S3I-POINTHOP: SO(3)-INVARIANT POINTHOP FOR 3D POINT CLOUD CLASSIFICATION

Pranav Kadam [1], Hardik Prajapati [1], Min Zhang [1], Jintang Xue [1], Shan Liu [2], C.-C. Jay Kuo [1]

University of Southern California, Los Angeles, California, USA[1]
Tencent Media Lab, Palo Alto, California, USA[2]

ABSTRACT

Many point cloud classification methods are developed under the assumption that all point clouds in the dataset are well aligned with the canonical axes so that the 3D Cartesian point coordinates can be employed to learn features. When input point clouds are not aligned, the classification performance drops significantly. In this work, we focus on a mathematically transparent point cloud classification method called PointHop, analyze its reason for failure due to pose variations, and solve the problem by replacing its pose dependent modules with rotation invariant counterparts. The proposed method is named SO(3)-Invariant PointHop (or S3I-PointHop in short). We also significantly simplify the PointHop pipeline using only one single hop along with multiple spatial aggregation techniques. The idea of exploiting more spatial information is novel. Experiments on the ModelNet40 dataset demonstrate the superiority of S3I-PointHop over traditional PointHop-like methods.

Index Terms— point cloud classification, rotation invariance, PointHop

1. INTRODUCTION

Due to numerous applications in autonomous vehicles and robotics perception, immersive media processing, 3D graphics, etc., 3D Point Clouds have emerged to be a popular form of representation for 3D vision tasks. Research and development on point cloud data processing has attracted a lot of attention. Recent trends show a heavy inclination towards development of learning-based methods for point clouds.

One of the primary tasks in point cloud understanding is object classification. The task is to assign a category label to a 3D point cloud object scan. The unordered nature of 3D point cloud demands methods to be invariant to $N!$ point permutations for a scan of N points. It was demonstrated in the pioneering work called PointNet [1] that permutation invariance can be achieved using a symmetric function such as the maximum value of point feature responses. Besides permutations, invariance with respect to rotations is desirable in many applications such as 3D registration. In particular, point cloud features are invariant with any 3D transformation in the SO(3) group; namely, the group of 3×3 orthogonal matrices representing rotations in 3D.

Achieving rotation invariance can guarantee that point clouds expressed in different orientations are regarded the same and, thereby, the classification result is unaffected by the pose. State-of-the-art methods do not account for rotations, and they perform poorly in classifying different rotated instances of the same object. In most cases, objects are aligned in a canonical pose before being fed into a learner. Several approaches have been proposed to deal with this problem. First is data augmentation, where different rotated instances of the same object are presented to a learner. Then, the learner implicitly learns to reduce the error metric in classifying similar objects with different poses. This approach leads to an increase in the computation cost and system complexity. Yet, there is no guarantee in rotational invariance. A more elegant way is to design point cloud representations that are invariant to rotations. Thus, point cloud objects expressed in different orientations are indistinguishable to classifiers. Another class of methods are based on SO(3) equivariant networks, where invariance is obtained as a byproduct of the equivariant point cloud features.

Point cloud classification based on the green learning principle was first introduced in PointHop [2]. The work is characterized by its mathematical transparency and lightweight nature. The methodology has been successfully applied to point cloud segmentation [3, 4] and registration [5]. For point cloud classification, both PointHop and its follow-up work PointHop++ [6] assume that the objects are pre-aligned. Due to this assumption, these methods fail when classifying objects with different poses.

In this work, we propose an SO(3) invariance member for the PointHop family, and name it S3I-PointHop. This is achieved through the derivation of invariant representations by leveraging principal components, rotation invariant local/global features, and point-based eigen features. Our work has two main contributions. First, the pose dependent octant partitioning operation in PointHop is replaced by an ensemble of three rotation invariant representations to guarantee SO(3) invariance. Second, by exploiting the rich spatial information, we simplify multi-hop learning in PointHop to one-hop learning in S3I-PointHop. Specifically, two novel aggregation schemes (i.e., conical and spherical aggregations in local regions) are proposed, which makes one-hop learning possible.

The rest of this paper is organized as follows. Related material is reviewed in Sec. 2. The S3I-PointHop method is proposed in Sec. 3. Experimental results are presented in Sec. 4. Finally, concluding remarks are given in Sec. 5.

2. RELATED WORK

2.1. Green Point Cloud Learning

Green learning (GL) [7] is a data-driven learning methodology. It uses training data statistics to derive representations without labels. The learning process utilizes the Saab transform [8] or the channel-wise Saab transform [9]. GL is a radical departure from neural

The authors acknowledge the gift support from the Tencent Media Lab as well as the Center for Advanced Research Computing (CARC) at the University of Southern California for providing computing resources that have contributed to the research results reported within this publication. URL: https://carc.usc.edu.

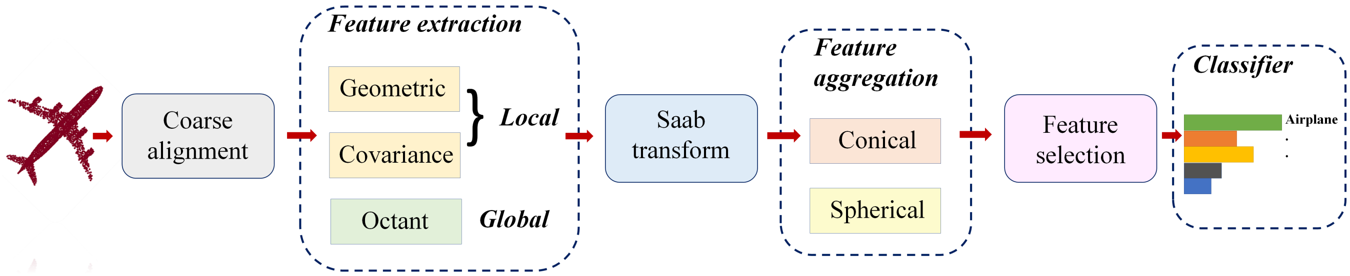

Fig. 1. An overview of the proposed S3I-PointHop method: 1) an input point cloud scan is approximately aligned with the principal axes, 2) local and global point features are extracted and concatenated followed by the Saab transform, 3) point features are aggregated from different conical and spherical volumes, 4) discriminant features are selected using DFT and a linear classifier is used to predict the object class.

networks. It has achieved impressive results for point cloud data processing. For example, PointHop and PointHop++ offer competitive performance in classification of aligned point clouds. They both have three main modules: 1) hierarchical attribute construction based on the distribution of neighboring points in the 3D space and attribute dimensionality reduction using the Saab/channel-wise Saab transform, 2) feature aggregation, and 3) classification. The capability of GL has been demonstrated by follow-ups, including point cloud segmentation [3, 4], registration [5, 10], odometry [11], and pose estimation [12]. GL-based point cloud processing techniques are summarized in [13]. Similar to early point cloud classification methods, PointHop and PointHop++ fail to classify objects of arbitrary poses.

2.2. Rotation Invariant Networks

Early pioneering deep networks for point cloud processing tasks such as PointNet [1], PointNet++ [14], DGCNN [15] and PointCNN [16] are susceptible to point cloud rotations. Designing rotation invariant networks has been popular for 3D registration when global alignment is needed. Methods such as PPFNet [17] and PPF-FoldNet [18] achieve partial and full invariance to 3D transformations, respectively. The idea behind any rotation invariant method is to design a representation that is free from the pose information. This is done by exploiting properties of 3D transformations such as preservation of distances, relative angles, and principal components. Global and local rotation invariant features for classification were proposed in [19], which form a basis of our method. Ambiguities associated with global PCA alignment were analyzed and a disambiguation network was proposed in [20]. Another approach is the design of equivariant neural networks that achieve invariance via certain pooling operations. SO(3)- and SE(3)-equivariant convolutions make networks equivariant to the 3D rotation and 3D roto-translation groups, respectively. Exemplary work includes the Vector Neurons [21] for classification and segmentation, results in [22, 23] for category-level pose estimation.

3. PROPOSED S3I-POINTHOP METHOD

The S3I-PointHop method assigns a class label to a point cloud scan, X, whose points are expressed in an arbitrary coordinate system. Its block diagram is shown in Fig. 1. It comprises of object coarse alignment, feature extraction, dimensionality reduction, feature aggregation, feature selection and classification steps as detailed below.

3.1. Pose Dependency in PointHop

The first step in PointHop is to construct a 24-dimensional local descriptor for every point based on the distribution of 3D coordinates of the nearest neighbors of that point. 3D rotations are distance preserving transforms and, hence, the distance between any two points remains the same before and after rotation. As a consequence, the nearest neighbors of points are unaffected by the object pose. However, the use of 3D coordinates makes PointHop sensitive to rotations since the 3D Cartesian coordinates of every point change with rotation. Furthermore, the 3D space surrounding the current point is partitioned into 8 octants using the standard coordinate axes. The coordinate axes change under different orientations of the point cloud scan. We align an object with its three principal axes. The PCA alignment only offers a coarse alignment, and it comes with several ambiguities as pointed out in [20]. Furthermore, object asymmetries may disturb the alignment since PCA does not contain semantic information. Yet, fine alignment is not demanded. Here, we develop rotation invariant features based on PCA aligned objects.

3.2. Feature Extraction

Local and global information fusion is effective in feature learning for point cloud classification [15]. To boost the performance of S3I-PointHop, three complementary features are ensembled. The first feature set contains the omni-directional octant features of points in the 3D space as introduced in PointHop. That is, the 3D space is partitioned into eight octants centered at each point as the origin. The mean of 3D coordinates of points in each octant then constitute the 24D octant feature. The second feature set is composed by eigen features [24] obtained from the covariance analysis of the neighborhood of a point. They are functions of the three eigen values derived from the Singular Value Decomposition (SVD) of the local covariance matrix. The 8 eigen features comprise of linearity, planarity, anisotropy, sphericity, omnivariance, verticality, surface variation and eigen entropy. They represent the surface information in the local neighborhood. The third feature set is formed by geometric features derived from distances and angles in local neighborhoods as proposed in [19]. For simplicity, we replace the geometric median in [19] with the mean of the neighboring coordinates. The 12D feature representation is found using the K nearest neighbors, leading to a pointwise $12 \times K$ matrix. While a small network is trained in [19] to aggregate these features into a single vector, we perform a channel-wise max, mean and l_2-norm pooling to yield a 36D vector of local geometric feature. The octant, covariance and geometric features are concatenated together to build a 68D $(24 + 8 + 36 = 68)$

feature vector. After that, the Saab transform is performed for dimension reduction.

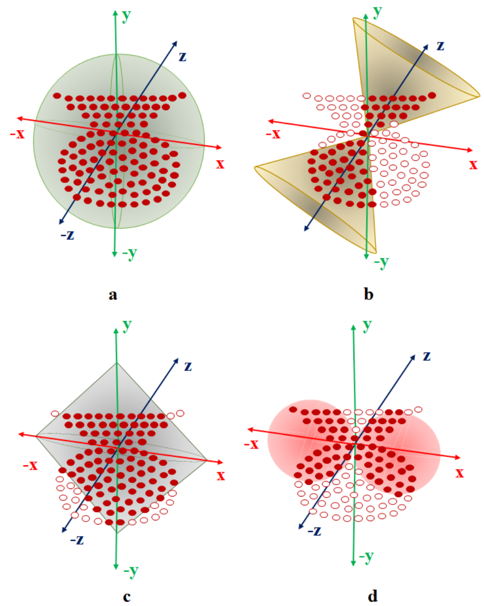

Fig. 2. Illustration of conical and spherical aggregation. The conventional "global pooling" is shown in (a), where features of all points are aggregated at once. The proposed "regional pooling" schemes are depicted in (b)-(d), where points are aggregated only in distinct spatial regions. Only, the solid red points are aggregated. For better visual representation, cones/spheres along only one axis are shown. (b) and (c) use the conical pooling while (d) adopts spherical pooling in local regions.

3.3. Feature Aggregation

The point features need to be aggregated into a global point cloud feature for classification. A symmetric aggregation function such as max or average pooling is a popular choice for feature aggregation. Four aggregations (the max, mean, l_1 norm, and l_2 norm) have been used in PointHop and PointHop++. Instead of aggregating all points globally at once as shown in Fig. 2 (a), we propose to aggregate subsets of points from different spatial regions here. We consider regions of the 3D volume defined by cones and spheres.

For conical aggregation, we consider two types of cones, one with tip at the origin and the other with tip at a unit distance along the principal axes. They are illustrated in Figs. 2 (b) and (c), respectively. The latter cone cuts the plane formed by the other two principal axes in a unit circle and vice versa for the former. For each principal axis, we get four such cones, two along the positive axis and two along the negative. Thus, 12 cones are formed for all three axes in total. For each cone, only the features of points lying inside the cone are pooled together. The pooling methods are the max, mean, variance, l_1 norm, and l_2 norm. This means for a single point feature dimension, we get a 5D feature vector from each cone.

For spherical aggregation, we consider four spheres of a quarter radius centered at a distance of positive/negative one and three quarters from the origin along each principal axis. One example is illustrated in Fig. 2 (d). This gives 12 spheres in total. Points lying in each sphere are pooled together in a similar manner as cones. For instance, points lying in different cones for four point cloud objects are shaded in Fig. 3.

Unlike max/average pooling, aggregating local feature descriptors into a global shape descriptor such as Bag of Words (BoW) or Vector of Locally Aggregated Descriptors (VLAD) [25] is common in traditional literature. On the other hand, the region-based local spatial aggregation has never been explored before. These resulting features are powerful in capturing local geometrical characteristics of objects.

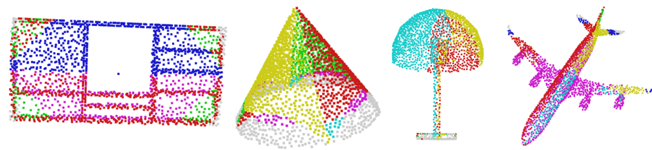

Fig. 3. An example of conical aggregation. For every point cloud object, points lying in each cone are colored uniquely.

3.4. Discriminant Feature Selection and Classification

In order to select a subset of discriminant features for classification, we adopt the Discriminant Feature Test (DFT) as proposed in [26]. DFT is a supervised learning method that can rank features in the feature space based on their discriminant power. Since they are calculated independently of each other, the DFT computation can be parallelized. Each 1D feature f^i of all point clouds are collected and the interval $[f^i_{min}, f^i_{max}]$ is partitioned into two subspaces S^i_L and S^i_R about an optimal threshold f^i_{op}. Then, the purity of each subspace is measured by a weighted entropy loss function. A smaller loss indicates stronger discriminant power. DFT helps control the number of features fed to the classifier. As shown in Sec. 4, it improves the classification accuracy significantly and prevents classifier from overfitting. In our experiments, we select top 2700 features. Finally, we train a linear least squares classifier to predict the object class.

4. EXPERIMENTS

We evaluate the proposed S3I-PointHop method for the point cloud classification task on the ModelNet40 dataset [27], which consists of 40 object classes. Objects in ModelNet40 are pre-aligned. We rotate them in the train and test sets in the following experiments. The rotation angles are uniformly sampled in $[0, 2\pi]$. We use z to denote random rotations along the azimuthal axis and $SO(3)$ to indicate rotations about all three orthogonal axes. In Tables 1, 2 and 3, $z/SO(3)$ means that the training set follows the z rotations while the test set adopts $SO(3)$ rotations, and so on. For all experiments, we set the numbers of nearest neighbors in calculating geometric, covariance, and octant features to be 128, 32, and 64, respectively.

4.1. Comparison with PointHop-family Methods

Table 1 compares the performance of S3I-PointHop, PointHop [2], PointHop++ [6] and R-PointHop [5]. Clearly, S3I-PointHop outperforms the three benchmarking methods by a huge margin. Although R-PointHop was proposed for point cloud registration and not classification, we include it here due to its rotation invariant feature characteristics. Similar to the global aggregation in PointHop and PointHop++, we aggregate the point features of R-PointHop and train a Least Squares classifier. We also report the classification accuracy with only one hop for these methods. Both PointHop and PointHop++ perform poor since their features are not invariant to rotations. Especially, for the $z/SO(3)$ case, there is an imbalance in the train and test sets, the accuracy is worse. R-PointHop only considers local octant features with respect to a local reference frame. Although they are invariant to rotations, they are not optimal for classification.

Table 1. Classification accuracy comparison of PointHop-family methods.

Method	# hops	z/z	z/SO(3)	SO(3)/SO(3)
PointHop [2]	1	70.50	21.35	45.70
	4	75.12	22.85	50.48
PointHop++ [6]	1	9.11	7.90	9.09
	4	82.49	20.62	57.61
R-PointHop [5]	1	53.44	53.42	53.44
	4	64.87	64.86	64.86
S3I-PointHop	1	**83.10**	**83.10**	**83.10**

4.2. Comparison with Deep Learning Networks

We compare the performance of S3I-PointHop with 4 deep-learning-based point cloud classification networks in Table 2. They are PointNet [1], PointNet++ [14], PointCNN [16] and Dynamic Graph CNN (DGCNN) [15]. Since these methods were originally developed for aligned point clouds, we retrain them with rotated point clouds and report the corresponding results. We see from the table that S3I-PointHop outperforms these benchmarking methods significantly. These methods offer reasonable accuracy when rotations are restricted about the azimuthal (z) axis. However, they are worse when rotations are applied about all three axes.

Table 2. Comparison with Deep Learning Networks.

Method	z/z	z/SO(3)	SO(3)/SO(3)
PointNet [1]	70.50	21.35	45.70
PointNet++ [14]	75.12	22.85	50.48
PointCNN [16]	82.11	24.89	51.66
DGCNN [15]	82.49	20.62	57.61
S3I-PointHop	**83.10**	**83.10**	**83.10**

4.3. Ablation Study

It is worthwhile to consider the contributions of different elements in S3I-PointHop. To do so, we conduct an ablation study and report the results in Table 3. From the first three rows, it is evident that the global octant features are most important, and their removal results in the highest drop in accuracy. The results also reinforce the fact that locally oriented features such as those in R-PointHop are not optimal for classification. In rows 4 and 5, we compare the proposed spatial aggregation scheme (termed as local aggregation) with global pooling as done in PointHop. The accuracy sharply drops by 12% when only the global aggregation is used. Clearly, global aggregation is not appropriate in S3I-PointHop. Finally, we show in the last row that the accuracy drops to 78.56% without DFT. The is because, when the feature dimension is too high, the classifier can overfit easily without DFT.

Table 3. Ablation Study

Feature			Aggregation		DFT	SO(3)/SO(3)
Geometric	Covariance	Octant	Local	Global		
	✓	✓	✓		✓	82.49
✓		✓	✓		✓	82.45
✓	✓		✓		✓	80.75
✓	✓	✓	✓		✓	**83.10**
✓	✓	✓		✓	✓	71.02
✓	✓	✓	✓			78.56

4.4. Discussion

One advantage of S3I-PointHop is that its rotation invariant features allow it to handle point cloud data captured from different orientations. To further support this claim, we retrain PointHop with PCA coarse alignment as a pre-processing step during the training and the testing. The test accuracy is 78.16% and 74.10% with four-hop and one-hop, respectively. This reinforces that only the PCA alignment is not the reason for the performance gain of S3I-PointHop. While efforts to learn rotation invariant features were already made in R-PointHop, we see that the lack of global features in it degrades its performance. On the other hand, appending the same global feature to R-PointHop does not help in the registration problem.

An interesting aspect of S3I-PointHop is its use of a single hop (rather than four hops such as in PointHop). It is generally perceived that deeper networks perform better than shallower counterparts. However, the use of multiple spatial aggregations on top of a single hop, S3I-PointHop can achieve good performance. This leads to the benefit of reducing the training time and the model size as explained below.

In any point-cloud processingn method, one of the most costly operations is the nearest neighbor search. To search the k nearest neighbors of each of N points, the complexity of an efficient algorithm is $O(k \log N)$. PointHop uses the nearest neighbor search in four hops and three intermediate farthest point downsampling operations. In contrast, the nearest neighbor search is only conducted once for each point in S3I-PointHop. Another costly operation is the PCA in the Saab transform. It is performed only once in S3I-PointHop. Its model size is 900 kB, where only one-hop Saab filters are stored.

5. CONCLUSION AND FUTURE WORK

A point cloud classification method called S3I-PointHop was proposed. It extends PointHop-like methods for 3D classification of objects which have arbitrary orientations. S3I-PointHop extracts local and global point neighborhood information using an ensemble of geometric, covariance and octant features. Only a single hop is adopted in S3I-PointHop followed by conical and spherical aggregations of point features from multiple spatial regions. There are several possible extensions of this work. It is desired to further improve the performance of S3I-PointHop and compare it with that of state-of-the-art rotation invariant and equivariant networks. Furthermore, it is interesting to examine the application of single-hop rotation invariant methods to the registration problem and the pose estimation problem.

6. REFERENCES

[1] 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.

[2] M. Zhang, H. You, P. Kadam, S. Liu, and C.-C. J. Kuo, "Pointhop: An explainable machine learning method for point cloud classification," *IEEE Transactions on Multimedia*, vol. 22, no. 7, pp. 1744–1755, 2020.

[3] M. Zhang, P. Kadam, S. Liu, and C.-C. J. Kuo, "Unsupervised feedforward feature (uff) learning for point cloud classification and segmentation," in *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2020, pp. 144–147.

[4] Zhang, Min and Kadam, Pranav and Liu, Shan and Kuo, C-C Jay, "GSIP: Green semantic segmentation of large-scale indoor point clouds," *Pattern Recognition Letters*, vol. 164, pp. 9–15, 2022.

[5] Kadam, Pranav and Zhang, Min and Liu, Shan and Kuo, C-C Jay, "R-pointhop: A green, accurate, and unsupervised point cloud registration method," *IEEE Transactions on Image Processing*, 2022.

[6] M. Zhang, Y. Wang, P. Kadam, S. Liu, and C.-C. J. Kuo, "Pointhop++: A lightweight learning model on point sets for 3d classification," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 3319–3323.

[7] C.-C. J. Kuo and A. M. Madni, "Green learning: Introduction, examples and outlook," *Journal of Visual Communication and Image Representation*, p. 103685, 2022.

[8] C.-C. J. Kuo, M. Zhang, S. Li, J. Duan, and Y. Chen, "Interpretable convolutional neural networks via feedforward design," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 346–359, 2019.

[9] Y. Chen, M. Rouhsedaghat, S. You, R. Rao, and C.-C. J. Kuo, "Pixelhop++: A small successive-subspace-learning-based (ssl-based) model for image classification," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 3294–3298.

[10] P. Kadam, M. Zhang, S. Liu, and C.-C. J. Kuo, "Unsupervised point cloud registration via salient points analysis (spa)," in *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2020, pp. 5–8.

[11] P. Kadam, M. Zhang, J. Gu, S. Liu, and C.-C. J. Kuo, "Greenpco: An unsupervised lightweight point cloud odometry method," in *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2022, pp. 01–06.

[12] P. Kadam, Q. Zhou, S. Liu, and C.-C. J. Kuo, "Pcrp: Unsupervised point cloud object retrieval and pose estimation," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 1596–1600.

[13] S. Liu, M. Zhang, P. Kadam, and C.-C. J. Kuo, *3D Point Cloud Analysis: Traditional, Deep Learning, and Explainable Machine Learning Methods*. Springer, 2021.

[14] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[15] 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 (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[16] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "Pointcnn: Convolution on x-transformed points," *Advances in neural information processing systems*, vol. 31, 2018.

[17] H. Deng, T. Birdal, and S. Ilic, "Ppfnet: Global context aware local features for robust 3d point matching," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 195–205.

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

[19] X. Li, R. Li, G. Chen, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "A rotation-invariant framework for deep point cloud analysis," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[20] F. Li, K. Fujiwara, F. Okura, and Y. Matsushita, "A closer look at rotation-invariant deep point cloud analysis," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 218–16 227.

[21] C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. J. Guibas, "Vector neurons: A general framework for so (3)-equivariant networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 200–12 209.

[22] H. Chen, S. Liu, W. Chen, H. Li, and R. Hill, "Equivariant point network for 3d point cloud analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14 514–14 523.

[23] X. Li, Y. Weng, L. Yi, L. J. Guibas, A. Abbott, S. Song, and H. Wang, "Leveraging se (3) equivariance for self-supervised category-level object pose estimation from point clouds," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15 370–15 381, 2021.

[24] T. Hackel, J. D. Wegner, and K. Schindler, "Fast semantic segmentation of 3D point clouds with strongly varying density," *ISPRS annals of the photogrammetry, remote sensing and spatial information sciences*, vol. 3, pp. 177–184, 2016.

[25] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3304–3311.

[26] Y. Yang, W. Wang, H. Fu, C.-C. J. Kuo *et al.*, "On supervised feature selection from high dimensional feature spaces," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1.

[27] 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.