GPr-Net: Geometric Prototypical Network for Point Cloud Few-Shot Learning

Tejas Anvekar*, Dena Bazazian†

* KLE Technological University, Center of Excellence in Visual Intelligence (CEVI)
† University of Plymouth, Faculty of Science and Engineering

* anvekartejas@gmail.com, † dena.bazazian@plymouth.ac.uk

Abstract

In the realm of 3D-computer vision applications, point cloud few-shot learning plays a critical role. However, it poses an arduous challenge due to the sparsity, irregularity, and unordered nature of the data. Current methods rely on complex local geometric extraction techniques such as convolution, graph, and attention mechanisms, along with extensive data-driven pre-training tasks. These approaches contradict the fundamental goal of few-shot learning, which is to facilitate efficient learning. To address this issue, we propose GPr-Net (Geometric Prototypical Network), a lightweight and computationally efficient geometric prototypical network that captures the intrinsic topology of point clouds and achieves superior performance. Our proposed method, IGI++ (Intrinsic Geometry Interpreter++) employs vector-based hand-crafted intrinsic geometry interpreters and Laplace vectors to extract and evaluate point cloud morphology, resulting in improved representations for FSL (Few-Shot Learning). Additionally, Laplace vectors enable the extraction of valuable features from point clouds with fewer points. To tackle the distribution drift challenge in few-shot metric learning, we leverage hyperbolic space and demonstrate that our approach handles intra and inter-class variance better than existing point cloud few-shot learning methods. Experimental results on the ModelNet40 dataset show that GPr-Net outperforms state-of-the-art methods in few-shot learning on point clouds, achieving utmost computational efficiency that is $170\times$ better than all existing works. The code is publicly available at https://github.com/TejasAnvekar/GPr-Net.

1. Introduction

The domain of computer vision has witnessed a remarkable surge in the significance of 3D data processing, with point cloud data emerging as a prominent representation obtained via real-time acquisition using LiDAR scanners.

Laplace Vectors $\vec{L} = p_i \bigoplus \frac{1}{K} \sum_{j=0}^{K} (p_j - p_i)$
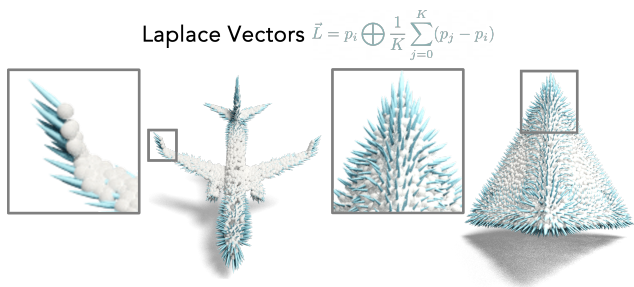
Figure 1. We demonstrate Laplace vectors, a simple yet effective geometric signature that captures the statistics of group deviations, facilitating the abstraction of edges and corners in point clouds. The efficacy of Laplace vectors is highlighted in the visualization of an airplane and a tetrahedron. In the airplane, Laplace vectors capture the uni-directed high group deviation at the ends of the wings, indicating a sudden change or an edge in local topology. Similarly, the tetrahedron exhibits high and uniform group deviation at its ends, indicating a corner.

Point cloud object classification plays a critical role in several applications such as indoor SLAM [44], robotics [27], and autonomous vehicles [18], facilitating efficient navigation and decision-making. Deep learning-based techniques [24] [22] have revolutionized 3D point cloud classification by enabling the extraction of representative features from shape projections [32] or raw points, thereby enhancing performance compared to traditional handcrafted feature-based methods [26] [42].

Despite the recent progress in geometric deep learning, the need for large amounts of labeled training data remains a significant challenge, both in terms of cost and practicality [25]. While self-supervised approaches [1] [41], data augmentation [15] [29], and regularisation [22] techniques have helped alleviate the aforementioned issue, they may not perform well on new tasks or unseen classes without sufficient labeled training data. This has led to a growing demand for methods that enable geometric deep networks

to quickly adapt to novel settings with limited labeled data, much like humans who can learn new concepts with only a few examples by drawing on prior knowledge and inductive bias [3].

To address this challenge, few-shot learning (FSL) techniques [30] [33] [9] have shown remarkable progress in 2D visual understanding tasks such as image classification [12], object detection [11], and semantic segmentation [7]. However, FSL on 3D data is still in its nascent stages and presents unique challenges. Previous approaches to 3D-FSL [28] [8] [39] have focused on determining the best FSL algorithm, network design, and deep learning methodology, often relying on complex pre-training tasks or intricate deep learning modules. These approaches may not effectively capture the human-inspired characteristics that researchers aim to incorporate, leading to limited generalization.

Towards addressing the aforementioned challenges, we introduce a novel 3D-FSL approach, the Geometric Prototypical Network (GPr-Net), which leverages geometric priors to achieve fast and efficient point cloud few-shot learning. Unlike conventional approaches that rely on complex pre-training [28] procedures or sophisticated deep learning modules [39], GPr-Net is engineered to transfer geometric prior knowledge directly to novel tasks with minimal training. To capture these valuable geometric priors for 3D-FSL, we propose Intrinsic Geometry Interpreters++ (IGI++), which efficiently captures the local intrinsic topology of the point cloud using the IGI features inspired by the VG-VAE [2]. Additionally, we propose Laplace vectors to extract abstract information about the edges and corners present in point clouds. The coherently combined intrinsic and Laplace vectors of IGI++ provide a comprehensive representation of the crucial geometric properties for few-shot learning on point clouds as shown in Figure 1. Furthermore, we address the distribution shift in prototypical networks by mapping our geometric priors to the Hyperbolic metric. Extensive experiments on the ModelNet40 dataset demonstrate the superiority of GPr-Net in few-shot learning on point clouds compared to state-of-the-art methods. GPr-Net achieves up to $170\times$ fewer parameters that facilitate faster performance and a 5% increase in accuracy compared to related works.

Our contributions can be summarized as:

- We propose GPr-Net: a lightweight Geometric Prototypical Network designed for fast and efficient point cloud few-shot learning.

- We propose an Intrinsic Geometry Interpreters++ (IGI++) to cohere intrinsic and high-frequency geometric signatures of a point cloud which comprises the following modules: 1) an Intrinsic Geometric Interpreter (IGI) to efficiently capture the local topology of the point cloud; 2) our proposed novel Laplace vec-

tors to capture the abstraction of edges and corners in point clouds.

- We propose employing the Hyperbolic / poincaré metric to mitigate the challenge of distribution shift in prototypical networks.

- We demonstrate the impact of our derived geometric signatures on ModelNet40 and outperforms existing state-of-the-art few-shot learning techniques by 5% in accuracy with $170\times$ fewer parameters.

2. Related Works

Point Cloud Analysis has been revolutionized by deep learning models for 3D point cloud classification by allowing us to learn more intricate and representative features. Unlike traditionally handcrafted methods [26] [42], these models can learn these features without any human intervention. There are two types of deep learning methods: projection-based and point-based networks. Projection-based [32] networks first transform irregular points into a structured representation such as voxel [43] or lattices and then use standard convolution neural networks to extract view-wise or structural features. However, they may encounter explicit information loss or higher memory consumption. Point-based methods have become more popular, exemplified by the likes of PointNet [24] and other approaches like PointNet++ [25] and PointCNN [16], and DGCNN [34], Point-MLP [20], HyCoRe [22], EDC-Net [5], DCG-Net [4], which utilize convolution or graph-based networks to achieve state-of-the-art performance. While these deep learning methods require a significant amount of annotated data, their generalization capabilities on novel classes during training may be limited. This limitation could potentially be a subject of future research.

Few-shot learning (FSL) has emerged as a crucial field in machine learning that aims to overcome the limitations of traditional supervised learning methods, which require large labeled datasets to generalize to new tasks. To achieve this, several approaches have been proposed, including Prototypical Networks [30], which introduced the concept of prototypes for few-shot classification, and Relation Networks [33], which proposed a novel architecture that captures relations between different instances to improve accuracy. Model-Agnostic Meta-Learning (MAML) [9] takes a meta-learning approach to few-shot classification, learning an initialization of the model that can be quickly adapted to new tasks with only a few labeled examples. Recent research has tackled the challenge of 3D point cloud learning with limited training data. Sharma et al. [28]. explore feature representation through self-supervision, while LSSB [31] aim to
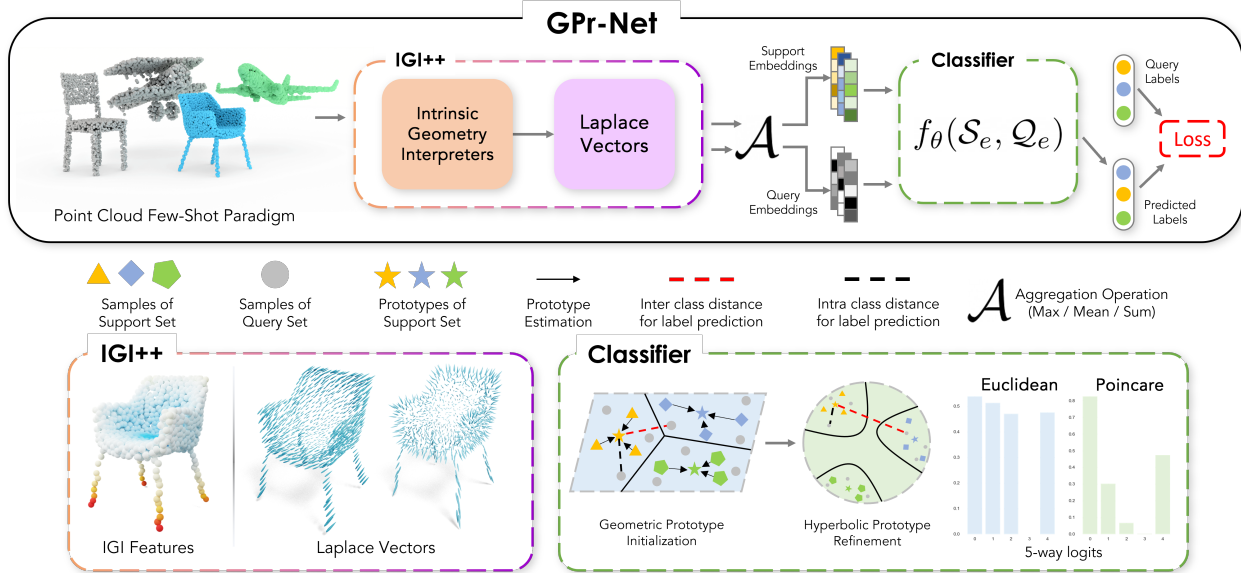
Figure 2. We present an overview of the proposed GPr-Net framework, which processes point clouds in a few-shot episodic paradigm using the proposed IGI [2] and Laplace vectors to generate geometric feature sets. These features are then mapped to a higher dimensional permutation invariant feature using the symmetric operation $\mathcal{A}$ and a single Multilayer Perceptron (MLP) f_θ. The Prototypical network f_θ, utilizes the support and query geometric embeddings $\vec{L}(\Psi(x_s)) = \mathcal{S}_e$ and $\vec{L}(\Psi(x_q)) = \mathcal{Q}_e$ to predict few-shot labels. To overcome the distribution drift challenge in Prototypical Networks, we employ the Hyperbolic Distance of Euclidean. For more details, please refer to Section 3.2.2.

learn a discriminative embedding space for 3D model multi-view images. However, authors in Enrich-Features [8] proposed a novel few-shot point cloud classification paradigm that effectively combines current fully supervised methods. This approach utilizes feature fusion and channel-wise attention to improve feature learning accuracy. These works represent important strides in addressing the challenge of 3D point cloud learning with limited data.

Hyperbolic Metric Learning embeds hierarchical structures with low distortion [23]. It has been used for non-Euclidean manifolds in various representation learning frameworks. Early on, it was used for natural language processing. Hyperbolic neural network [10] layers have been shown to be better than Euclidean ones, and hyperbolic variants have been explored for images and graphs [19]. Euclidean embeddings are insufficient for complex visual data. Hyperbolic Image Embeddings [12] address this by capturing hierarchical relationships with negative curvature, improving few-shot classification accuracy on benchmarks like miniImageNet [6] and CUB [35]. HyCoRe [22] introduced a new method for using hyperbolic space embeddings to capture the part-whole hierarchy of 3D objects in point clouds. This approach significantly improves the performance of point cloud classification models. To the best of our knowledge, no research has explored using hyperbolic representations for few-shot learning of point clouds, despite their inherent hierarchical structure and ability to

mitigate distribution drift of Prototypical networks. There exists a need for learning hyperbolic embeddings to capture the compositional nature of 3D objects with can facilitate the capture tree-like geometric hierarchy of the data, making them a superior prior for 3D-FSL.

Our method for point cloud few-shot learning introduces a novel approach that utilizes geometric signatures and Hyperbolic space to improve performance. It is distinguished from existing methods by its lightweight, fast, and pragmatic nature, requiring only a few episodes to train the challenging few-shot classification task.

3. GPr-Net

We present GPr-Net, a lightweight Geometric Prototypical Network designed for fast and efficient point cloud few-shot learning. By leveraging intrinsic geometric features, GPr-Net captures abstract information necessary for superior few-shot learning. Our proposed Intrinsic Geometry Interpreters++ (IGI++) extracts fundamental features like local topology, edges, and corners, while a single fully connected layer maps aggregation of geometric features to a higher dimensional point for episodic few-shot classification. Furthermore, we enhance our model's performance by incorporating Hyperbolic space that yields sharp logits for few-shot learning as depicted in Figure 2. Unlike previous methods, GPr-Net relies on statistical geometric fea-

tures and is trained using a few-shot paradigm to simulate real-life scenarios.

3.1. Notations and Strategies

Let P denote point cloud such that $P = \{p_1, ...p_n\}$ where $p_i \in \mathbb{R}^d$ and n represents total number of points. Inspired by the episodic paradigm of the Few-Shot Classification (FSL) task on Images [12] we incorporate a similar algorithm with minimal changes for point cloud FSL. The train set D_{train} and test set D_{test} are designed such that the categories of $D_{train} \cap D_{test} = \emptyset$.

FSL is optimized for episodes containing pair of K-ways and N-shots of the support set and query set. $N_\mathcal{S}$ samples are drawn from N_C at randomly selected K categories to form the support set $\mathcal{S} = \{(P_s^1, y_s^1), ...(P_s^{N_\mathcal{S} \times K}, y_s^{N_\mathcal{S} \times K})\}$. The remaining $N_\mathcal{Q}$ form the query set $\mathcal{Q} = \{(P_q^1, y_q^1), ...(P_q^{N_\mathcal{Q} \times K}, y_q^{N_\mathcal{Q} \times K})\}$. The goal is to predict y_q^i via a model $f_\theta(\mathcal{S}, \mathcal{Q})$ by only utilizing labels support set y_s^i.

3.2. Network Design

We advocate using Prototypical Networks [30] as a method of choice for point cloud few-shot learning due to its simplicity and remarkable generalization of metric learning. Prototypical Networks large rely on network design as it plays a key role in initializing representation metrics that significantly enhances the performance of few-shot learning. Our network design comprises two fundamental components: geometric feature extraction using proposed IGI++ and metric learning with a single fully connected MLP, both of which synergistically facilitate point cloud few-shot learning.

3.2.1 Intrinsic Geometry Interpreter++

The FSL hypothesis of our network is notably aided by learning the basic local geometric interpreters IGI++ that incorporate Intrinsic Geometry Interpreters Ψ and Laplace vectors $\vec{L}$. Following IGI which is proposed by VG-VAE [2], we introduce IGI++ and introduced group deviation vectors to facilitate the extraction of essential topological features as shown in Figure 3. Laplace vectors play a crucial role in capturing abstract information related to edges and corners of point clouds, which are essential for few-shot learning.

Intrinsic Geometry Interpreter Ψ is a set of basic local geometric features that are fast to compute and capture the local intrinsic topology of the point cloud. Ψ of a point
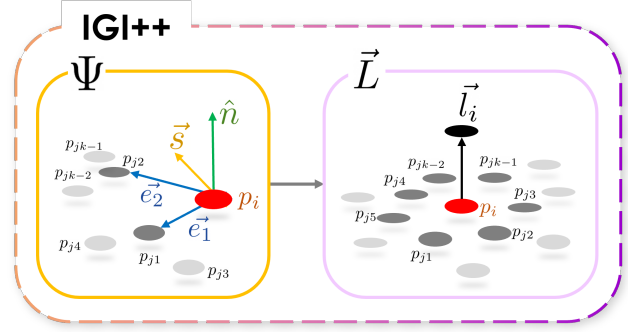

Figure 3. Illustration of proposed framework Intrinsic Geometry Interpreters++ (IGI++). **Left**: Ψ depicts computation of IGI features (normal $\hat{n}$, std vector $\vec{s}$, and edge vectors $\vec{e_1}, \vec{e_2}$) using query point p_i and two nearest neighbour p_{j1}, p_{j2}. **Right**: depicts computation of Laplace vectors $\vec{L}$ using query point p_i and neighbour points $p_{j1 \to k-1}$.

cloud P is given by:

$$\Psi = \begin{cases} p_i = x, y, z; & p_i \in \mathbb{R}^3, \\ \vec{e_1} = \frac{p_{j1} - p_i}{||\vec{e_1}||_2}; & \vec{e_1} \in \mathbb{R}^3, \\ \vec{e_2} = \frac{p_{j2} - p_i}{||\vec{e_2}||_2}; & \vec{e_2} \in \mathbb{R}^3, \\ \hat{n} = \vec{e_1} \times \vec{e_2}; & \hat{n} \in \mathbb{R}^3. \\ \vec{s} = std(p_j); & \vec{s} \in \mathbb{R}^3. \end{cases} \quad (1)$$

where $(\vec{e_1}, \vec{e_2})$ represents edge (relative positions) $(1, 2)$ respectively of a given point (position vector) p_i, $(|\vec{e_1}|, |\vec{e_2}|)$ represents edge lengths, $\hat{n}$ represent normals of point cloud p_i and $\vec{s}$ represent group deviation vector as illustrated in Figure 3. Our proposed $\Psi \in \mathbb{R}^{15}$ captures a superior geometric features due to group deviation vector $\vec{s}$, unlike Enrich Features [8] where $\Psi \in \mathbb{R}^{14}$. The relative positions and normals along with the position vector facilitate to capture of local geometric information that aids point cloud FSL. However, to capture abstract information such as edges and corners, we propose Laplace vectors, which extract high-frequency information to further improve our network's performance.

Laplace Vectors $\vec{L}$ are simple yet effective geometric signatures that capture the distribution, magnitude, and direction of the query to group deviation that facilitates extracting abstract information of edges and corners in a point clouds as shown in Figure[1] 2. Laplace vectors are given by:

$$\vec{L} = p_i \bigoplus \frac{1}{K} \sum_{j=0}^{K} (p_j - p_i) \quad (2)$$

---

[1]We acknowledge Ruben Wiersma for assisting us with these vector renderings.

where p_j are local points of p_i in a given k-NN (k-Nearest Neighbors) and $p_i \in \mathbb{R}^3$, $\vec{L} \in \mathbb{R}^6$ and $\oplus$ is concatenation operation. The Illustration of Laplace vectors is depicted in Figure 3. Laplace vectors allow us to operate with lower point density since they capture changes in local neighbourhoods. We propose to extract 30-dimensional Laplace vectors of 15-dimensional Ψ, i,e. $\vec{L}(\Psi)$ capturing superior geometric features as shown in Figure 1 towards facilitating Point Cloud FSL.

3.2.2 Metric Learning

Metric learning is essential for few-shot learning since it enables the computation of similarity metrics with limited labeled data [30] [33]. To achieve this, we aim to learn a distance metric that can robustly compare the similarity between examples. In contrast to Euclidean spaces, hyperbolic spaces offer unique properties, such as exponential growth of volume with distance, that allow for more effective modeling of complex hierarchical structures. To achieve this, we propose using the Poincaré ball model to embed features in hyperbolic spaces and perform efficient computations such as distance measurements and gradient updates. This is particularly useful in few-shot learning scenarios with limited labeled data and complex hierarchical structures [12]. Hyperbolic distance considers hierarchical relations between support and query examples, leading to improved discrimination, as shown in Figure 2. The distance between two points x, y with curvature κ in Hyperbolic (Poincaré) manifolds $\mathbb{P}_\kappa$ is given by:

$$d_{\mathbb{P}_\kappa}(x,y) = \frac{1}{\sqrt{-\kappa}} arcosh\left(\frac{-2\kappa\|x-y\|_2}{(1+\kappa\|x\|_2)(1+\kappa\|y\|_2)} + 1 \right) \quad (3)$$

3.3. Prototypical Classification

We transform support and query point clouds into geometric feature vectors by $\vec{L}(\Psi(p^i))$ given by Equation 1 and 2. The geometric features are aggregated using $\mathcal{A}$ a symmetric operation (max, mean and, sum) to get a permutation invariant global feature. The invariant global feature is used as inputs to a single fully connected MLP f_θ, which makes predictions on the label of the query point cloud by computing metric/embedding distance $d(,)$ between prototypes μ and query embedding $f_\theta(\vec{L}(\Psi(p_q^i)))$, where μ represents the mean of support embedding $f_\theta(\vec{L}(\Psi(p_s^i)))$ as depicted in Algorithm 1. To ensure that the predictions are accurate, we normalize the results into a probability distribution [40] and calculate the cross-entropy loss [14] between this distribution and the actual ground truth labels. This process facilitates the optimization of the network and improves its ability to perform few-shot learning on point clouds.

Algorithm 1: Training episode loss computation for prototypical point cloud networks

Input: $\mathcal{D} = \{(p_1, y_1), ..., (p_N, y_N)\}, d(.)$
/* where each $y_i \in \{1, ..., K\}$ and $d(.)$ is distance metric (Euclidean or Hyperbolic) as mentioned in Eq.(3)   */
Output: The loss J for a randomly generated training episode.

1 $V \leftarrow$ RANDOMSAMPLE$(\{1,...,K\}, N_C)$
2 **for** $k = 1$ **to** K **do**
3 $\quad \mathcal{S}_k \leftarrow$ RANDOMSAMPLE $(\mathcal{D}_k, N_\mathcal{S})$
4 $\quad \mathcal{Q}_k \leftarrow$ RANDOMSAMPLE $(\mathcal{D}_k/\mathcal{S}_k, N_\mathcal{Q})$
5 $\quad \mu \leftarrow \frac{1}{N_C} \sum_{(p_i, y_i) \in \mathcal{S}_k} f_\phi(p_i)$
$\quad$ /* where μ is point cloud prototype and $f_\phi() = f_\theta(\vec{L}(\Psi))$ as mentioned in Eq.(1) and Eq.(2)   */
6 $J \leftarrow 0$ **for** $k = 1$ **to** K **do**
7 $\quad$ **for** (p, y) *in* $\mathcal{Q}_k$ **do**
8 $\quad\quad J \leftarrow J + \frac{1}{N_C N_\mathcal{Q}} \Big[d(f_\phi(p), \mu) +$
$\quad\quad \log \sum_{k'} \exp -d(f_\phi(p), \mu) \Big]$

---

4. Experiments

In this section, we investigate the effectiveness of the topological point embeddings generated by our classifier f_θ for few-shot 3D object classification, using the dataset ModelNet40 [37]. ModelNet40 dataset encompasses 40 object categories that include a collection of 12,311 models. Note that our model is trained on an Nvidia GTX 1050ti GPU and PyTorch 1.11 and we use geoopt [13] for the hyperbolic operations.

4.1. Few-Shot 3D object classification

We evaluate the impact of the proposed IGI++ in our network for point cloud few-shot learning. We report the mean and standard deviation of our results with 95% confidence scores across 6 experiments with different seeds towards better reproducibility in Table 1. Unlike Enrich Features [8] and SS-FSL [28] our model is trained and tested in a few-shot setting, using only the coordinates (x, y, z) of each point. To compute Laplace vectors the number of k for the nearest neighbors is set to k = 40. The 30-dimensional Laplace vector is mapped to a 32-dimensional point using single MLP f_θ as explained in Section 3.3. We use the SGD (Stochastic Gradient Descent) optimizer for Euclidean metric and RSGD [13] for hyperbolic metric, with a momentum of 0.9 and weight decay of 0.0001, with the learning rate reduced from 0.1 to 0.001 through the cosine annealing in Algorithm 1 for 50 epochs of 4 train and 300 test few-shot episodes. We ensure the FSL paradigm via maintain-

Table 1. Few-shot classification results on ModelNet40 [37] dataset. With only 512 points, our method GPr-Net (Hyp) achieves state-of-the-art accuracy on 5-way 10-shots and 10-way 10-shots, where Hyp represents Hyperbolic and Euc represnts Euclidean distance metric in Algorithm 1. Additionally, we have provided the parameters and performed Forward/Backward pass evaluations for the methods their source code was publicly available, using PyTorch-summary for a batch size of 150 indicating a 5-way 10-shot 20-query setting. The quantitative results (accuracies in %) are represented in three styles: (1) **best**, (2) **second best**, (3) third best and ± represents mean and standard deviation of 6 experiments with a different random seed.

Method	Num Points	5-way		10-ways		#Params	F/B pass
		10-shots	20-shots	10-shots	20-shots		
3D-GAN [1]	1024	55.80 ± 10.68	65.80 ± 09.90	40.25 ± 06.49	48.35 ± 05.59	-	-
Latent-GAN [36]	1024	41.60 ± 16.91	46.20 ± 19.68	32.90 ± 09.16	25.45 ± 09.90	-	-
PointCapsNet [41]	1024	42.30 ± 17.37	53.00 ± 18.72	38.00 ± 14.30	27.15 ± 14.86	2.15M	39GB
FoldingNet [38]	1024	33.40 ± 13.11	35.80 ± 18.19	18.55 ± 06.49	15.44 ± 06.82	0.67M	5.7GB
PointNet++ [25]	1024	38.53 ± 15.98	42.39 ± 14.18	23.05 ± 06.97	18.80 ± 05.41	1.48M	149GB
PointCNN [16]	1024	65.41 ± 08.92	68.64 ± 07.00	46.60 ± 04.84	49.95 ± 07.22	-	-
PointNet [25]	1024	51.97 ± 12.17	57.81 ± 15.45	46.60 ± 13.54	35.20 ± 15.25	3.47M	8.5GB
DGCNN [34]	1024	31.60 ± 08.97	40.80 ± 14.60	19.85 ± 06.45	16.85 ± 04.83	1.82M	53GB
SS-FSL (PointNet) [28]	1024	63.20 ± 10.72	68.90 ± 09.41	49.15 ± 06.09	50.10 ± 05.00	3.47M	8.5GB
SS-FSL (DGCNN) [28]	1024	60.00 ± 08.87	65.70 ± 08.37	48.50 ± 05.63	53.00 ± 04.08	1.82M	53GB
Enrich-Features [8]	1024	76.69 ± NA	**85.76 ± NA**	68.76 ± NA	**80.72 ± NA**	-	-
GPr-Net (Euc)	1024	74.37 ± 02.00	75.12 ± 02.08	62.14 ± 01.91	63.43 ± 02.05	**1.24K**	**50KB**
GPr-Net (Hyp)	1024	80.40 ± 00.55	81.99 ± 00.91	70.42 ± 01.80	72.83 ± 01.78	**1.24K**	**50KB**
GPr-Net (Euc)	512	74.04 ± 02.33	74.98 ± 02.42	62.31 ± 02.01	63.33 ± 02.21	**1.24K**	**50KB**
GPr-Net (Hyp)	512	**81.13 ± 01.51**	**82.71 ± 01.28**	**71.59 ± 01.16**	73.78 ± 01.99	**1.24K**	**50KB**

ing $D_{train} \cap D_{test} = \emptyset$ such that, for each experiment, we randomly sample $\mathcal{T}$ categories of data to form D_{train} and rest categories without replacement form D_{test} this meta-training strategy aids in understanding true robustness of the proposed method in FSL. For our experiments, we randomly sampled $\mathcal{T} = 24$ categories for training and 16 for testing in ModelNet40 as suggested by [8]. Note that the results of all other networks except Enrich-Features [8] in Table 1 are derived from SS-FSL [28].

4.2. Comparison with State-of-the-art Methods

Our novel GPr-Net not only outperforms existing methods that rely on data-heavy pre-training tasks like SS-FSL [28] or complex feature extractors [8] [39] in terms of accuracy, but it also operates much faster. To demonstrate this, we compared our proposed backbone architecture to several open-source backbones and evaluated parameters, few-shot classification accuracy, and parameters on the ModelNet40 dataset [37], as suggested by PointMLP [20]. For example, SS-FSL (DGCNN) is a cumbersome model that achieves impressive results with 1.82M parameters and a forward/backward pass of 53GB, as shown in Table 1. In contrast, our GPr-Net achieves state-of-the-art FSL accuracy on point clouds while maintaining only 1.24K parameters, which is 280 times less than SS-FSL, and a forward/backward pass of 50KB. This is particularly essential for applications like robotics, self-driving cars, and others that require deploying these models efficiently.

Our results with Hyperbolic-metric, presented in Table 1,

demonstrate that our method surpasses Enrich-Features [8] by 5% for 5-way 10-shots and 3% for 10-way 10-shots. However, when compared to SS-FSL [28], our method achieves significant improvements of 18% for 5-way 10-shots, 14% for 5-way 20-shots, 22% for 10-way 10-shots, and 21% for 10-way 20-shots. These results highlight the efficacy of our approach in addressing the challenging problem of few-shot learning for point clouds. *Notice that we achieve this by only 50 epochs, 4 training episodes and only 512 points.* Remarkably, our method achieves the smallest standard deviation across 6 experiments with different random seeds, indicating its robustness and lack of bias towards a particular category.

4.3. Ablation Studies

This section presents ablation studies to analyse the impact of different designs of our proposed module on few-shot point cloud classification.

Significance Laplace Vectors $\vec{L}$ is demonstrated in the left of Figure 4. The results show that the GPr-Net with $\vec{L}$ outperforms the one without in all cases of point density variation, as reported by the mean and standard deviation accuracy of 6 experiments for 5-way 10-shot tasks on the Hyperbolic variant of GPr-Net. The experiments were conducted with 128, 256, 512, and 1024 points, and the superior performance of the classifier with $\vec{L}$ suggests the significance of extracting geometric features using Laplace vectors for effective few-shot learning on point clouds.
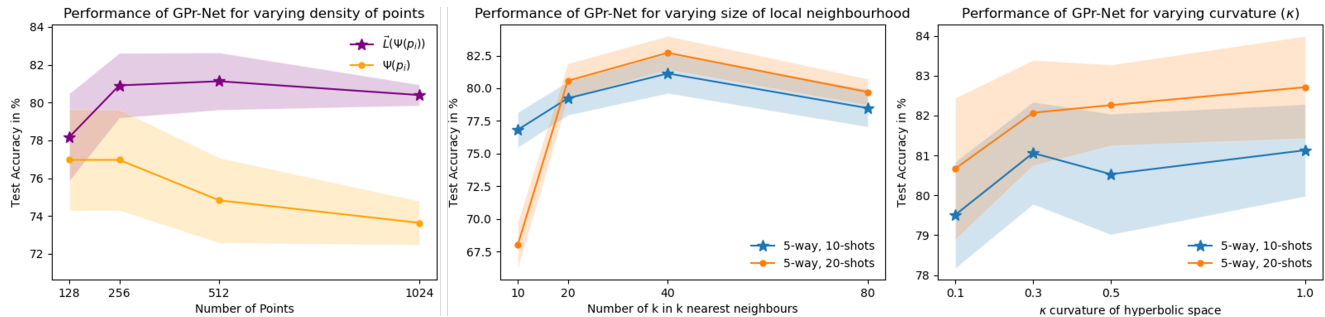
Figure 4. The **Left**: compares the few-shot accuracy of GPr-Net with and without Laplace vectors $\vec{L}$ as the point density varies. The results indicate that incorporating Laplace vectors leads to superior performance. The **Center**: impact of k-Nearest Neighbors (k-NN) is analyzed by varying k on GPr-Net with Laplace vectors. It is observed that k=40 yields the best results for 5-way 10 and 20-shots. Finally, the **Right**: investigates the effect of hyperbolic curvature on few-shot accuracy. GPr-Net with $\kappa \to 1$ hyperbolic metric outperforms the Euclidean metric with $\kappa \to 0$ for both 5-way 10 and 20-shots. Further details are provided in Section 4.3.

Local Neighbourhood Size justifies the effectiveness of the Laplace vectors in our proposed GPr-Net for few-shot learning on point clouds, we conducted additional experiments to determine the appropriate value of k in k-NN for computing Laplace vectors. We aim to determine the best value of k that describes local topological changes such as edges and corners, which are largely dependent on the size of the local neighborhood. In the center of Figure 4, we present our findings on the need for selecting an appropriate value of k. We report the mean and standard deviation accuracy of 6 experiments for 5-way 10-shot and 5-way 20-shot tasks on the Hyperbolic variant of GPr-Net. The experiments were conducted with k=10,20,40 and 80 for 512 points in a point cloud. Our findings suggest that k=40 is the optimal value for 512-point density in both 10 and 20-shot cases.

Influence of Curvature κ plays a critical role in the performance of hyperbolic metrics for point cloud few-shot learning. The negative curvature allows for more efficient space utilization, increasing the ability to distinguish between points. We perform experiments that show an increase in curvature κ in Eq. 3 results in improved performance for few-shot learning tasks, as the embeddings are better able to capture the similarities and differences between point clouds. However, as κ approaches zero, the hyperbolic space approaches a Euclidean space, and the benefits of the negative curvature are lost as Depicted in the right of Figure 4. Therefore, it is essential to find the optimal curvature for a given few-shot learning task to achieve the best results. Our findings indicate that $\kappa = 1.0$ is the best suited for 5-way 10-shots and 5-way 20-shots tasks for 512 points with Laplace vectors and k = 40.

Performance Efficiency of proposed method GPr-Net is conducted and we compare with CGNN [17] and CIA [39] in a 5-way 1-shot and 5-way 5-shot learning setting. Our method was trained for only 4 episodes, and we observed competitive performance with the other two models, as

Table 2. The comparison of our proposed GPr-Net with state-of-the-art methods CGNN [17] and CIA [39] in a 5-way 1-shot and 5-way 5-shot learning setting. The results demonstrate the competitive performance of our method with significantly fewer parameters and faster training speed. Although CIA [39] achieves state-of-the-art accuracy in both settings, our method achieved the second-best result, trained with only 4 episodes.

Method	5-way	
	1-shot	5-shots
CGNN [17]	-	$76.85 \pm$ NA
CIA [39]	$\mathbf{75.70 \pm 0.74}$	$\mathbf{87.15 \pm 0.47}$
GPr-Net (Euc)	64.12 ± 0.73	74.56 ± 1.03
GPr-Net (Hyp)	$\underline{67.91 \pm 1.07}$	$\underline{79.09 \pm 0.97}$

shown in Table 2. CGNN [17] and CIA [39] also aim to learn representations and relations between prototypes and query features by utilizing feature-matching methods such as graph neural networks or self-attention mechanisms, respectively. Although CIA [39] achieves state-of-the-art accuracy in 5-way, 1-shot and 5-shot settings, we still achieved the second-best result with significantly fewer parameters and faster training speed. Unfortunately, since the code for CIA [39] was not open-source, we were unable to make a direct comparison in terms of speed.

4.4. Embedding Visualization

In the context of a *5way-10shot-50query* setting on the ModelNet40 [37] dataset, we present a visualization of the features generated by our proposed GPr-Net. Specifically, we compare the features obtained using the hyperbolic/Poincaré metric with those obtained using the Euclidean metric. The left-hand side of Figure 5 corresponds to the former, while the right-hand side depicts the latter.

It is worth noting that there exists a significant distribution shift between the support and query features in this set-

ting. However, our proposed hyperbolic metric helps to mitigate the challenge of inter-class ambiguity by causing the features to move towards the boundary of the Poincaré manifold. This allows for better differentiation between classes, leading to more accurate predicted labels.
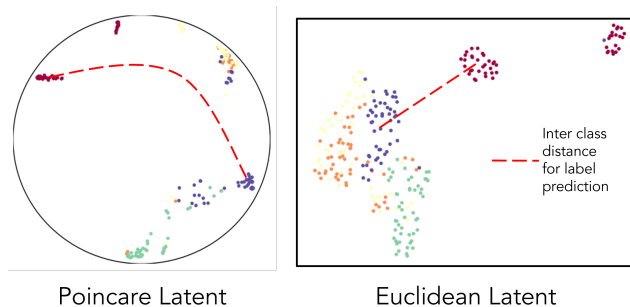

Figure 5. A visualization of the embeddings learned for the few–shot task. **Left**: Our 5-way task on Modelnet40 with Poincaré metric. **Right**: Our 5-way task on Modelnet40 with Euclidean metric. The two-dimensional projection was computed using the UMAP [21].

4.5. Limitations

Notwithstanding the promising results achieved in our study, we acknowledge certain limitations that need to be addressed. First, due to the nature of our approach incorporating Hyperbolic Projection, it is currently hard to perform part segmentation on per-point embeddings. This is a known limitation also faced by other works in the field, such as HyCoRe [22]. As a result, we were unable to apply our method to datasets that require part segmentation, limiting the scope of our study. Another limitation of our method is susceptibility towards noise due to the use of k-NN for local grouping. Therefore, we exclude certain real-world datasets, such as the Sydney and ScanObjectNN datasets, which contain large amounts of noise.

Despite these limitations, we believe that our study offers valuable insights into few-shot learning in point cloud settings. We hope that our findings will inspire further research to address these limitations and lead to the development of more robust and effective methods for few-shot learning in point clouds.

5. Conclusions

In this work, we have proposed a new perspective on point cloud few-shot learning by challenging the assumption of complex network designs and training strategies. Our proposed lightweight Geometric Prototypical Network, GPr-Net, leverages simple yet effective geometric signatures, including the Intrinsic Geometry Interpreter and Laplace vectors, to efficiently capture the intrinsic topology of point clouds and achieve superior performance on the ModelNet40 dataset. Additionally, employing a hyperbolic/poincaré metric for learning 3D few-shot features further improves the effectiveness of our approach. Our experimental results demonstrate that GPr-Net outperforms state-of-the-art point cloud few-shot learning techniques with 5% higher accuracy and $170\times$ fewer parameters than all existing works.

6. Broader Impact

The focus of this study is on local geometric features and selecting appropriate metric space that can enhance the performance of few-shot learning in point cloud settings, even when labeled data is limited.

Point cloud few-shot learning has the potential to impact a wide range of fields. For example, robotics can accelerate the training of robots to recognize new objects with a small number of labeled examples, resulting in faster and more cost-effective learning. In architecture and engineering, it can facilitate the design of complex structures with limited labeled data by allowing shape analysis and thermal analysis. Additionally, it can assist in environmental studies, such as land surveying, by making it easier to classify and analyze 3D point cloud data with minimal labels.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning Representations and Generative Models for 3D Point Clouds, 2018. 1, 6

[2] Tejas Anvekar, Ramesh Ashok Tabib, Dikshit Hegde, and Uma Mudengudi. VG-VAE: A Venatus Geometry Point-Cloud Variational Auto-Encoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2978–2985, June 2022. 2, 3, 4

[3] J. Baxter. A Model of Inductive Bias Learning. *Journal of Artificial Intelligence Research*, 12:149–198, mar 2000. 2

[4] Dena Bazazian and Dhananjay Nahata. DCG-net: Dynamic capsule graph convolutional network for point clouds. *IEEE Access*, 8:188056–188067, 2020. 2

[5] Dena Bazazian and M Eulàlia Parés. EDC-Net: Edge detection capsule network for 3D point clouds. *Applied Sciences*, 11(4):1833, 2021. 2

[6] Guneet S. Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A Baseline for Few-Shot Image Classification. *CoRR*, abs/1909.02729, 2019. 3

[7] Henghui Ding, Hui Zhang, and Xudong Jiang. Self-regularized prototypical network for few-shot semantic segmentation. *Pattern Recognition*, 133:109018, 2023. 2

[8] Hengxin Feng, Weifeng Liu, Yanjiang Wang, and Baodi Liu. Enrich Features for Few-Shot Point Cloud Classification. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2285–2289, 2022. 2, 3, 4, 5, 6

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 06–11 Aug 2017. 2

[10] Octavian Ganea, Gary Becigneul, and Thomas Hofmann. Hyperbolic Neural Networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 3

[11] Xinyu Jiang, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Duoqian Miao. Few-shot object detection via improved classification features. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5386–5395, January 2023. 2

[12] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic Image Embeddings, 2020. 2, 3, 4, 5

[13] Max Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian Optimization in PyTorch, 2020. 5

[14] Dirk P Kroese, Sergey Porotsky, and Reuven Y Rubinstein. The cross-entropy method for continuous multi-extremal optimization. *Methodology and Computing in Applied Probability*, 8:383–407, 2006. 5

[15] Ruihui Li, Xianzhi Li, Pheng-Ann Heng, and Chi-Wing Fu. PointAugment: An Auto-Augmentation Framework for Point Cloud Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1

[16] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. PointCNN: Convolution On X-Transformed Points. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 2, 6

[17] Yangfan Li, Cen Chen, Weiquan Yan, Zhongyao Cheng, Hui Li Tan, and Wenjie Zhang. Cascade Graph Neural Networks for Few-Shot Learning on Point Clouds. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, 2023. 7

[18] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep Continuous Fusion for Multi-Sensor 3D Object Detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1

[19] Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic Graph Neural Networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 3

[20] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking Network Design and Local Geometry in Point Cloud: A Simple Residual MLP Framework. In *International Conference on Learning Representations*, 2022. 2, 6

[21] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018. 8

[22] Antonio Montanaro, Diego Valsesia, and Enrico Magli. Rethinking the compositionality of point clouds through regularization in the hyperbolic space. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 8

[23] Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic Deep Neural Networks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10023–10044, 2022. 3

[24] Charles Ruizhongtai Qi, Hao Su, Mo Kaichun, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017. 1, 2

[25] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 1, 2, 6

[26] 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, 2008. 1, 2

[27] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, Mihai Dolha, and Michael Beetz. Towards 3D Point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927–941, 2008. Semantic Knowledge in Robotics. 1

[28] Charu Sharma and Manohar Kaul. Self-Supervised Few-Shot Learning on Point Clouds. *CoRR*, abs/2009.14168, 2020. 2, 5, 6

[29] Shivanand Venkanna Sheshappanavar, Vinit Veerendraveer Singh, and Chandra Kambhamettu. PatchAugment: Local Neighborhood Augmentation in Point Cloud Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2118–2127, October 2021. 1

[30] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical Networks for Few-shot Learning. *CoRR*, abs/1703.05175, 2017. 2, 4, 5

[31] Stefan Stojanov, Anh Thai, and James M. Rehg. Using Shape to Categorize: Low-Shot Learning with an Explicit Shape Bias. 2021. 2

[32] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proc. ICCV*, 2015. 1, 2

[33] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip Torr, and Timothy Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. pages 1199–1208, 06 2018. 2, 5

[34] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics (TOG)*, 2019. 2, 6

[35] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona.

Caltech-UCSD Birds 200. Technical Report CNS-TR-201, Caltech, 2010. 3

[36] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. *CoRR*, abs/1610.07584, 2016. 6

[37] Zhirong Wu, Shuran Song, Aditya Khosla, Xiaoou Tang, and Jianxiong Xiao. 3D ShapeNets for 2.5D Object Recognition and Next-Best-View Prediction. *CoRR*, abs/1406.5670, 2014. 5, 6, 7

[38] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation. pages 206–215, 06 2018. 6

[39] Chuangguan Ye, Hongyuan Zhu, Yongbin Liao, Yanggang Zhang, Tao Chen, and Jiayuan Fan. What Makes for Effective Few-shot Point Cloud Classification? In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 267–276, 2022. 2, 6, 7

[40] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Few-shot 3D Point Cloud Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 5

[41] Yongheng Zhao, Tolga Birdal, Haowen Deng, and Federico Tombari. 3D Point Capsule Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1009–1018, 2019. 1, 6

[42] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3D object recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 689–696, 2009. 1, 2

[43] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. pages 4490–4499, 06 2018. 2

[44] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3357–3364, 2017. 1