# TRIANGLE-NET: TOWARDS ROBUSTNESS IN POINT CLOUD CLASSIFICATION

*Chenxi Xiao*

School of Industrial Engineering
Purdue University
xiao237@purdue.edu

*Juan Wachs*

School of Industrial Engineering
Purdue University
jpwachs@purdue.edu



## ABSTRACT

3D object recognition is becoming a key desired capability for many computer vision systems such as autonomous vehicles, service robots and surveillance drones to operate more effectively in unstructured environments. These real-time systems require effective classification methods that are robust to sampling resolution, measurement noise, and pose configuration of the objects. Previous research has shown that sparsity, rotation and positional variance of points can lead to a significant drop in the performance of point cloud based classification techniques. In this regard, we propose a novel approach for 3D classification that takes sparse point clouds as input and learns a model that is robust to rotational and positional variance as well as point sparsity. To this end, we introduce new feature descriptors which are fed as an input to our proposed neural network in order to learn a robust latent representation of the 3D object. We show that such latent representations can significantly improve the performance of object classification and retrieval. Further, we show that our approach outperforms PointNet and 3DmFV by 34.4% and 27.4% respectively in classification tasks using sparse point clouds of only 16 points under arbitrary SO(3) rotation.

***Index Terms***— Point cloud, Deep learning, Descriptor

## 1. INTRODUCTION

As commodity cameras and laser based sensors become more affordable, point cloud based object classification is becoming the default approach for 3D sensing. For example, autonomous vehicles rely on point cloud maps sampled by Lidar sensors or depth cameras for effective navigation. One challenge often faced in such applications is that the density of sampling points decreases significantly as the distance from vehicle embedded sensor to the object increases. This makes it hard to recognize objects that are far from such sensors due to their sparse inherent structure [1]. As reported in the literature [2, 3, 4], the classification accuracy of these algorithms drops radically as the density of the point cloud decreases, and is further affected when the pose configuration of the object is not known in advance. Similarly, consider the scenario of tactile based object recognition using a robotic hand. The time complexity of sampling is proportional to the number of points sampled along the manipulator's trajectory [5]. This implies that in addition to performance degradation, there is an additional cost related to the amount of sampling required to make an acceptable prediction. Thus, it is necessary to come up with new 3D machine learning techniques that can classify objects based on "limited" sparse point cloud data and that can operate in real-time, whether for effective navigation (e.g. autonomous driving case) or for user's meaningful perception (e.g. tactile sampling).

In this paper, we propose a new technique for 3D object classification that is meant to perform well when recognizing objects with only few sample points. In this regard, the main contributions of the paper are as follows: 1) A new 3D descriptor for extracting features from sparse point clouds, in order to relax the requirements on the point cloud size or density. This descriptor is rotational and positional invariant so that the discriminative ability of the classifier remains same when the object undergoes arbitrary affine transformations, and 2) a deep neural network model to learn a latent representation that can be used for common machine learning tasks such as 3D classification and reconstruction.

## 2. RELATED WORK

### 2.1. 3D Object Classification

Existing approaches for point cloud classification mainly include but not limited to: 1) Directly performing classification on point cloud data [2, 3]. 2) Projecting the point cloud data into other dimensions that are easier for classification, such as voxelized objects [6, 7] or 2D images taken from multiple view angles [8, 9, 10]. 3) Learning from hand-crafted features that can be created using point cloud data [4, 5, 11, 12]. 4) Finding latent feature representations by reconstruction [13, 14, 15].

In addition to those main groups, other approaches such as [3, 11] set an explicit threshold on the minimum number of points and therefore cannot be applied to point cloud classification when the structure is too sparse. More recently, a new family of approaches based on 2D convolutions [8, 9] or 3D convolutions [6] have been suggested. However, they have been found not suitable when the points are too sparse

due to the low magnitude of correlations between neighboring regions (most regions are void). The work done by [12] can be extended to classify sparse point cloud and hence our work is closely aligned with their approach.

### 2.2. 3D Feature Descriptors

Descriptors are feature representations that contain statistical information about the 3D objects being represented by them. Some well known descriptors are PFH [16], FPFH [17] and HKS [18] to mention a few. However, most of the existing 3D descriptors are designed to work on dense points or meshes since the statistical information is more faithful when the observed data is abundant. There is limited literature that builds on sparse points. Among them we build on the descriptor presented by [12], which uses a triangle parameter based descriptor for classification. This is because the discriminability of this descriptor is accurate even when the points in the point cloud are sparse while it requires less amount of information.

### 2.3. Position and Orientation Invariance

Real-world objects can be found in arbitrary shapes and poses and that is why feature descriptors should stay invariant to rotation and position changes of those objects. In this context, the robustness to positional and rotational changes is either learned [2, 3, 4] or manually introduced using prior knowledge [11]. However, [11] showed that learned robustness can degrade when generalized to scenarios where the rotation is not present in the training process, leading to under-performance when compared to scenarios where no rotation is applied. [11] showed that this issue has been observed in most techniques used for point cloud object classification. To address this problem, we propose to use a descriptor that is invariant to rotation and position changes because it leverages on global and local intrinsic structure information of the point-cloud.

## 3. METHODOLOGY

Our approach for 3D object classification is two-fold. First, we introduce a new feature descriptor in order to improve the object discriminability. Next, we focus on a deep learning approach that can transform descriptor representation to a latent representation, which can then facilitate object classification, retrieval as well as reconstruction.

### 3.1. Proposed Descriptor

We achieve robust object classification through: 1) Rotational and positional descriptor invariance; 2) Combination of information from local and global scales in order to improve the representative ability. Based on the above design criteria, we have developed a series of descriptors shown in Figure 1.

Figure 1 (a) is a Type-A descriptor that can be constructed only using 2 points. The surface normal is the simplest local feature. The distance between two points is a global feature that contains global shape information.

Figure 1 (b) is a Type-B descriptor which can be constructed using 3 points. This descriptor is a combination of three Type-A descriptors but emphasizes the superimposed positional relationship.

Figure 1 (c) is a Type-C descriptor made of connecting vertex points to the center point, and then using the segment distance and vertex angles as descriptor features. These precomputed information can slightly improve the accuracy compared to Type-B descriptor according to our experiments.

Efficiency is obtained by using only $K$ points, we can compute $C_K^2$ non-repetitive Type-A descriptors, or compute $C_K^3$ non-repetitive Type-B or Type-C descriptors. We generate a fix number of descriptors and then feed these descriptors to the input of the network proposed in section 3.2.

Optionally, scale invariant can be achieved by dividing each side length by $d_{max}$, where $d_{max}$ is the maximum side length among all the input descriptors.

### 3.2. Network Architecture

We use a deep neural network for object classification. The backbone of the network architecture is shown in Figure 2. This architecture corresponds to a feature extraction network to map the descriptors to latent space representations. These latent features are used for both object reconstruction and object classification. We learn both classification and reconstruction tasks simultaneously. This is a multi-task learning architecture that rewards the network to learn the underlying structure of the input data. For simplicity, we omit the network structure of classification network as it shares the same structure as the output MLP network in [2].

Note that the rows of the input descriptor matrix can be arbitrarily permuted without affecting the network output. This is induced by a Max-Pooling layer that compresses all descriptor information into a single vector. We borrow this idea of isolating input permutations from [2].

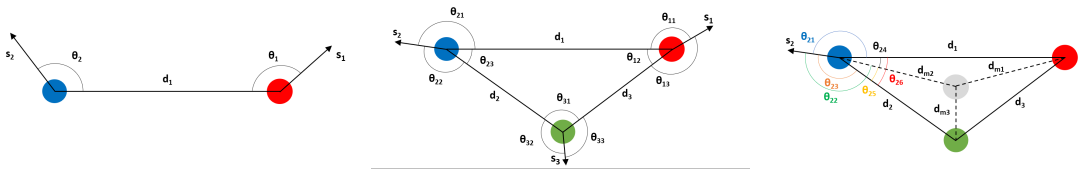
## 4. EXPERIMENTS AND RESULTS

### 4.1. Experimental Setting

Several experiments are conducted on ModelNet 40 dataset [19]. All the experiments are conducted on a desktop workstation with Intel I7-9700k CPU and Nvidia RTX 2070 GPU. Our experimental code is available at `https://github.com/MegaYEye/sparse-paper-code`.

### 4.2. Ablation Study 1: Rotation and Sparsity Variation

We design an experiment to show the effect of SO(3) transformation on the PointNet model. For this, we both trained and tested the PointNet model under 3 conditions: a) Point clouds



(a) (b) (c)

**Fig. 1**. We propose three variations of descriptors. Type-A descriptor is constructed using only 2 points with surface normal vectors. Type-B descriptor is constructed using 3 points with surface normal vectors. Type-C descriptor also uses 3 points with surface normal vectors with more pre-computed information.

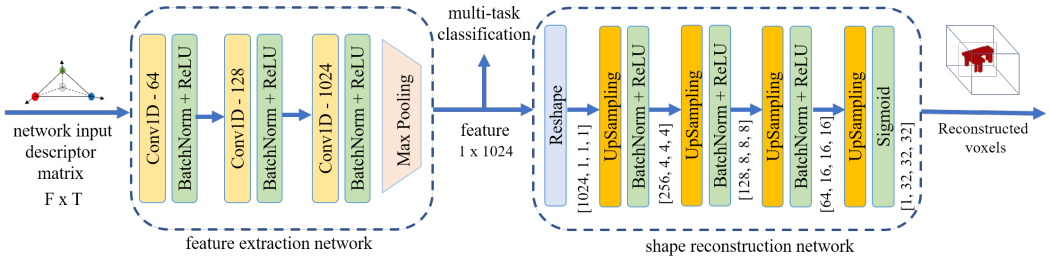



**Fig. 2**. Our proposed network uses the rotational and positional invariant descriptor as the input. We show the network structure of feature extraction network and object reconstruction network.

**Table 1**. The experiment shows the performance degradation of PointNet model in (in %) when using a) No rotation b) Only rotate around Z axis or c) Arbitrary SO(3) rotation (as indicated in different rows) under different point cloud density (as shown in columns)

| No. of points | 1024 | 256 | 64 | 16 |
|---|---|---|---|---|
| No rotation applied | 88.51 | 86.89 | 82.49 | 76.40 |
| Rotated around Z axis | 84.01 | 77.33 | 69.31 | 53.33 |
| Arbitrary SO(3) rotation | 79.08 | 72.01 | 56.79 | 35.28 |

with no rotation b) Only rotate around Z axis or c) Arbitrary SO(3) rotations.

The results in Table 1 show that PointNet performs well with dense points cloud under all 3 rotational conditions, as indicated in the first column. The result also shows that PointNet scales well when no rotation is applied, as indicated in the first row. However, performance decays as rotation is applied to the data. It can be observed that rotation around the Z-axis affects overall performance, and this is further aggravated when arbitrary SO(3) rotations are applied.

### 4.3. Classification on Sparse and Rotated Points

In this experiment, we first compare the performance of our method to other approaches under various point density configurations. The objects in the ModelNet 40 dataset undergo arbitrary SO(3) transformations both in training and testing sets. The experimental results are shown in Table 2. The last row of the table shows our result. Likewise, the highest performance is labeled in **bold** digits.

PointNet was tested under two conditions. Vanilla PointNet[1] trained with 1024 points with random input dropout [3]. PointNet[2] is also PointNet but trained and tested using the same number of points. We see that PointNet[1] fail to generalize sufficiently well to sparse point clouds.

From the same table, we can see that 3DmFV [4] can perform better than PointNet when the input points are sparse, but it also decays when the point cloud becomes largely sparse when arbitrary SO(3) rotations are also applied.

We also compared our algorithm with RI-CONV [11], which relies on rotational-invariant descriptors. However, this approach is constrained to operate over a minimum number of points per region, which makes it unsuitable for sparse points classification.

The above comparisons show that our approach can outperform others by a large margin when points are sparse. Conversely, our approach does not perform the best for class prediction when using dense point clouds. We believe this is mainly due to 2 reasons. 1) Part of relative positional information between points is discarded, as each descriptor is constructed using 3 points rather than all points. 2) When the point cloud is dense, there is almost an infinite number of descriptors that can be constructed (e.g. $1 \times 10^9$ possible descriptors when using 1024 points) and the subset of descriptors chosen by our method may be sub-optimal to represent the object of interest

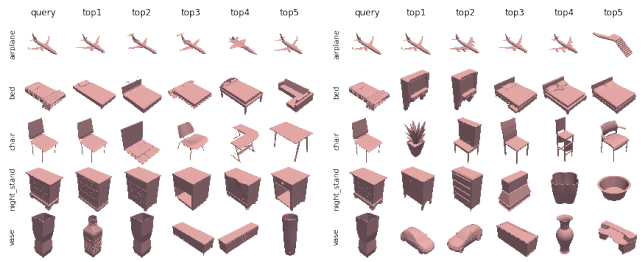
**Table 2**. Comparison of classification performance on both dense and sparse points. Our algorithm shows the advantage when points become sparse

| | Dense | | | | Sparse | | | |
|---|---|---|---|---|---|---|---|---|
| Num of points | 1024 | 512 | 256 | 128 | 64 | 32 | 16 | 8 |
| PointNet[1] | 73.09 | 72.67 | 64.48 | 39.93 | 21.08 | 9.79 | 2.65 | 2.07 |
| PointNet[2] | 79.08 | 75.14 | 72.01 | 72.64 | 56.79 | 48.34 | 35.28 | 23.91 |
| PointNet++ | 84.76 | 83.87 | 83.31 | 78.60 | N/A | N/A | N/A | N/A |
| 3DmFV | **86.63** | **85.69** | 84.70 | 82.32 | 76.56 | 63.45 | 42.26 | 23.68 |
| RI-CONV | 86.5 | 84.4 | 80.8 | 76.0 | N/A | N/A | N/A | N/A |
| Ours | 85.98 | 85.53 | **85.12** | **83.26** | **81.32** | **79.13** | **69.69** | **48.14** |

### 4.4. Learned Representation on Shape Similarity

Our learned representation can be used as a metric for comparing shape similarity. The following experiments conducted show that this metric is valid even when the point cloud is sparse.

The experiment below shows the performance of learned shape similarity metric using only 16 points. The model was trained and tested with the respective training and testing dataset. The top 5 similar objects are found within the test dataset using the nearest neighborhood algorithm with the $L^2$ metric. The retrieval results of our approach is shown in Figure 3 (a). The comparison with PointNet is shown in Figure 3 (b).




(a) (b)

**Fig. 3**. Results of the retrieval operation of our approach (a) vs PointNet model retrieval results (b) using only 16 points.

We used retrieval MAP (mean averaged precision) as a metric for a quantitative comparison of our approach to PointNet. Our approach achieves 59.97% and 56.67% in top-5 and top-10 retrieval results respectively. while PointNet achieved 34.80% and 35.04% correspondingly. We believe that the boost in peformance is from a better discriminative ability of our descriptor and a better similarity metric learned by object reconstruction.

### 4.5. Object Reconstruction Using Sparse Points

In this section, we show object reconstruction results. We used networks trained by 4096 descriptors generated by 16 points. Note a voxel is placed only when the output (binary Sigmoid function) is larger than 0.2 (instead of 0.5, as the normal Sigmoid case) because we found the network output becomes less "confident" as the input points become sparse. We demonstrate the reconstruction result in Figure 4. While the reconstruction result resembles the original object, some reconstruction artifacts can still be seen. These includes cluttered points and inaccurate shape details. We believe this is mainly due to limited descriptive information from the sparse points dataset.

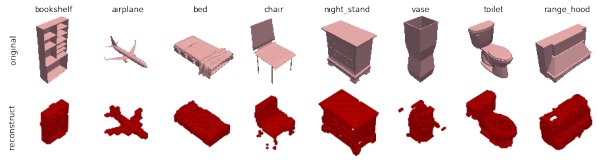



**Fig. 4**. Object shape reconstruction using only 16 points. Even though the input information is very scarce, reasonable reconstruction results can still be achieved.

### 4.6. Ablation Study 2: Classification and Reconstruction

We compare several variations of our approach quantitatively through an experiment using 16 points only. We found that all our proposed descriptors can outperform the others using raw triangle parameters [12], which only achieves 32.57%. Classification results increase as more information added to the descriptor. Accuracies of 60.08%, 65.92%, and 67.26% are achieved when using type A/B/C descriptors respectively, and the classification accuracy is boosted to 69.69% when being trained together with object reconstruction. The later scenario corresponds to the highest accuracy we achieved.

## 5. CONCLUSIONS

In this paper, we proposed a 3D classifier that is designed for sparse point cloud classification challenges. A novel 3D descriptor is proposed to improve the discriminability on sparse points datasets. Our descriptor is used in combination with deep neural network for learning a latent representation, which then facilitates object classification, retrieval and reconstruction. Our results show that our approach performs better than the state of the art on sparse points with arbitrary SO(3) rotations.

## 6. ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant NSF NRI #1925194. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## 7. REFERENCES


[1] Igor Bogoslavskyi and Cyrill Stachniss. Efficient online segmentation for sparse 3d laser scans. *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 85(1):41–52, 2017.

[2] 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.

[3] 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.

[4] 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.

[5] Mabel M Zhang, Nikolay Atanasov, and Kostas Daniilidis. Active end-effector pose selection for tactile object recognition through monte carlo tree search. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3258–3265. IEEE, 2017.

[6] 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.

[7] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv preprint arXiv:1608.04236*, 2016.

[8] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 945–953, 2015.

[9] Jong-Chyi Su, Matheus Gadelha, Rui Wang, and Subhransu Maji. A deeper look at 3d shape classifiers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[10] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Multi-view 3d models from single images with a convolutional network. In *European Conference on Computer Vision (ECCV)*, 2016.

[11] Zhiyuan Zhang, Binh-Son Hua, David W Rosen, and Sai-Kit Yeung. Rotation invariant convolutions for 3d point clouds deep learning. In *2019 International Conference on 3D Vision (3DV)*, pages 204–213. IEEE, 2019.

[12] Mabel M Zhang, Monroe D Kennedy, M Ani Hsieh, and Kostas Daniilidis. A triangle histogram for object classification by tactile sensing. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4931–4938. IEEE, 2016.

[13] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in neural information processing systems*, pages 82–90, 2016.

[14] David Stutz and Andreas Geiger. Learning 3d shape completion under weak supervision. *International Journal of Computer Vision*, pages 1–20, 2018.

[15] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5868–5877, 2017.

[16] 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.

[17] 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.

[18] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, volume 28, pages 1383–1392. Wiley Online Library, 2009.

[19] 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.