2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Hyperbolic Image-and-Pointcloud Contrastive Learning for 3D Classification

Naiwen Hu†, Haozhe Cheng†, Yifan Xie, Pengcheng Shi and Jihua Zhu*

***Abstract*— 3D contrastive representation learning has exhibited remarkable efficacy across various downstream tasks. However, existing contrastive learning paradigms based on cosine similarity fail to deeply explore the potential intra-modal hierarchical and cross-modal semantic correlations about multi-modal data in Euclidean space. In response, we seek solutions in hyperbolic space and propose a hyperbolic image-and-pointcloud contrastive learning method (HyperIPC). For the intra-modal branch, we rely on the intrinsic geometric structure to explore the hyperbolic embedding representation of point cloud to capture invariant features. For the cross-modal branch, we leverage images to guide the point cloud in establishing strong semantic hierarchical correlations. Empirical experiments underscore the outstanding classification performance of HyperIPC. Notably, HyperIPC enhances object classification results by 2.8% and few-shot classification outcomes by 5.9% on ScanObjectNN compared to the baseline. Furthermore, ablation studies and confirmatory testing validate the rationality of HyperIPC's parameter settings and the effectiveness of its submodules.**

I. INTRODUCTION

With the popularity of Foundation Model, self-supervised representation learning has achieved great success in the fields of natural language processing (NLP) [1],[2], computer vision [3],[4], video signals [5], and multi-modality [6],[7]. These methods use extreme amounts of data in the pre-training stage to obtain powerful representations for downstream tasks. In the 3D vision field, data collection and annotation are time-consuming and labor-intensive compared to 2D vision and NLP. Considering the issues of data scarcity and imbalance, it is challenging to obtain high-quality representations using self-supervised representation learning methods with limited data.

Various 3D self-supervised representation learning methods have been developed. Contrastive learning encourages the representations of the same category to be closer and the representations of different categories to be farther apart [8],[9]. Recently, many contrastive learning methods [10],[11] have been proposed to deal with point cloud through various strategies for constructing positive-negative sample pairs.

It is well known that an image conveys various semantic information. Even for the same category of objects, humans can reason about their relative details and organize these concepts into a meaningful visual semantic hierarchy. As shown in Fig. 1, for the category of "Airplane", the point cloud located at a higher level has a more abstract description.

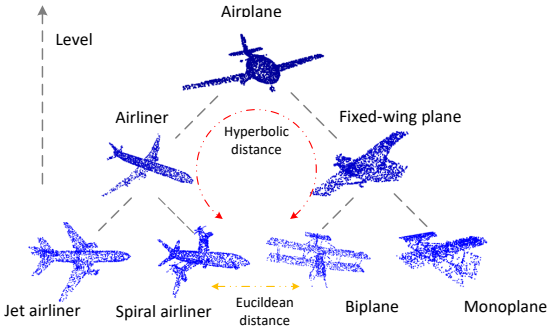

Fig. 1: **Illustration of semantic hierarchy in hyperbolic space.** "Airplane" can be organized into a tree-like structure according to their flying modes and other semantic information, where the lower the level, the more detailed the object's description. The point cloud features located at different nodes of the same level should pass through the root node (red line) when calculating distance. However, the distance in Euclidean space is defined according to cosine similarity (yellow line).

Unfortunately, current 3D self-supervised representation learning methods embed the point cloud in the Euclidean space using the same distance metric, which cannot capture the semantic hierarchy of the data. This may cause potential issues, as illustrated in Fig. 1, the specific concept ("Biplane") is closer to other specific concepts ("Spiral airliners") rather than the generic concept ("Airplane"). As a space with a constant negative curvature, the volume of hyperbolic space grows exponentially concerning the radius. Thus, the hyperbolic space can embed tree-like graphs with minimal distortion. To mine the latent semantic hierarchy in the point cloud, this property of hyperbolic space motivates us to embed point cloud representations into hyperbolic space.

In this work, we propose a simple and effective Hyperbolic Image-and-Pointcloud Contrastive Learning (HyperIPC) model. By projecting latent vectors to the hyperbolic space, we can efficiently extract the intrinsic semantic hierarchy of unlabeled data. We first map point cloud features from Euclidean space to the hyperbolic space and use the distance defined in the hyperbolic space for contrastive learning. Then, we compute their parent node in hyperbolic space, which is closer to the origin and can be regarded as a more abstract representation of the two different views. To leverage the semantic hierarchy information inherent in images, we employ a pre-trained image encoder to extract

†:equal contribute. *:corresponding author(zhujh@xjtu.edu.cn). The authors are with School of Software Engineering, Xi'an Jiaotong University, Xi'an710000, China and Shaanxi Joint Key Laboratory for Artifact Intelligence, China.

2D information from rendered images, then map vectors to hyperbolic space for contrastive learning with the parent nodes. Moreover, to ensure the comprehensive exploitation of hyperbolic space, we optimize the nodes in Poincaré disk according to their level information.

The main contributions of our approach can be summarized as follows:

- We propose HyperIPC, a simple and effective hyperbolic contrastive learning framework for self-supervised 3D point cloud pre-training incluing the intra-modal and cross-modal hyperbolic contrastive learning.
- We introduce the 2D VIT pre-trained by CLIP, which leverages the 2D knowledge to guide the point cloud to construct hierarchy in hyperbolic space.
- HyperIPC achieves state-of-the-art performance for contrastive learning on various downstream tasks, which indicates contrastive learning with hyperbolic distance outperforms the contrastive learning methods in the Euclidean space.

II. RELATED WORK

A. Contrastive Learning in Point Cloud

Contrastive learning leverages optimized contrastive loss to encourage augmentation of the same input to produce more comparable representations. In 3D vision, there are predominantly two categories of contrastive learning methods: object-level and scene-level. The former captures the global representation of the point cloud by treating the whole point cloud as an object [12],[13]. For example, Du et al.[14] use self-similar patches within a single point cloud to facilitate semantic understanding. The latter focuses more on the interaction between point cloud and its scene [10],[15]. In contrast to previous 3D contrastive learning methods, our HyperIPC extends the contrastive loss to the hyperbolic space.

B. Hyperbolic Embedding

With the development of deep learning, Euclidean space has become the standard manifold [16]. At the same time, hyperbolic space has been successfully applied to NLP [17] due to the inherently hierarchical nature of language. HCNN [18] further extend deep neural network modules in the hyperbolic space. As a result, hyperbolic space has achieved success in image representation [19],[20]. EDGCNet [21] proposes a dynamic hyperbric graph convolution module for 3D point cloud segmentation. Chen et al.[22] propose a self-supervised learning method based on hyperbolic homotopy embedding to explore the nonlinear relationship of behavior trajectories. HIE [23] leverages the distance between data nodes and the origin node in hyperbolic space, deriving hierarchical information to optimize the existing hyperbolic models. In the 3D vision, HyCoRe [24] captures 3D part-whole hierarchy in supervised learning. In this paper, our HyperIPC aims to capture the semantic hierarchy among 3D objects in self-supervised representation learning.

III. METHOD

The schematic overview of the proposed method is depicted in Fig. 2. In this section, we first introduce how to obtain the hierarchical structure in the hyperbolic space. Then, we will discuss our overall framework diagram and describe the loss functions.

A. Hyperbolic Geometric Embedding

We introduce hyperbolic space as the latent space to extract the latent semantic hierarchical information of the point cloud. In contrast to the Euclidean space with zero curvature, the n-dimensional hyperbolic space H^n is a Riemannian manifold with constant negative curvature. The Poincaré, Lorentz, and Klein models are commonly used, equivalent representations of hyperbolic space, which can be interconverted and are suitable for different tasks [25]. We use the Poincaré disk model $\left(\mathbb{D}_c^n, g^{\mathbb{D}}\right)$ because it can maintain numerical stability in the gradient-based learning process. The manifold $\mathbb{D}_c^n = \left\{x \in \mathbb{R}^n : c\|x\|^2 < 1, c \geq 0\right\}$ is equipped with a Riemannian metric $g^{\mathbb{D}} = \lambda_c^x g^E$, where g^E is the metric tensor, $\lambda_c^x = \frac{2}{1-c\|x\|^2}$ is the conformal factor depending on the curvature c and the position of the calculated point x in the Poincaré disk. The metric of the points closer to the edge of the disk is scaled more by the conformal factor.

Conventional data operations are not applicable to the hyperbolic space, so we need to expand the operations in the hyperbolic space. Let $\|.\|$ be the Euclidean norm and $\langle ., . \rangle$ represent the Minkowski inner product. Given a pair of $\mathbf{x}, \mathbf{y} \in \mathbb{D}_c^n$, the addition operation is defined as:

$$\mathbf{x} \oplus_c \mathbf{y} = \frac{\left(1 + 2c\langle \mathbf{x}, \mathbf{y} \rangle + c\|\mathbf{y}\|^2\right)\mathbf{x} + \left(1 - c\|\mathbf{x}\|^2\right)\mathbf{y}}{1 + 2c\langle \mathbf{x}, \mathbf{y} \rangle + c^2\|\mathbf{x}\|^2\|\mathbf{y}\|^2}. \quad (1)$$

The distance between $\mathbf{x}, \mathbf{y} \in \mathbb{D}_c^n$ in the hyperbolic space is defined as:

$$D_{hyp}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{c}} arctanh\left(\sqrt{c}\,\|-\mathbf{x} \oplus_c \mathbf{y}\|\right), \quad (2)$$

in the above formulas, when the curvature c approaches zero, the formulas for addition and distance become identical to those in the traditional Euclidean space.

In the hyperbolic space, the geodesic is a generalization of the shortest path between two points or planes. As the curvature decreases, the distance between two points increases, and the geodesic is closer to the boundary [26]. The midpoint of the geodesic between two points in hyperbolic space tends to be closer to the origin, resembling the concept of the tree [16],[27]. This midpoint can be considered a more abstract parent node. This unique feature in the hyperbolic space can be reflected in Fig. 2. The point closer to the edge represents more specific categories, while closer to the origin represents more inductive instances. Given n hyperbolic node vectors $(\mathbf{z}_1, \cdots, \mathbf{z}_n)$, the midpoint is computed in gyrovector space, which is given by:

$$\mathbf{z}_{mid} = \frac{1}{2} \oplus_c \left(\frac{\sum_{i=1}^n \lambda_c^x \mathbf{z}_i}{\sum_{i=1}^n \left(\lambda_c^x - 1\right)}\right). \quad (3)$$

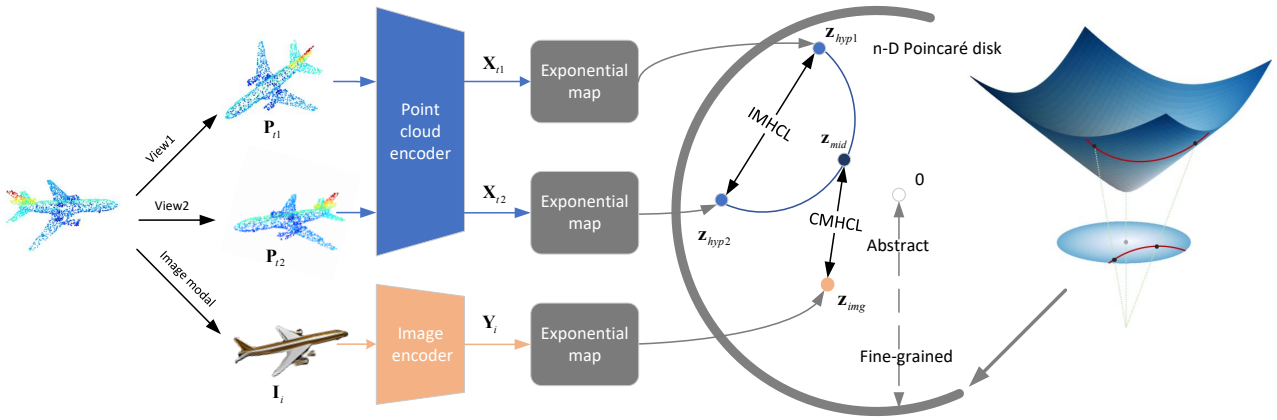

Fig. 2: **Our proposed model architecture.** Point cloud branch: Intra-Modal Hyperbolic Contrastive Learning (IMHCL) makes the modal learn the invariance between two augmented point cloud. Image branch: Cross-Modal Hyperbolic Contrastive Learning (CMHCL) leverages rendered image guide point cloud to establish a hierarchical structure.

To map the embeddings to the hyperbolic space, we need to define a mapping relation from the Euclidean space to the Poincaré disk called the exponential map. The hyperbolic manifold $\mathbb{D}_c^n$ at x has first order linear approximation tangent space $\mathcal{T}_{\mathbf{x}}\mathbb{D}_c^n \cong \mathbb{R}^n$, where $\mathcal{T}_{\mathbf{x}}\mathbb{D}_c^n = \left\{\mathbf{v} \in \mathbb{R}^d : \langle \mathbf{v}, \mathbf{x} \rangle = 0\right\}$. The exponential map is defined as:

$$\exp_{\mathbf{x}}^c(\mathbf{v}) = \mathbf{x} \oplus_c \left(\tanh\left(\sqrt{c} \frac{\lambda_{\mathbf{x}}^c \|\mathbf{v}\|}{2} \right) \frac{\mathbf{v}}{\sqrt{c}\|\mathbf{v}\|} \right). \quad (4)$$

B. Model Architecture

Due to the advantages of the hyperbolic space, our model consists of two branches: Intra-Modal Hyperbolic Contrastive Learning (IMHCL) and Cross-Modal Hyperbolic Contrastive Learning (CMHCL). These two branches obtain the semantic hierarchical structure of the point cloud from intra-modal and cross-modal perspectives in the hyperbolic space. Given the sample data $\mathcal{D} = \{(\mathbf{P}_i, \mathbf{I}_i)\}_{i=1}^{|\mathcal{D}|}$, where $\mathbf{P}_i$ denotes the 3D point cloud and $\mathbf{I}_i$ denotes the corresponding 2D image rendered from a random viewpoint. To enhance the discriminative ability of the point cloud encoder, we first apply IMHCL. Specifically, for a given point cloud sample $\mathbf{P}_i$, we apply random sampling transformations such as rotation, scaling, and translation to form two augmented point cloud $\mathbf{P}_{t1}$ and $\mathbf{P}_{t2}$. We use a shared-weight point cloud encoder to extract global features $\mathbf{X}_{t1}$ and $\mathbf{X}_{t2}$. Instead of regularizing the output in the Euclidean space, we use the exponential map Eq.(4) to map the features from the Euclidean space to the hyperbolic space to obtain $\mathbf{z}_{hyp1}$ and $\mathbf{z}_{hyp2}$. Contrastive learning methods in Euclidean space define distance using squared Euclidean distance or cosine similarity. In hyperbolic space, we use Eq.(2) to define the distance for contrastive learning.

To prevent confusion, we first ignore the domain identifier $\mathbf{z}_{hyp1}$ and $\mathbf{z}_{hyp2}$. Given a positive sample pair (i, j) and its representation $(\mathbf{z}_i, \mathbf{z}_j)$ in hyperbolic space, we define the hyperbolic contrastive learning loss function as:

$$l(\mathbf{z}_i, \mathbf{z}_j) = -\log \frac{\exp\left(-D_{hyp}\left(\mathbf{z}_i, \mathbf{z}_j\right)/\tau\right)}{\sum_{k=1, k\neq i}^{N} \exp\left(-D_{hyp}\left(\mathbf{z}_i, \mathbf{z}_k\right)/\tau\right)}, \quad (5)$$

where D_{hyp} is the distance calculated by hyperbolic space, τ is the temperature coefficient, and N represents the number of samples in a batch. The loss is calculated by all the positive samples (i, j) and (j, i).

$$\mathcal{L}(\mathbf{z}_i, \mathbf{z}_j) = \frac{1}{2N} \sum_{i=1}^{N} [l(\mathbf{z}_i, \mathbf{z}_j) + l(\mathbf{z}_j, \mathbf{z}_i)]. \quad (6)$$

After obtaining embeddings of the same sample in hyperbolic space, the mean of the two embeddings $\mathbf{z}_{mid}$ is calculated using Eq.(3). This mean point locate at the midpoint of the geodesic line between $\mathbf{z}_{hyp1}$ and $\mathbf{z}_{hyp2}$ is closer to the origin. This property is crucial for constructing a semantic hierarchy in hyperbolic space, similar to a tree structure where the mean of two leaf nodes represents a more general parent node rather than another leaf node.

Ermolov et al.[20] demonstrated that images have hierarchical information in hyperbolic space. We introduce the auxiliary CMHCL to guide the point cloud to establish a semantic hierarchy in hyperbolic space. During the model initialization process, the embeddings obtained by the image encoder are inaccurate. To prevent inaccurate 2D features guiding point cloud from being incorrectly embedded in hyperbolic space, the 2D pre-trained model is used to initialize the image encoder. We first use the visual encoder to obtain the embedding $\mathbf{Y}_i$ for the 2D image $\mathbf{I}_i$ of point cloud $\mathbf{P}_i$. Then, we apply the exponential map to project this Euclidean latent code $\mathbf{Y}_i$ into hyperbolic space to obtain $\mathbf{z}_{img}$. The goal of CMHCL is to maximize the similarity of $\mathbf{z}_{mid}$ to corresponding $\mathbf{z}_{img}$. In summary, CMHCL captures the image-pointcloud hierarchy to improve model discrimination.

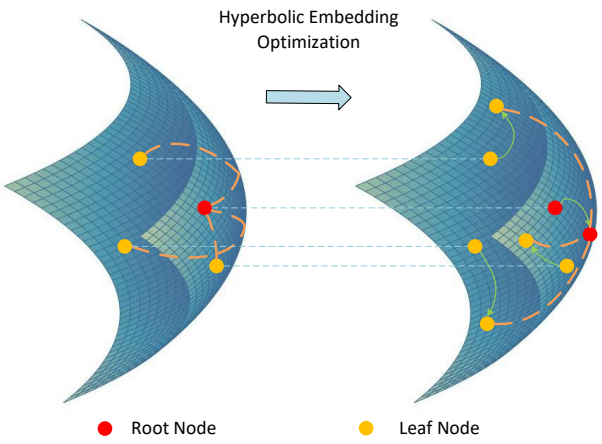

Fig. 3: **Illustration of the hyperbolic embedding optimization.** Before optimization (Left), the root node deviates from the center of hyperbolic space and the leaf nodes are far from the boundary of the Poincaré disk. After hyperbolic optimization (Right), the root node of the data is aligned with the origin of the hyperbolic space, and the leaf nodes make full use of the characteristics of the hyperbolic space to disperse as much as possible.

C. Hyperbolic Embedding Optimization

Since the hyperbolic space grows exponentially, the regions far from the origin are more spacious. The Leaf nodes in the tree structure occupy the majority and are as far from the origin as possible. Therefore, we hope that the point cloud embedding root node is optimized to the highest level, and the overall point cloud embedding should fully use the hyperbolic space's expansibility to disperse as much as possible. To solve the above problem, the first step is to identify the root node of the data, align it with the origin of the hyperbolic space, and then optimize the node according to their level information.

We first define the hyperbolic embedding center as the root node $\mathbf{z}_c$ by Eq.(3). This node comes from the hyperbolic embedding and can be regarded as a super node connecting all subtrees. Then, we employ the root alignment strategy as defined:

$$\overline{\mathbf{z}} = \mathbf{z}_{mid} \oplus_c (-\mathbf{z}_c). \tag{7}$$

To efficiently access level information and guide hierarchical learning. We align the hyperbolic embedding center with the origin of the hyperbolic space, it reflects the relative distance between the leaf node and the root node, indicating its hierarchical level.

$$\mathbf{z}_{hdo} = \frac{1}{|N|} \sum_{i \in N} w_i D_{hyp}(\overline{\mathbf{z}}_i, \mathbf{o}), \tag{8}$$

in Eq.(8), w_i indicates the node level in hyperbolic space which is computed by $\sigma(D_{hyp}(\overline{\mathbf{z}}_i, \mathbf{o}))$, the σ is sigmoid function. The loss function is:

$$\mathcal{L}_{dho} = \sigma(-\mathbf{z}_{hdo}). \tag{9}$$

By optimizing the loss function, the high-level nodes close to the origin are assigned lower weights to prevent them from being pushed away. The low-level nodes far from the origin are assigned larger weights to help them reach correct positions in hyperbolic space.

D. Overall Objective

The overall loss function of the model consists of three parts: intra-modal hyperbolic contrastive loss, cross-modal hyperbolic contrastive loss, and hyperbolic embedding optimization loss, as follows:

$$\mathcal{L} = \mathcal{L}(\mathbf{z}_{hyp1}, \mathbf{z}_{hyp2}) + \mathcal{L}(\mathbf{z}_{mid}, \mathbf{z}_{img}) + \lambda \mathcal{L}_{dho}, \tag{10}$$

where λ is a hyperparameter.

IV. Experiments

A. Pre-training

Dataset. Our model is pretrained on ShapeNet[28], with over 50,000 CAD models in 55 categories. For given point cloud, a 2D image is randomly selected from the rendered images, captured from various viewpoint[29]. Each point cloud consists of 2,048 points with only x, y, z coordinate, and the corresponding rendered image is resized to 224 × 224 pixels. Augmentation operations such as rotation and cropping are applied to increase the diversity of rendered images from random viewpoints.

Implementation Details. For a fair comparison with previous work, we apply DGCNN [30] as point cloud feature extractor, which exploits local geometric structures by constructing a local neighborhood graph and applying convolution-like operations on the edges connecting neighboring pairs of points. As for the image encoder, we utilize VIT-S [31] that divides an image into patches, embeds them, and processes these embeddings through transformer layers to capture global image features. In addition, we use a two-layer MLP (384-128) as the projection head, and finally produce 128-dimensional feature projected in the hyperbolic space. The contrastive learning utilizes a curvature parameter $c = 0.1$, temperature $\tau = 0.2$ and the hyperbolic embedding optimization incorporates $\lambda = 0.01$.

For the image encoder, we use the AdamW optimizer [32] with a learning rate value of 3×10^{-5}, a weight decay value of 0.01. We use the AdamW optimizer [32] for the image encoder, with a learning rate of 1×10^{-3} and a weight decay of 1×10^{-4}. All experiments are conducted on a single NVIDIA 3090Ti GPU with 100 epochs. After pretraining, we discard the image encoder and two projection heads. All downstream tasks are performed on the point cloud encoder.

B. Downstream Tasks

We evaluate the transferability of HyperIPC on two widely used downstream tasks in point cloud representation learning: (i) 3D object classification (synthetic and real-world), (ii) Few-shot object classification (synthetic and real-world)

3D Object Classfication. We demonstrate the generalizability of our approach in learning 3D shape representation

TABLE I: **Comparison of ModelNet40 and ScanObjectNN linear classification results with previous self-supervised methods.** A linear classifier is fit onto the training split using the pretrained model and overall accuracy for classification in test split is reported. The dagger (†) denotes that the model was reproduced using DGCNN backbone.

Method	ModelNet40	ScanObjectNN
3D-GAN[33]	83.3	-
Latent-GAN[27]	85.7	-
FoldingNet[34]	88.4	-
DepthContrast[35]	85.4	-
ClusterNet[36]	86.8	-
STRL†[37]	90.9	77.9
OcCo†[38]	89.2	78.3
CrossPoint†[39]	91.2	81.7
CrossNet†[40]	91.5	83.9
HyperIPC(Ours)†	91.8	84.5

from synthetic and real-world data through classification experiments on ModelNet40 [41] and ScanObjectNN [41]. ModelNet40 obtains point cloud by sampling 3D CAD models, and it contains 12,331 objects (9,843 for training and 2,468 for testing) from 40 categories. ScanObjectNN is more realistic and challenging for 3D point cloud classification, consisting of occluded objects from real-world indoor scans. It includes 2,880 objects (2,304 for training and 576 for testing) from 15 categories.

We follow the standard protocols of STRL [37] and Crosspoint [39] to test the accuracy of our network model in object classification. We freeze the point cloud encoder and fit the Support Vector Machine (SVM) classifier on the split of the training dataset. We randomly sample 1,024 points from each object for training and testing, with a batch size of 128 on the DGCNN backbone. Table. I reports the linear classification results on ModelNet40 and ScanObjectNN. HyperIPC outperforms the previous state-of-the-art self-supervised methods in contrastive learning. More notably, we achieve 0.6% and 2.8% improvement over the baseline on the ModelNet40 and ScanObectNN. It can be observed that the underlying structure of real data tends to be hierarchical compared to synthetic datasets, hence leading to relatively more conspicuous results.

Few-shot Object Classification. We conduct Few-Shot Learning (FSL) experiments on the ModelNet40 and ScanObjectNN, using randomly selected n classes from the dataset and m samples from each class, with limited training data that can test the model's generalization ability. We perform ten FSL tasks and reported the mean and standard deviation for a fair comparison with previous methods [43],[38]. Table. II presents the FSL results on ModelNet40 and ScanObjectNN with the setting of $n \in \{5, 10\}$ and $m \in \{10, 20\}$. HyperIPC outperforms the previous work in few-shot classification tasks. These results demonstrate that HyperIPC can learn more discriminative latent representation with limited data, which can alleviate the overfitting issue and acquire semantic information of unknown data.

TABLE II: **Comparison of few-shot classification accuracy with existing methods on ModelNet40 and ScanObjectNN.** We performed ten few-shot tasks and report the mean and standard devia.

Method	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
	ModelNet40			
Rand	31.6±2.8	40.8±4.6	19.9±2.1	16.9±1.5
Jigsaw[44]	34.3±1.3	42.2±3.5	26.0±2.4	29.9±2.6
cTree [43]	60.0±2.8	65.7±2.6	48.5±1.8	53.0±1.3
OcCo[38]	90.6±2.8	92.5±1.9	82.9±1.3	86.5±2.2
CrossPoint[39]	91.0±2.9	95.0±3.4	82.2±6.5	87.8±3.0
ViPFormer[45]	91.1±7.2	93.4±4.5	80.8±4.2	87.1±5.8
HyperIPC(Ours)	94.5±4.4	95.4±2.4	86.7±4.6	91.6±2.1
	ScanObjectNN			
Rand	62.0±5.6	67.8±5.1	37.8±4.3	41.8±2.4
Jigsaw[44]	65.2±3.8	72.2±2.7	45.6±3.1	48.2±2.8
cTree[43]	68.4±3.4	71.6±2.9	42.4±2.7	43.0±3.0
OcCo[38]	72.4±1.4	77.2±1.4	57.0±1.3	61.6±1.2
CrossPoint [39]	72.5±8.3	79.0±1.2	59.4±4.0	67.8±4.4
ViPFormer[45]	74.2±7.0	82.2±4.9	63.5±3.8	70.9±3.7
HyperIPC(Ours)	79.6±7.6	86.0±6.0	68.0±4.6	75.7±3.5

C. Ablations and Analysis

In this section, we report the results of the ablation experiments. We use DGCNN as the point cloud feature extractor in all the classification experiments, with Linear SVM on ScanObjectNN.

Impact of joint learning objective.

We hypothesize that joint learning objectives in hyperbolic space exhibit a more discernible capacity than separate learning objectives. IMHCL encourages the model to acquire the invariance of the point cloud, making the distance between the point cloud of the same category close in hyperbolic space. CMHCL provides the point cloud encoder with semantic information from the visual domain, helping to establish a hierarchical structure in hyperbolic space. The joint learning in hyperbolic space can produce better results as shown in Fig.5. To verify the CMHCL captures the semantic hierarchy, we compare the proposed objective Eq.(10) to the objective that replaces $\mathbf{z}_{mid}$ with $\mathbf{z}_{hyp1}$ in Eq.(10). It can be observed that the representations not only capture the semantic level of the data but also incorporate the information of the 2D images. Fig. 4 illustrates how learned embeddings are arranged on the Poincaré disk. Compared with single hyperbolic contrastive learning method, the joint hyperbolic contrastive learning approach clusters samples according to their labels more effectively. Each category is also closer to the boundary of the Poincaré disk, indicating that the encoder has successfully separated the classes.

TABLE III: **The experimental results under different curvatures.**

c	0.01	0.1	0.3	0.5	1.0
Accuracy	84.0	**84.3**	83.8	83.7	83.7

Curvatures.

Table. III shows the results of the model under different curvatures c. Our model is robust when the curvature is

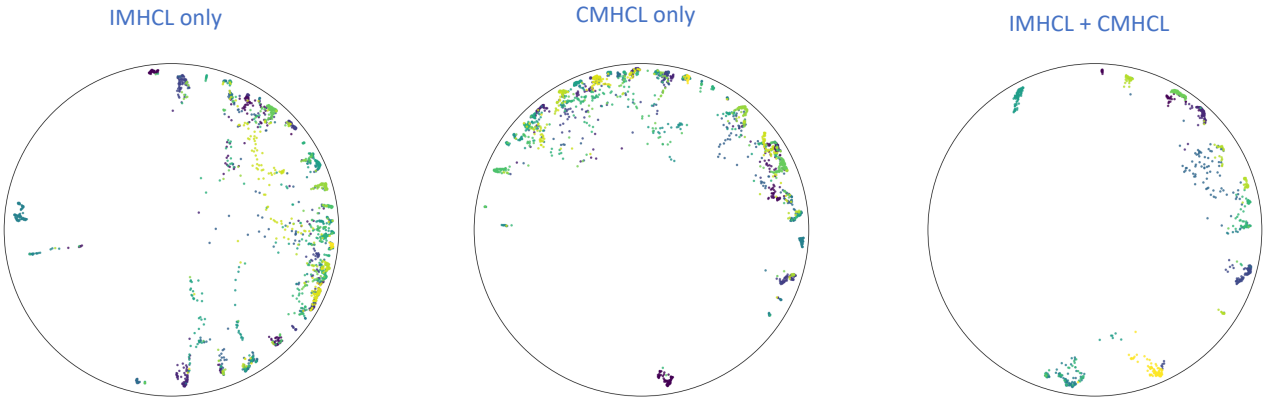

Fig. 4: **UMAP[42] embeddings for ModelNet10 (evaluation sets) on the Poincaré disk.** Each point inside the Poincaré disk corresponds to a sample. Different colors indicate different classes. After IMHCL and CMHCL, the samples are clustered according to the labels, and each category is also closer to the boundary of the Poincaré disk.

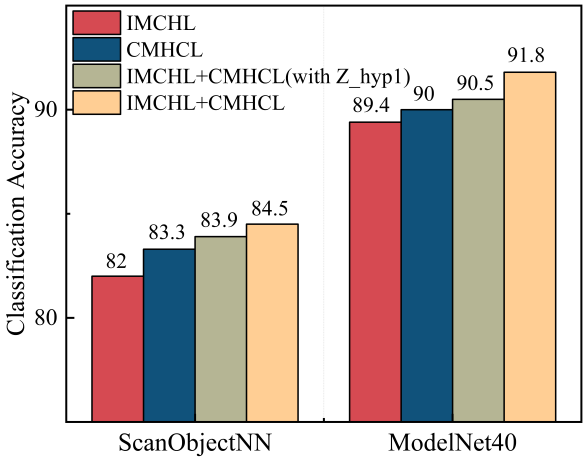

Fig. 5: **Impact of joint learning objective.** Classification accuracy of intra-modal and cross-modal and joint learning objectives on ScanObjectNN and ModelNet40.

small, while larger values cause the model to degrade. It is worth noting that when the curvature is small and gradually approaches zero, the radius of the hyperbolic space becomes infinite and tends to the Euclidean space, which provides better stability.

Image Encoder.

In Crosspoint[39], initializing the image encoder with a normal distribution leads to inaccurate image embedding, resulting in an unsatisfactory hierarchical structure of the point cloud in hyperbolic space. We introduce the CLIP [46] pre-trained model as the image encoder, which can utilize the information implied by the images. CLIP [46] employs a two-tower network that aligns global representation of languages and images using extensive data. As shown in Table. IV, updating the parameters of the image encoder during training allows for accurate semantic hierarchical information to be captured in hyperbolic space.

TABLE IV: **The experimental results under different image encoder.**

Backbone	Back-propagation	Accuracy
VIT-S[46]	✓	**84.3**
VIT-S[46]	×	83.7
Resnet[47]	✓	81.7

V. Conclusion

In this paper, we propose a simple yet effective method to capture the point cloud semantic hierarchy in hyperbolic space. After learning the semantic hierarchy from images, our model can continuously edit the semantic hierarchical features of the point cloud, achieving better results and more discriminating models. Experiments demonstrate that our model outperforms methods that use Euclidean representations. In future work, we will explore combining hyperbolic space with generative models and addressing segmentation tasks within hyperbolic space.

VI. Acknowledgements

This work was supported by the Key Research and Development Program of Shaanxi Province under Grants 2021GY-025.

References

[1] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1, 2019, p. 2.

[2] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[3] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[4] H. Cheng, J. Zhu, N. Hu, J. Chen, and W. Yan, "Ptm: Torus masking for 3d representation learning guided by robust and trusted teachers," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[5] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[6] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[7] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds *et al.*, "Flamingo: a visual language model for few-shot learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 716–23 736, 2022.

[8] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[9] L. Huang, Y. Liu, B. Wang, P. Pan, Y. Xu, and R. Jin, "Self-supervised video representation learning by context and motion decoupling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 886–13 895.

[10] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "Pointcontrast: Unsupervised pre-training for 3d point cloud understanding," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 2020, pp. 574–591.

[11] Y. Chen, M. Nießner, and A. Dai, "4dcontrast: Contrastive learning with dynamic correspondences for 3d scene understanding," in *European Conference on Computer Vision*. Springer, 2022, pp. 543–560.

[12] Z. Lu, Y. Dai, W. Li, and Z. Su, "Joint data and feature augmentation for self-supervised representation learning on point clouds," *Graphical Models*, vol. 129, p. 101188, 2023.

[13] H. Cheng, X. Han, P. Shi, J. Zhu, and Z. Li, "Multi-trusted cross-modal information bottleneck for 3d self-supervised representation learning," *Knowledge-Based Systems*, vol. 283, p. 111217, 2024.

[14] B. Du, X. Gao, W. Hu, and X. Li, "Self-contrastive learning with hard negative sampling for self-supervised point cloud learning," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 3133–3142.

[15] J. Hou, B. Graham, M. Nießner, and S. Xie, "Exploring data-efficient 3d scene understanding with contrastive scene contexts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 587–15 597.

[16] D. Surís, R. Liu, and C. Vondrick, "Learning the predictability of the future," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 607–12 617.

[17] A. Tifrea, G. Bécigneul, and O.-E. Ganea, "Poincar\'e glove: Hyperbolic word embeddings," *arXiv preprint arXiv:1810.06546*, 2018.

[18] I. Chami, Z. Ying, C. Ré, and J. Leskovec, "Hyperbolic graph convolutional neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[19] V. Khrulkov, L. Mirvakhabova, E. Ustinova, I. Oseledets, and V. Lempitsky, "Hyperbolic image embeddings," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6418–6428.

[20] A. Ermolov, L. Mirvakhabova, V. Khrulkov, N. Sebe, and I. Oseledets, "Hyperbolic vision transformers: Combining improvements in metric learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7409–7419.

[21] H. Cheng, J. Zhu, J. Lu, and X. Han, "Edgcnet: Joint dynamic hyperbolic graph convolution and dual squeeze-and-attention for 3d point cloud segmentation," *Expert Systems with Applications*, vol. 237, p. 121551, 2024.

[22] J. Chen, Z. Jin, Q. Wang, and H. Meng, "Self-supervised 3d behavior representation learning based on homotopic hyperbolic embedding," *IEEE Transactions on Image Processing*, vol. 32, pp. 6061–6074, 2023.

[23] M. Yang, M. Zhou, R. Ying, Y. Chen, and I. King, "Hyperbolic representation learning: Revisiting and advancing," *arXiv preprint arXiv:2306.09118*, 2023.

[24] A. Montanaro, D. Valsesia, and E. Magli, "Rethinking the compositionality of point clouds through regularization in the hyperbolic space," *Advances in Neural Information Processing Systems*, vol. 35, pp. 33 741–33 753, 2022.

[25] J. W. Cannon, W. J. Floyd, R. Kenyon, W. R. Parry *et al.*, "Hyperbolic geometry," *Flavors of geometry*, vol. 31, no. 59-115, p. 2, 1997.

[26] J. M. Lee and J. M. Lee, *Smooth manifolds*. Springer, 2012.

[27] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," in *International conference on machine learning*. PMLR, 2018, pp. 40–49.

[28] L. Yi, V. G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, A. Sheffer, and L. Guibas, "A scalable active framework for region annotation in 3d shape collections," *ACM Transactions on Graphics (ToG)*, vol. 35, no. 6, pp. 1–12, 2016.

[29] Q. Xu, W. Wang, D. Ceylan, R. Mech, and U. Neumann, "Disn: Deep implicit surface network for high-quality single-view 3d reconstruction," *Advances in neural information processing systems*, vol. 32, 2019.

[30] 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.

[31] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, "How to train your vit? data, augmentation, and regularization in vision transformers," *arXiv preprint arXiv:2106.10270*, 2021.

[32] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[33] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," *Advances in neural information processing systems*, vol. 29, 2016.

[34] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Point cloud auto-encoder via deep grid deformation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 206–215.

[35] Z. Zhang, R. Girdhar, A. Joulin, and I. Misra, "Self-supervised pretraining of 3d features on any point-cloud," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 252–10 263.

[36] L. Zhang and Z. Zhu, "Unsupervised feature learning for point cloud understanding by contrasting and clustering using graph convolutional neural networks," in *2019 international conference on 3D vision (3DV)*. IEEE, 2019, pp. 395–404.

[37] S. Huang, Y. Xie, S.-C. Zhu, and Y. Zhu, "Spatio-temporal self-supervised representation learning for 3d point clouds," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6535–6545.

[38] H. Wang, Q. Liu, X. Yue, J. Lasenby, and M. J. Kusner, "Unsupervised point cloud pre-training via occlusion completion," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9782–9792.

[39] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, and R. Rodrigo, "Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9902–9912.

[40] Y. Wu, J. Liu, M. Gong, P. Gong, X. Fan, A. Qin, Q. Miao, and W. Ma, "Self-supervised intra-modal and cross-modal contrastive learning for point cloud understanding," *IEEE Transactions on Multimedia*, 2023.

[41] 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.

[42] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[43] C. Sharma and M. Kaul, "Self-supervised few-shot learning on point clouds," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7212–7221, 2020.

[44] J. Sauder and B. Sievers, "Self-supervised deep learning on point clouds by reconstructing space," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[45] H. Sun, Y. Wang, X. Cai, X. Bai, and D. Li, "Vipformer: Efficient vision-and-pointcloud transformer for unsupervised pointcloud understanding," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7234–7242.

[46] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.