentive Pooling. Global pooling layers are integrated as part of the network for spatial reduction of the representation. As many common tasks, such as classification, benefit from rotation invariance of the learned feature, global pooling is utilized by most rotation-equivariant architectures to aggregate information into an invariant representation. To integrate attention mechanism with global pooling, we propose *group attentive pooling (GA pooling)*, which is given by

$$\mathcal{F}_{inv} = \frac{\sum_g \exp(a_g / T) \mathcal{F}_G(g)}{\sum_g \exp(a_g / T)}, \tag{11}$$

where $\mathcal{F}_G(g)$ and a_g are the input rotation-equivariant feature and attention weight on rotation g. T is a temperature score to control the sharpness of the function response. As visualized in Fig. 1, the output feature is invariant given a rotated input point cloud. The confidence weight a can be learned by minimizing the loss $\mathcal{L} = \mathcal{L}_{task} + \lambda\mathcal{L}_{sa}$, where $\mathcal{L}_{task}$ is a task-specific loss (e.g. cross-entropy loss for classification and triplet loss for correspondence matching); $\mathcal{L}_{sa}$ is a optional cross-entropy loss that encourages the network to learn the canonical axis from the candidate orientations when ground truth canonical pose is available for supervision.

3.4. Implementation Details

The core element of our network is the SPConv block as shown in Fig. 2(d). It consists of one SE(3) point convolution and one SE(3) group convolution operator, with a batch normalization and a leaky ReLU activation inserted in between and after. We employ a 5-layer hierarchical convolutional network. Each layer contains two SPConv blocks, with the first one being strided by a factor of 2. The network outputs spatially pooled features that are equivariant to the rotation group G. It can be then pooled into an invariant feature through a GA pooling layer. For the classification network, the feature is fed into a fully connected layer and a *softmax* layer. For the task of metric learning, the feature is processed with an L2 normalization. We provide detailed network parameters and downsampling strategy in the supplemental materials.

4. Experiments

We hypothesize that our approach is most suitable for tasks where the objects of interest are rotated arbitrarily. To this end, we evaluate our approach on two rotation-related datasets: the rotated Modelnet40 dataset [51] and the 3DMatch dataset [54]. To ensure a fair comparison to previous works, in all experiments, we use the implementation provided by the authors or the reported statistics if no source code is available. We provide the training details of the experiments in the supplemental materials.

4.1. Experiments on Rotated ModelNet40

Dataset. The official aligned Modelnet40 dataset provides a setting where canonical poses are known, and therefore it allows us to evaluate the effectiveness of pose supervision. We create the rotated ModelNet40 dataset based on the train/test split of the aligned ModelNet40 dataset [51]. We mainly focus on a more challenging "rotated" setting where each object is randomly rotated. For each object, we randomly subsample 1,024 points from the surface of the 3D object and perform random rotation augmentation before feeding it into the network.

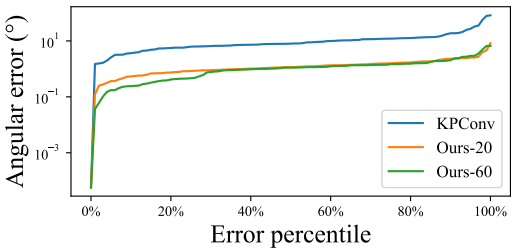


Figure 3: Percentile of errors comparing KPConv [43] and two equivariant models (Ours-N) varied in number of SO(3) elements.

	Mean (°)	Median (°)	Max(°)
KPConv [43]	11.46	8.06	82.32
Ours-20	1.36	1.16	8.30
Ours-60	**1.25**	**1.11**	**6.63**

Table 1: Angular errors in point cloud pose estimation.

Pose Estimation. The pose estimation task predicts the rotation $\mathcal{R} \in$ SO(3) that aligns a rotated input shape to its canonical pose. To avoid ambiguities indued by rotationally symmetric objects, we only use the airplane category from the dataset. We train the network with N=1252 airplane point clouds and test it with N=101 held-out point cloud, each augmented with random rotations. The evaluation compares equivariant models with KPConv [43], a network that has similar kernel function to our implementation of point convolution, while not equivariant to 3D rotation. The equivariant models (Ours-N) are varied by the size of rotation group (N), similar to the setting in [14], and use the multitask detection loss described in Sec. 3.3. KPConv directly regresses the output rotation. Each model is trained for 80k iterations. The regressors in all models produce a rotation in the quaternion representation. We evaluate the performance by measuring angular errors between the predicted rotations and the ground-truth rotations. Tab. 1 shows the mean, median and max angular errors in each setting, and Fig. 3 plots the error percentile curves. As shown in the results, the equivariant networks significantly outperform the baseline network, with Ours-60 having the lowest errors. The equivariant networks also perform significantly more stable (max angular errors are kept within 9 degrees), while KPConv could produce unstable results for a certain inputs. This experiment showcases that a hierarchical rotation model can be much more effective in task that requires direct prediction of 3D rotation.

Classification and Retrieval. The classification and retrieval tasks on Modelnet40 follow evaluation metric from [51]. In addition, our network is trained with GA pooling and pose supervision introduced in Sec. 3.3. In Tab. 2, we show the results comparing with the state-of-the-art methods in the setting where models are both trained and tested with rotation augmentation. We categorize the base-

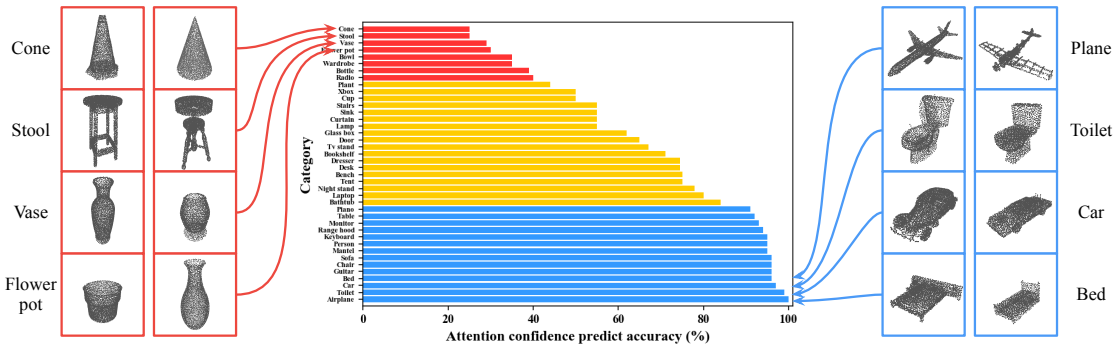


Figure 4: Classification accuracy based on the attention confidence for each object category. The attention layer is trained on rotated dataset to learn a canonical orientation for the given object.

Representation	Methods	Acc (%)	Retrieval (mAP)
3D Surface	RotationNet [26]	80.0	74.2
	Sph. CNN [12]	86.9	-
Point cloud	QENet [55]	74.4	-
	PointNet [36]	83.6	-
	PointNet++ [38]	85.0	70.3
	DGCNN [36]	81.1	-
	PointCNN [38]	84.5	-
	KPConv [43]	86.7	77.5
	Ours	**88.3**	**79.7**

Table 2: Results on shape classification and retrieval on randomly rotated objects of ModelNet40.

line approaches based on the input 3D representations: 3D surface and point cloud.

In the classification and retrieval task, our models also achieve the best performance, as shown in Tab. 2. This indicates that our proposed framework can learn more effective and discriminative features even in the challenging cases that all the objects are randomly rotated.

Ablation Analysis. We further conduct an ablation study to validate the effectiveness of each algorithmic component. In particular, we experiment with five variants of our model by altering key designs in our network under the same architecture as shown in Tab. 3. By using the supervised attentive pooling, we can improve the classification accuracy with the same number of parameters compared to the max and mean pooling. However, the unsupervised attentive pooling does not outperform max pooling. This may be partly due to the difficulty of learning canonical pose in an unsupervised manner. In addition, only using point convolution will lead to a decline in performance, indicating the effectiveness of group convolution.

How well does the attention layer learn? It is possible that the performance of GA pooling in distinguishing canonical poses could be compromised by the rotational symmetry of the object. If a shape is circularly symmetric, and the canonical poses prescribed by the rotational label is aligned with an axis of symmetry, the attention layer would naturally fail to provide a deterministic prediction. We summarize the classification accuracy based on the attention confidence for each category of ModelNet objects, as shown in Fig. 4. The results indeed support our intuition: the attention layer is ambiguous on objects with circular symmetry (e.g. cone and flower pot) and very confident on categories that have distinctive canonical orientation. On one hand, this shows that when the object of interest is asymmetric in rotation, the GA pooling does help improve classification performance by establishing a local reference frame. On the other hand, the GA pooling only fails at symmetric object that benefits relatively less from a equivariant representation. In the extreme case, the attention layer could be reduced to an average pooling.

conv	global pool	Loss	Acc (%)
Separable Conv	Attentive	$\mathcal{L}_{cls} + \mathcal{L}_{sa}$	**88.3**
Separable Conv	Attentive	$\mathcal{L}_{cls}$	87.7
Separable Conv	Max	$\mathcal{L}_{cls}$	87.7
Separable Conv	Mean	$\mathcal{L}_{cls}$	87.4
Point Conv	Attentive	$\mathcal{L}_{cls} + \mathcal{L}_{sa}$	86.1

Table 3: Results of ablation studies on ModelNet40 dataset. The *conv* column denotes the configuration of convolution layers. The *global pool* column denotes the type of global pooling method. Loss configuration follows notation from Sec. 3.3.

4.2. Shape Alignment on 3DMatch

Dataset. The 3DMatch dataset is a real-scan dataset consisting of 433 overlapping fragments from 8 indoor scenes for evaluation, and RGB-D data set from 62 indoor scenes for training. The pose of each fragment is determined by the camera angle during capturing, and two fragments at most overlap partially. Evaluating our model on this dataset is meaningful as shape registration in such setting would benefit from descriptors that are invariant to rigid camera motion. Each test fragment is a densely sampled point cloud with 150,000 to 600,000 points. To be consistent with our baselines, we use an evaluation metric based on the aver-

	SHOT[45]	3DMatch[54]	CGF[27]	PPFNet[10]	PPFF[9]	3DSNet[17]	Li[32]	Li[32]♭	Ours
Kitchen	74.3	58.3	60.3	89.7	78.7	97.5	92.1	**99.4**	99.0
Home 1	80.1	72.4	71.1	55.8	76.3	96.2	91.0	98.7	**99.4**
Home 2	70.7	61.5	56.7	59.1	61.5	93.2	85.6	94.7	**96.2**
Hotel 1	77.4	54.9	57.1	58.0	68.1	97.4	95.1	**99.6**	**99.6**
Hotel 2	72.1	48.1	53.8	57.7	71.2	92.8	91.3	**100.0**	97.1
Hotel 3	85.2	61.1	83.3	61.1	94.4	98.2	96.3	**100.0**	**100.0**
Study	64.0	51.7	37.7	53.4	62.0	95.0	91.8	95.5	**96.2**
MIT Lab	62.3	50.7	45.5	63.6	62.3	**94.1**	84.4	92.2	93.5
Average	73.3	57.3	58.2	62.3	71.8	95.6	91.0	97.5	**97.6**

Table 4: Comparisons of average recall of keypoint correspondences on 3DMatch. Li [32]♭ denotes results tested with point normal information provided by the authors. All other results are tested on the official 3DMatch evaluation set without point normals.

age recall of keypoints correspondence without performing RANSAC, following [10]. We also follow previous works [9, 10, 17] to set the matching threshold $\tau_1 = 0.1m$ and the inlier ratio $\tau_2 = 0.05$.

Comparison with baselines. We designed a Siamese network for this task and trained our model with the batch-hard triplet loss proposed in [17]. The input to the network is 1024-point patches extracted locally from keypoints in a fragment. The output is 64-dim invariant descriptors. Since a canonical ground truth pose is not known in this setting, the attentive pooling module in our model is trained in an unsupervised manner. Our results are shown in Tab. 4. To provide a comprehensive comparison, we select the state-of-the-art baselines using a variety of approaches: 1) convolutional network without rotational invariance, e.g. [54, 10] 2) handcrafted invariant features w/ and w/o deep learning, e.g. [27, 45, 9], 3) features learned from LRF aligned input [17], and 4) multi-view network [32]. We report the 64-dim results of [17] to match the feature dimension of our model. Since the official 3DMatch test dataset does not contain point normal information, we report two results of [32]: a result of their model trained and tested without normal information (Li [32] in Tab. 4) and one that is trained and tested with the authors' provided point normals (Li [32]♭ in Tab. 4). We evaluate our model with the interest points provided by the authors of the dataset, which is consistent with the reported results of our baselines. Overall, our model outperforms all of the baselines in average recall, without the need to precompute an invariant representation or a local reference frame. Compare to some baselines (e.g. [9, 32]) that requires dense point input, our model can learn discriminative features from very sparsely sampled sets of 1024 point. Our result is also better than the state-of-the-art method [32], even without needing normal information as input. In the official setting where point normal information is not available, the performance of our model marks a great leap forward.

Qualitative analysis. We provide a T-SNE visualization of the features learned by our network in Fig. 5. As different features are labeled with distinct colors, we can observe that the features learned by our network can robustly generate correct geometry correspondences even when the point cloud is incomplete, partially aligned, or significantly rotated. For instance, in the third column, the bottom scene is only partially aligned with the top one and is viewed at an entirely different angle, our network can still reliably label the corresponded points with similar features.

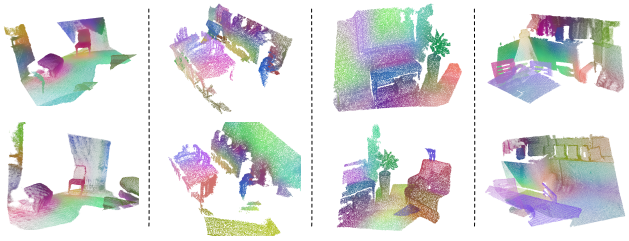


Figure 5: T-SNE visualization of features learned by our network. Each column contains a pair of fragments from the same scene. Regions in correspondence are automatically labeled with similar features.

5. Conclusions and Discussions

We have presented a novel framework that efficiently computes and leverages SE(3)-equivariant features for point cloud analysis. First, we introduce a novel formulation named SE(3) separable convolution that factorizes the naive SE(3) convolution into two concatenated operators performed in two subspaces. Second, we propose the incorporation of attention mechanism that can appreciate and maintain the expressiveness of SE(3)-equivariant features, which provides a novel way for 3D alignment tasks and can be used as a pooling layer that fuses the equivariant features into their more ready-to-use invariant counterparts. Such paradigm has led to leaps of performance in a variety of challenging tasks. Our approach is one of the earliest attempts of investigating SE(3)-equivariant features for point cloud analysis. We believe there are still ample opportunities for more efficient methods and extension of the equivariant features to a broader range of applications.

Acknowledgements

This research was sponsored by the Army Research Office and was accomplished under Cooperative Agreement Number W911NF-20-2-0053, and sponsored by the U.S. Army Research Laboratory (ARL) under contract number W911NF-14-D-0005, the CONIX Research Center, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA and in part by the ONR YIP grant N00014-17-S-FO14. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation.

References

[1] Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018. 3

[2] Yizhak Ben-Shabat, Michael Lindenbaum, and Anath Fischer. 3dmfv: Three-dimensional point cloud classification in real-time using convolutional neural networks. *IEEE Robotics and Automation Letters*, 3(4):3145–3152, 2018. 3

[3] Tolga Birdal and Slobodan Ilic. Point pair features based object detection and pose estimation revisited. In *2015 International Conference on 3D Vision*, pages 527–535. IEEE, 2015. 2

[4] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 4

[5] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999, 2016. 1, 3

[6] Taco S. Cohen, Mario Geiger, Jonas Koehler, and Max Welling. Spherical cnns. 2018. Proceedings of the 6th International Conference on Learning Representations (ICLR), 2018. 1, 3

[7] Taco S Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral cnn. *arXiv preprint arXiv:1902.04615*, 2019. 3

[8] Taco S Cohen and Max Welling. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016. 3

[9] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 602–618, 2018. 3, 8

[10] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 195–205, 2018. 8

[11] Gil Elbaz, Tamar Avraham, and Anath Fischer. 3d point cloud registration for localization using a deep neural network auto-encoder. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4631–4640, 2017. 3

[12] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018. 3, 7

[13] Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis. Polar transformer networks. *arXiv preprint arXiv:1709.01889*, 2017. 3

[14] Carlos Esteves, Yinshuang Xu, Christine Allen-Blanchette, and Kostas Daniilidis. Equivariant multi-view networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1568–1577, 2019. 3, 6

[15] Andrea Frome, Daniel Huber, Ravi Kolluri, Thomas Bülow, and Jitendra Malik. Recognizing objects in range data using regional point descriptors. In *European conference on computer vision*, pages 224–237. Springer, 2004. 2

[16] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 2

[17] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5545–5554, 2019. 3, 8

[18] Fabian Groh, Patrick Wieschollek, and Hendrik PA Lensch. Flex-convolution. In *Asian Conference on Computer Vision*, pages 105–122. Springer, 2018. 3

[19] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, Jianwei Wan, and Ngai Ming Kwok. A comprehensive performance evaluation of 3d local feature descriptors. *International Journal of Computer Vision*, 116(1):66–89, 2016. 2

[20] Yulan Guo, Ferdous Sohel, Mohammed Bennamoun, Min Lu, and Jianwei Wan. Rotational projection statistics for 3d local surface description and object recognition. *International journal of computer vision*, 105(1):63–86, 2013. 2

[21] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 3

[22] Binh-Son Hua, Minh-Khoi Tran, and Sai-Kit Yeung. Pointwise convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 984–993, 2018. 2

[23] Haibin Huang, Evangelos Kalogerakis, Siddhartha Chaudhuri, Duygu Ceylan, Vladimir G Kim, and Ersin Yumer. Learning local shape descriptors from part correspondences with multiview convolutional networks. *ACM Transactions on Graphics (TOG)*, 37(1):6, 2018. 3

[24] Chiyu Jiang, Jingwei Huang, Karthik Kashinath, Philip Marcus, Matthias Niessner, et al. Spherical cnns on unstructured grids. *arXiv preprint arXiv:1901.02039*, 2019. 3

[25] Andrew E. Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on pattern analysis and machine intelligence*, 21(5):433–449, 1999. 2

[26] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi 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*, pages 5010–5019, 2018. 7

[27] Marc Khoury, Qian-Yi Zhou, and Vladlen Koltun. Learning compact geometric features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 153–161, 2017. 8

[28] Jan Eric Lenssen, Matthias Fey, and Pascal Libuschewski. Group equivariant capsule networks. In *Advances in Neural Information Processing Systems*, pages 8844–8853, 2018. 3

[29] Jiaxin Li, Yingcai Bi, and Gim Hee Lee. Discrete rotation equivariance for point cloud recognition. *arXiv preprint arXiv:1904.00319*, 2019. 3

[30] Jiaxin Li, Ben M Chen, and Gim Hee Lee. So-net: Self-organizing network for point cloud analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9397–9406, 2018. 2

[31] Junying Li, Zichen Yang, Haifeng Liu, and Deng Cai. Deep rotation equivariant network. *Neurocomputing*, 290:26–33, 2018. 3

[32] Lei Li, Siyu Zhu, Hongbo Fu, Ping Tan, and Chiew-Lan Tai. End-to-end learning local multi-view descriptors for 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1919–1928, 2020. 3, 8

[33] Xinhai Liu, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Point2sequence: Learning the shape representation of 3d point clouds with an attention-based sequence to sequence network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8778–8785, 2019. 2

[34] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5048–5057, 2017. 3

[35] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928. IEEE, 2015. 3

[36] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2, 7

[37] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas 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*, pages 5648–5656, 2016. 3

[38] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017. 2, 7

[39] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217. IEEE, 2009. 2

[40] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3384–3391. IEEE, 2008. 2

[41] Riccardo Spezialetti, Samuele Salti, and Luigi Di Stefano. Learning an effective equivariant 3d descriptor without supervision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6401–6410, 2019. 3

[42] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2, 4

[43] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. *arXiv preprint arXiv:1904.08889*, 2019. 6, 7

[44] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 3, 5

[45] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique shape context for 3d data description. In *Proceedings of the ACM workshop on 3D object retrieval*, pages 57–62. ACM, 2010. 2, 8

[46] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *European conference on computer vision*, pages 356–369. Springer, 2010. 2

[47] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. In *Advances in Neural Information Processing Systems*, pages 10381–10392, 2018. 3

[48] Maurice Weiler, Fred A Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 849–858, 2018. 1, 3

[49] Daniel Worrall and Gabriel Brostow. Cubenet: Equivariance to 3d rotation and translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 567–584, 2018. 3

[50] Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5028–5037, 2017. 3

[51] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1912–1920, 2015. 3, 6

[52] Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 87–102, 2018. 2

[53] Jiaqi Yang, Qian Zhang, Yang Xiao, and Zhiguo Cao. Toldi: An effective and robust approach for 3d local shape description. *Pattern Recognition*, 65:175–187, 2017. 2

[54] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017. 6, 8

[55] Yongheng Zhao, Tolga Birdal, Jan Eric Lenssen, Emanuele Menegatti, Leonidas Guibas, and Federico Tombari. Quaternion equivariant capsule networks for 3d point clouds. *arXiv preprint arXiv:1912.12098*, 2019. 3, 7