REVISITING TRIPLET CONTRASTIVE LEARNING FOR 3D SHAPE UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

In multi-modal 3D pre-training, aligning 3D shapes, their 2D image counterparts, and language descriptions has made tremendous progress. However, we find that supervision from rendered images is unreliable, the semantics encoded in synthetic views are distorted due to domain gap, self-occlusion, and rendering artifacts. Instead, we learn transferable 3D representations from language only. We observe that, in real world, 3D shapes are organized in a tree-like structure, ranging from general concepts to fine-grained instances with specific attributes, while language typically denotes generic concepts and subsumes these various 3D instantiations. Hence, we propose to embed 3D representations in hyperbolic space, whose geometry is well-suited for modeling tree-like data without distortion. To capture the asymmetric hierarchy between language and 3D shapes, we further impose a partial-order relation that language entails 3D shapes. Our method, **Hyper**bolic **C**ontrastive **L**earning (HyperCL), jointly optimizes a hyperbolic geodesic-based contrastive objective and an entailment loss to learn discriminative cross-modal representations while explicitly preserving this hierarchy. We also introduce a rigorously verified benchmark for comprehensively evaluating cross-modal 3D alignment. Across two widely used benchmarks and ours, we improve the performance up to 4.6% absolute percentage on 3D shape classification and 5.5% on cross-modal text-to-3D retrieval under the same parameter budget. The code and evaluation benchmark will be made publicly available.

1 INTRODUCTION

Triplet contrastive learning. Multi-modal 3D contrastive pre-training (Liu et al., 2023; Wang et al., 2024; 2025; Xue et al., 2023; 2024; Zhang et al., 2024; Zhou et al., 2024; Gao et al., 2024b) has emerged as a predominant paradigm for learning transferable 3D representations across various downstream tasks, *e.g.*, zero-shot point cloud classification and cross-modal shape retrieval. However, unlike web-scale image-text data used to pretrain 2D vision-language models (Radford et al., 2021), high-quality language annotations for 3D shapes are extremely scarce. As a result, rendered images are often used as visual proxies for generating synthetic captions via image captioning models (Li et al., 2023; Achiam et al., 2023). Given these tri-modal triplets, existing methods align 3D shapes bidirectionally with language descriptions and rendered images in a shared semantic space.

While this triplet contrastive learning paradigm substantially alleviates the need for manually labeled 3D annotations, it has several limitations. Rendered images exhibit a severe domain gap *w.r.t.* natural images used to pretrain 2D vision-language models, *i.e.*, shifted foreground color statistics and large blank background areas (Liang et al., 2023; Zhang et al., 2024). Furthermore, manually positioned virtual cameras following a predefined trajectory cause severe self-occlusion in certain perspectives during rendering (Luo et al., 2024). The rendering pipeline itself could also introduce unexpected errors, such as color distortions and complete rendering failures. Consequently, learning from these incomplete and misleading semantics in rendered images inevitably has a negative impact on the alignment. Zhang et al. (2024) attempts to address the unreliable image supervision by first adapting CLIP (Radford et al., 2021) image encoder to rendered 3D views, thereby reducing the domain gap. However, how to avoid self-occlusion that fundamentally distorts the perceived 3D shape semantics still remains an open problem. For example, even a person may mistake a living room viewed behind its exterior wall for a rectangular box. Hence we advocate learning from language supervisions only to reduce the influence of low-fidelity and unreliable semantics in vision modality.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Hyperbolic representation learning. In the real world, 3D shapes often convey multifaceted semantics, ranging from various categories to intricate details like color, texture, geometry, design style, and functionality. This naturally organizes into a coarse-to-fine-grained tree that spans a wide spectrum of semantic granularities. In this hierarchy, general concepts described by language are typically positioned near the root, whereas leaf nodes represent specific 3D instantiations characterized by unique attributes. Existing approaches mainly use Euclidean space to model this tree-like structure. However, Euclidean space often struggles to preserve the underlying hierarchical structure without distortion due to its limited polynomial volume growth. As the depth of the underlying tree increases, many fine-grained concepts are compressed into a small region in the space, leading to reduced discriminative capacity. This limitation is particularly prominent in 3D shape understanding, where many real-world instances share a common coarse semantic but differ in subtle attributes.

In contrast, hyperbolic space is naturally suitable for tree-like data that has exponential volume growth *w.r.t.*its radius. This geometric property makes it useful to model an exponentially branching tree without losing discrimination capacity. Hence we propose to contrast 3D shapes against languages in hyperbolic space using a novel hyperbolic contrastive objective. The hyperbolic geometry also allows us to model the asymmetric relation between language and 3D shapes, *i.e.*, language entails 3D shapes. To encode it explicitly, we further propose an entailment loss that ensures positive 3D representations lie inside the entailment cone of its paired text, and negative 3D shapes outside the cone. Experimental results demonstrate superior performance of our HyperCL on both zero-shot 3D shape recognition and cross-modal retrieval, enabling fine-grained cross-modal discrimination while preserving the vision-language hierarchy embedded in data. Overall, our contributions are summarized as the following:

- We identify the limitations of existing triplet contrastive learning framework, and propose to learn from language supervision only.
- We introduce HyperCL, a novel approach leveraging hyperbolic geometry to model the tree-like structure embedded in 3D data and enforce a partial-order relation in which language entails 3D shapes, enhancing fine-grained discrimination while preserving the hierarchy.
- We create a high-quality, manually verified benchmark to rigorously evaluate 3D cross-modal alignment. Experiments show that our method surpasses existing state-of-the-art by a large margin on 3D shape classification, text-to-3D, and 3D-to-text retrieval tasks.

2 Related Work

2.1 3D Triplet Contrastive Learning

Multi-modal 3D pre-training has made tremendous progress in recent years. Besides learning from natural language supervision via bi-directional contrastive loss, the point cloud encoder also contrasts with rendered images of 3D shapes, forming a triplet of (point cloud, text, image). Due to the fact that 3D captioning is underexplored, pioneering works Liu et al. (2023); Xue et al. (2023; 2024); Luo et al. (2023; 2024) focus on how to accurately describe 3D shapes. They first render 3D shapes from preset viewpoints using Blender, then prompt pre-trained image captioning models (Achiam et al., 2023; Li et al., 2023) to generate language descriptions. To cope with the scarcity of 3D pre-training data, Uni3D Zhou et al. (2024) initializes its point cloud encoder directly from pretrained 2D image encoders (Caron et al., 2021; Radford et al., 2021; Sun et al., 2023) instead of training from scratch, establishing a strong optimization prior. MM-Mixing Wang et al. (2025) mixes different point clouds to learn robust and discriminative representations. MixCon3D Gao et al. (2024b) integrates the distinct yet complementary information between rendered images and point clouds to contrast with languages. MRD Wang et al. (2024) performs multi-modal distillation to maintain the semantic relations of both intra- and inter-modalities in the embedding space.

However, given rendered images, CLIP fails to encode meaningful semantics from rendered images due to domain gap. Besides, how to position virtual cameras and select appropriate viewpoints to avoid self-occlusion is still an open problem. Furthermore, rendering pipeline sometimes leads to unexpected errors. TAMM Zhang et al. (2024) adds an extra stage that adapts CLIP image encoder via adapter (Gao et al., 2024a) to rendered images for extracting faithful semantic information. However, better alignment with rendered images during training does not necessarily translate into better generalization, possibly due to overlapped semantics with language. Even if the distorted

semantics brought by domain gap can be faithfully restored, the multi-view rendering at scale is prohibitively expensive, and the fundamental self-occlusion issue inherent to any 2D projection of 3D shapes remains difficult to address. Hence we propose to learn from language only, bypassing the limitations of the supervision signal in rendered images.

2.2 Hyperbolic Learning

Characterized by the exponential volume expansion, hyperbolic geometry is naturally suited for modeling hierarchical data such as trees and graphs. Hyperbolic learning can be divided into two groups: 1) the first line of work builds hyperbolic neural networks (Ganea et al., 2018b; Shimizu et al., 2020) by designing hyperbolic counterparts of euclidean layers, such as convolution and multinomial logistic regression, and these networks are optimized using specialized riemannian optimizers; 2) the second group still keeps the main chunk of model in Euclidean space, but with an added exponential mapping projection to transform euclidean features onto hyperbolic data points. Foundational works Nickel & Kiela (2017; 2018) establish the efficacy of two representative models in hyperbolic geometry, *i.e.*, poincaré ball and lorentz model, for capturing lexical hierarchies within text modality. Ganea et al. (2018a) further introduces entailment cone to encode asymmetric is-a relations in hyperbolic space, which inspires our batch-wise entailment loss.

In computer vision and multi-modal learning, hyperbolic geometry has been adapted to various tasks including 2D representation learning (Ermolov et al., 2022; Ge et al., 2023), image and emotion classification (Chen et al., 2023; Khrulkov et al., 2020; Guo et al., 2022), image segmentation (Atigh et al., 2022), open-vocabulary perception (Doan et al., 2024; Kong et al., 2024; Peng et al., 2025), 2D vision-language models (Desai et al., 2023; Ramasinghe et al., 2024; Pal et al., 2025), and scaling models to billions of parameters (Mandica et al., 2024). However, applying hyperbolic geometry to model 3D data is less explored. Liu et al. (2025) is the first to utilize hyperbolic learning in 3D domain but without large-scale pre-training. Following prior works Desai et al. (2023); Kong et al. (2024) that enforce the partial relation such as object entails scenes or caption entails images/objects in 2D, we enforce caption entails 3D shapes to build vision-language hierarchy and adopt the lorentz model which is numerically more stable and suitable for gradient optimization than poincaré ball.

3 Method

We first briefly introduce hyperbolic geometry and the triplet contrastive learning paradigm in Section 3.1. Then we analyze the current pitfall of existing triplet contrastive learning framework in Section 3.2, followed by our proposed method HyperCL with hyperbolic geodesic-based contrastive loss and entailment loss in Section 3.3.

3.1 Preliminaries

Hyperbolic geometry. Unlike Euclidean and hypersphere space with zero and constant positive curvature, hyperbolic space is a type of riemannian manifold with constant negative curvature. There are five equivalent models of hyperbolic space, and we focus on lorentz model in this work since it is suitable for optimization. We refer to Ratcliffe (2006) for a complete treatment.

In lorentz model, hyperbolic space $\mathbb{L}^n$ is represented as $\left\{\mathbf{h} \in \mathbb{R}^{n+1} : \langle \mathbf{h}, \mathbf{h} \rangle_{\mathcal{L}} = -1/c, c > 0\right\}$ on the upper half of a two-sheeted hyperboloid. The curvature c measures the degree to which hyperbolic space deviates from flat Euclidean space. A hyperbolic representation $\mathbf{h}$ is $[\mathbf{h}_{\text{time}}; \mathbf{h}_{\text{space}}]$ with time and space component $\mathbf{h}_{\text{time}} \in \mathbb{R}$, $\mathbf{h}_{\text{space}} \in \mathbb{R}^n$. $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ is lorentz inner product, computed as $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{L}} = \langle \mathbf{u}_{\text{space}}, \mathbf{v}_{\text{space}} \rangle_{\mathcal{E}} - \mathbf{u}_{\text{time}} \mathbf{v}_{\text{time}}$, and $\langle \cdot, \cdot \rangle_{\mathcal{E}}$ is euclidean inner product. Given the space component, the corresponding time component is calculated as $\mathbf{h}_{\text{time}} = \sqrt{1/c + \|\mathbf{h}_{\text{space}}\|^2}$.

Analogous to straight lines in Euclidean space, the shortest path connecting two points in hyperbolic space is called geodesic. In lorentz model, geodesics are curves traced by the intersection of the hyperboloid with hyperplanes passing through the origin $\mathbf{O}$ of $\mathbb{R}^{n+1}$. The lorentz distance between two points in hyperbolic space is:

$$d_{\mathcal{L}}(\mathbf{u}, \mathbf{v}) = \sqrt{1/c} \cdot \cosh^{-1}(-c \langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{L}}) \,, \tag{1}$$

Table 1: We re-train OpenShape on ensembled dataset. CLIP ViT-B/16 is used as the frozen image and text encoder. Rendered images are substantially worse than languages. Acc@1 is reported.

Image	Text	OLVIS	MN40
✓	✓	46.9	84.2
✓	✗	26.0$_{\downarrow 20.9}$	74.2$_{\downarrow 10.0}$
✗	✓	45.5$_{\downarrow 1.4}$	84.5$_{\uparrow 0.3}$

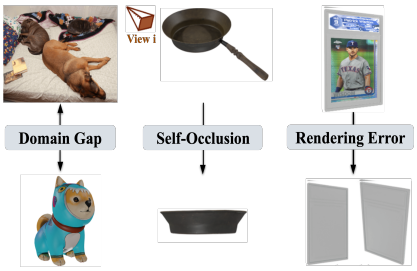


Figure 1: Problems when contrasting with rendered images only.

Table 2: Hyperbolicity of the representation space induced by Uni3D. A lower value indicates a stronger underlying tree-like hierarchy.

Model	δ-hyperbolicity ↓	
	OLVIS	MN40
Uni3D-B	0.24	0.26
Uni3D-L	0.20	0.24
Uni3D-G	0.19	0.24

The tangent space $\mathcal{T}_{\mathbf{u}}\mathbb{L}^n = \left\{\mathbf{h} \in \mathbb{R}^{n+1} : \langle \mathbf{h}, \mathbf{u} \rangle_{\mathcal{L}} = 0\right\}$ at a hyperbolic point $\mathbf{u}$ is an Euclidean space that locally approximates hyperbolic space around $\mathbf{u}$. The exponential map is used to project a point $\mathbf{e} \in \mathbb{R}^n$ from tangent space onto hyperbolic manifold:

$$\mathbf{h}_{\text{space}} = \exp\mathrm{m}_{\mathbf{O}}(\mathbf{e}) = \frac{\sinh(\sqrt{c}\|\mathbf{e}\|)}{\sqrt{c}\|\mathbf{e}\|}\mathbf{e}. \tag{2}$$

Triplet contrastive learning. As shown in Figure 2 (a), existing methods typically take triplets $\{\mathbf{x}_i^P, \mathbf{x}_i^I, \mathbf{x}_i^T\}_{i=1}^B$ with point cloud, rendered image, and language description as input. They are first encoded into $\widetilde{\mathbf{e}}_i^P = f_P(\mathbf{x}_i^P)$, $\widetilde{\mathbf{e}}_i^I = f_I(\mathbf{x}_i^I)$, $\widetilde{\mathbf{e}}_i^T = f_T(\mathbf{x}_i^T)$ by point cloud, image, and text encoders. Then they are normalized to project onto unit sphere space: $\mathbf{e}_i^P = \widetilde{\mathbf{e}}_i^P / \|\widetilde{\mathbf{e}}_i^P\|$, $\mathbf{e}_i^T = \widetilde{\mathbf{e}}_i^T / \|\widetilde{\mathbf{e}}_i^T\|$, $\mathbf{e}_i^I = \widetilde{\mathbf{e}}_i^I / \|\widetilde{\mathbf{e}}_i^I\|$. Let $l(\mathbf{a}, \mathbf{b}) = \log \frac{\exp(\mathcal{K}(\mathbf{a},\mathbf{b})/\tau)}{\sum_{j=1}^B \exp(\mathcal{K}(\mathbf{a},\mathbf{b}_j)/\tau)}$ be the log-softmax operation where $\mathcal{K}(\cdot)$ is a similarity measure and τ is temperature. In Euclidean space, $\mathcal{K}(\cdot)$ is cosine similarity. The triplet contrastive learning objective $\mathcal{L}_E$ consists of the following terms:

$$\mathcal{L}_E^{P\leftrightarrow I} = -\frac{1}{B}\sum_{i=1}^{B}\left(l(\mathbf{e}_i^P, \mathbf{e}_i^I) + l(\mathbf{e}_i^I, \mathbf{e}_i^P)\right), \quad \mathcal{L}_E^{P\leftrightarrow T} = -\frac{1}{B}\sum_{i=1}^{B}\left(l(\mathbf{e}_i^P, \mathbf{e}_i^T) + l(\mathbf{e}_i^T, \mathbf{e}_i^P)\right). \tag{3}$$

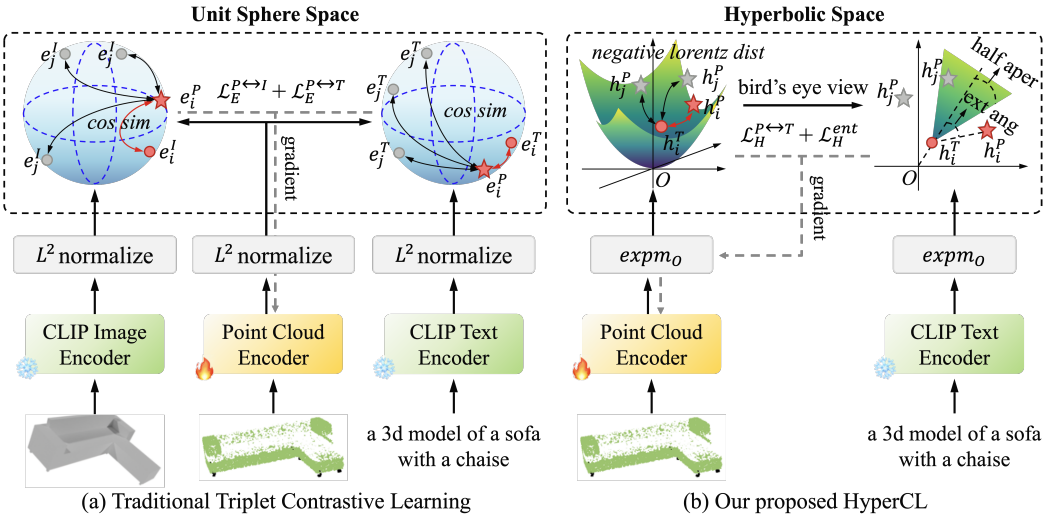
Note that only the point cloud encoder is trainable, image and text encoders are directly from pre-trained CLIP Radford et al. (2021) and they remain frozen.

3.2 Analysis on Triplet Contrastive Learning

We investigate one representative work OpenShape Liu et al. (2023). As shown in Table 1, compared to triplet contrastive learning, the performance degrades severely when language supervision is removed. In contrast, after removing image supervision, it is on par with and can even marginally surpass triplet contrastive learning. This confirms that the gap mainly stems from the inability of CLIP to encode reliable semantics from rendered images due to domain gap (Liang et al., 2023), self-occlusion, and rendering errors. As shown in Figure 1, the frying pan might be mistaken for a shallow bowl, and CLIP is hard to extract meaningful semantics given the rendered large blank area (Liang et al., 2023). In contrast, since language descriptions are aggregated from multiple views and inconsistent descriptions are filtered out (Luo et al., 2024; Xue et al., 2024), they are more robust. Hence, we propose to learn from language supervision only.

3.3 HyperCL

We diagnose whether the learned representation space resembles a tree structure. For this, we choose Uni3D Zhou et al. (2024), a state-of-the-art model pretrained for 3D understanding. We use δ-hyperbolicity in Khrulkov et al. (2020); Fournier et al. (2015) to measure how far the space deviates from a tree, which is zero for a perfect tree and increases as the underlying structure becomes less tree-like (more details are in Appendix A.3). As reported in Table 2, the representations induced by euclidean triplet contrastive learning are already intrinsically hierarchical. This reflects the fact that real-world 3D concepts form a vast taxonomy with exponential branching factor. However, the geometry in which these representations are optimized is misaligned with this tree-like structure.



(a) Traditional Triplet Contrastive Learning (b) Our proposed HyperCL

Figure 2: We embed 3D shapes in hyperbolic space to preserve exponentially expanding tree-like data. In addition, we propose an entailment loss that enforces the relation in which *caption entails 3D shapes*.

Euclidean space only has a polynomial growth with insufficient discrimination capacity. In contrast, hyperbolic space can embed trees without distortion due to its exponential volume growth. Motivated by this, we embed both 3D shapes and language descriptions into hyperbolic space. We next detail our proposed HyperCL with hyperbolic contrastive loss and entailment loss.

3.3.1 Hyperbolic Contrastive Learning.

As shown in Figure 2 (b), we first map $\widetilde{\mathbf{e}}_i^P, \widetilde{\mathbf{e}}_i^T$ onto hyperbolic space via Equation. 2:

$$\mathbf{h}_{i,\text{space}}^T = \exp\mathbf{m}_{\mathbf{O}}\left(\widetilde{\mathbf{e}}_i^T\right). \tag{4}$$

Note that encoders only give space component, time component can be correspondingly derived as mentioned in Section 3.1. In contrast to Euclidean space that uses cosine similarity, we adopt the negative lorentz distance in Equation 1 as similarity measure $\mathcal{K}(\cdot) = -d_{\mathcal{L}}(\mathbf{u}, \mathbf{v})$. Therefore, our hyperbolic contrastive loss is defined as:

$$\mathcal{L}_H^{P\leftrightarrow T} = -\frac{1}{B}\sum_{i=1}^{B}\left(l(\mathbf{h}_i^P, \mathbf{h}_i^T) + l(\mathbf{h}_i^T, \mathbf{h}_i^P)\right). \tag{5}$$

The dynamic range of negative lorentz distance is fundamentally different from the bounded $[-1, 1]$ cosine similarity in Euclidean space. It is unbounded towards negative infinity, allowing for fine-grained semantics to be pushed arbitrarily far without saturation. This exponential growth in hyperbolic space naturally induces a larger discriminative capacity for the model.

3.3.2 Batch-wise Entailment Loss.

We further propose to enforce the relation in which language entails 3D shapes: text denotes general concepts, whereas 3D shapes are specific instantiations of those concepts. To capture this latent vision-language hierarchy, we define an entailment cone for each text $\{\mathbf{h}_i^T\}_{i=1}^B$, see Figure 2 (b) for an illustration in two dimensions. This cone is defined by half-aperture:

$$\text{aper}(\mathbf{h}_i^T) = \sin^{-1}\left(\frac{2K}{\sqrt{c}\|\mathbf{h}_{i,\text{space}}^T\|}\right), \tag{6}$$

where K is a small constant 0.1 used for setting boundary conditions near the origin following previous practice (Desai et al., 2023; Ganea et al., 2018a). We now aim to identify and penalize: 1)

a positive 3D shape $\mathbf{h}_i^P$ lies outside this entailment cone of its paired text; and 2) negative 3D shapes $\mathbf{h}_j^P, j = 1, \ldots, B, j \neq i$ lie inside this cone. To achieve this, we measure:

$$\text{ext}(\mathbf{u}, \mathbf{v}) = \cos^{-1}\left(\frac{\mathbf{v}_{\text{time}} + \mathbf{u}_{\text{time}} c\langle \mathbf{u}, \mathbf{v}\rangle_{\mathcal{L}}}{\|\mathbf{u}_{\text{space}}\|\sqrt{(c\langle \mathbf{u}, \mathbf{v}\rangle_{\mathcal{L}})^2 - 1}}\right). \tag{7}$$

If the exterior angle $\text{ext}(\mathbf{h}_i^T, \mathbf{h}_i^P)$ between paired text and 3D shape is smaller than $\text{aper}(\mathbf{h}_i^T)$, then no loss is incurred, otherwise it will be penalized to force the partial-order relation, and vice versa for unpaired 3D shapes and texts. This is captured by the following objective:

$$\Delta_{ij} = \text{ext}(\mathbf{h}_i^T, \mathbf{h}_j^P) - \text{aper}(\mathbf{h}_i^T), \qquad \mathcal{L}_H^{ent} = \frac{1}{B}\sum_{i=1}^{B} h(\Delta_{ii}) + \frac{1}{B(B-1)}\sum_{i=1}^{B}\sum_{\substack{j=1 \\ j\neq i}}^{B} h(\gamma - \Delta_{ij}), \tag{8}$$

where $h(\cdot) = \max(0, \cdot)$ is hinge function. The second term ensures that $\text{ext}(\mathbf{h}_i^T, \{\mathbf{h}_j^P\}_{j=1,j\neq i}^B) \geq \text{aper}(\mathbf{h}_i^T) + \gamma$, thus encouraging $\{\mathbf{h}_j^P\}_{j=1,j\neq i}^B$ to lie outside the cone with a margin γ. Note that Desai et al. (2023) only imposes a pairwise entailment loss, whereas we additionally constrain negative 3D shapes in a batch to lie outside the cone.

4 Experiments

4.1 Implementation Details

Following Liu et al. (2023); Zhou et al. (2024), we pretrain on the ensembled dataset including Objaverse Deitke et al. (2023), ShapeNet Chang et al. (2015), 3D-FUTURE Fu et al. (2021), and ABO Collins et al. (2022). We adopt PointBERT Yu et al. (2022) initialized from EVA02 Fang et al. (2024) with 85.3M trainable parameters as our point cloud encoder unless otherwise noted. EVA02-E-14-plus from EVA Sun et al. (2023) is used as the frozen text encoder. To accelerate training, we precompute text embeddings $\{\widetilde{\mathbf{e}}^T\}$ and cache them offline. AdamW Loshchilov & Hutter (2017) with batch size 256 is adopted as the optimizer, and we train for 120 epochs. The final loss is $\mathcal{L}_H = \mathcal{L}_H^{P\leftrightarrow T} + \lambda \cdot \mathcal{L}_H^{ent}$, where λ is a balancing factor and set to 0.2. The margin γ in Equation 8 is set to 0.5. More implementation details and training cost are in the appendix A.1 and A.2.

4.2 Evaluation Datasets

We adopt two widely used datasets for assessment, including ModelNet40 (MN40) Wu et al. (2015) and Objaverse-LVIS (OLVIS) Deitke et al. (2023). MN40 is a synthetic CAD model dataset. It contains 2.5K testing samples that span 40 categories. OLVIS is a subset of the Objaverse dataset with human-verified category labels. It has 46K samples spanning 1,156 categories, making it suitable for more challenging open-vocabulary and zero-shot 3D shape classification and retrieval.

Our curated benchmark. Astonishingly, despite its manual labeling, many samples in OLVIS are mislabeled, likely due to varied labeling quality. Such noise can lead to misleading conclusions, as it obscures whether a model truly distinguishes fine-grained semantics or simply benefits from annotation artifacts. To obtain a more reliable testbed for cross-modal alignment, we manually verify each 3D shape in OLVIS. Detailed procedures and statistics are listed in appendix A.4 and A.5. We term the resulting benchmark as cleaned Objaverse-LVIS (c-OLVIS), it contains around 7.7K high-quality samples across 846 categories. c-OLVIS is more trustworthy, by eliminating label noise, it ensures performance gains stem from genuine improvements in 3D semantic understanding rather than fitting to erroneous annotations, while also maintaining a large concept pool to assess real-world 3D understanding.

4.3 Main Comparison

Zero-shot 3D shape classification. We compare with strong baselines under different pre-training data regimes and model parameters in Table 3. We do not compare with MM-MixingWang et al. (2025), MixCon3D Gao et al. (2024b), and MRD Wang et al. (2024) as their model weights or codes are not released. When pretrained on ShapeNet containing only 50K objects, we use a smaller

Table 3: Comparison on zero-shot 3D shape classification with state-of-the-art methods. To make a comprehensive comparison, we test other methods on our c-OLVIS benchmark if the model weights are released and available. Note that Uni3D-L and Uni3D-G are 3.6× and 11.9× more parameters than ours. The best is bold, and second-best is underlined.

Method	Data	OLVIS			c-OLVIS			MN40		
		Acc@1	Acc@3	Acc@5	Acc@1	Acc@3	Acc@5	Acc@1	Acc@3	Acc@5
ReCon Qi et al. (2023)	ShapeNet	1.1	2.7	3.7	-	-	-	61.2	73.9	78.1
CLIP2Point Huang et al. (2023)	ShapeNet	2.7	5.8	7.9	-	-	-	49.5	71.3	81.2
ULIP Xue et al. (2023)	ShapeNet	6.2	13.6	17.9	9.9	19.1	24.1	60.4	79.0	84.4
OpenShape Liu et al. (2023)	ShapeNet	10.8	20.2	25.0	16.5	27.3	32.1	70.3	86.9	91.3
TAMM Zhang et al. (2024)	ShapeNet	13.7	**24.2**	29.2	23.9	35.5	41.2	73.1	88.5	91.9
HyperCL (ours)	ShapeNet	**14.1**	24.1	**29.2**	**25.6**	**37.2**	**43.0**	**76.0**	**89.1**	**93.4**
ULIP2 Xue et al. (2024)	Objaverse (w/o LVIS) + ShapeNet	46.3	-	**75.0**	-	-	-	84.0	-	97.2
ULIP Xue et al. (2023)	Ensembled (w/o LVIS)	21.4	38.1	46.0	-	-	-	71.4	84.4	89.2
OpenShape Liu et al. (2023)	Ensembled (w/o LVIS)	39.1	60.8	68.9	60.9	81.6	86.7	85.3	96.2	97.4
Uni3D-B Zhou et al. (2024)	Ensembled (w/o LVIS)	45.9	**67.4**	74.8	72.6	89.2	92.7	86.1	**97.4**	**98.7**
Uni3D-L Zhou et al. (2024)	Ensembled (w/o LVIS)	45.9	67.2	74.3	73.4	89.6	**93.0**	86.8	96.6	97.8
TAMM Zhang et al. (2024)	Ensembled (w/o LVIS)	42.0	63.6	71.7	67.6	85.4	89.9	86.3	96.6	98.1
HyperCL (ours)	Ensembled (w/o LVIS)	**46.6**	67.2	74.3	**74.8**	**89.8**	92.9	**88.3**	97.3	98.3
ULIP2 Xue et al. (2024)	Objaverse + ShapeNet	50.6	-	79.1	78.8	92.9	95.6	84.7	-	97.1
ULIP Xue et al. (2023)	Ensembled	26.8	44.8	52.6	-	-	-	75.1	88.1	93.2
OpenShape Liu et al. (2023)	Ensembled	46.8	69.1	77.0	70.3	90.0	93.5	84.4	96.5	98.0
Uni3D-B Zhou et al. (2024)	Ensembled	51.7	74.0	80.8	81.6	95.3	97.7	86.3	96.5	97.9
Uni3D-L Zhou et al. (2024)	Ensembled	53.1	74.7	81.5	83.7	96.4	98.3	87.1	96.4	97.9
Uni3D-G Zhou et al. (2024)	Ensembled	53.5	75.3	81.7	84.1	96.7	98.3	87.4	**98.0**	**98.9**
TAMM Zhang et al. (2024)	Ensembled	50.7	73.2	80.6	80.2	95.4	97.6	85.0	96.6	98.1
HyperCL (ours)	Ensembled	**54.4**	**75.3**	**81.7**	**86.2**	**97.3**	**98.7**	**87.8**	97.1	98.0

model with 21.7M parameters. As shown in Table 3, our method achieves state-of-the-art performance across all benchmarks under the same parameter budget. Previous best model Uni3D-B achieves 81.6 on our c-OLVIS dataset, whereas we boost the performance to 86.2 with 4.6 absolute improvement. Besides, Uni3D scales its model to 306.7M (Uni3D-L) and 1B parameters (Uni3D-G) that is 3.6x and 11.9x larger than our model, however we surpass both Uni3D-L and Uni3D-G with non-trivial margin. Scaling to more parameters requires excessively long training time, and a strong regularization strategy is needed to avoid overfitting, hence we do not scale models but rather focus on improving performance via hyperbolic learning with smaller parameter budget. The strong performance indicates that our HyperCL learns more discriminative 3D representations that generalize well to challenging open-world scenarios.

Zero-shot cross-modal 3D retrieval. While prior work Zhou et al. (2024) has shown the potential to perform 3D shape retrieval, we aim to quantitatively assess cross-modal 3D retrieval capabilities. To build such a dataset, we take the intersection of 3D shapes that have human written captions in Cap3D Luo et al. (2023) and those in OLVIS as test samples, leaving us with 4,421 unique shapes spanning 879 categories. As shown in Table 4, our model consistently achieves the best performance, with particularly clear advantages in the more challenging text-to-3D retrieval setting. Under the strict zero-shot setting where models are pre-trained on the ensembled dataset without LVIS (never seeing any test samples during training), our approach outperforms all methods, achieving a notable 3.3% gain on text-to-3D retrieval compared to second best method TAMM Zhang et al. (2024). When pretrained on full ensembled corpus, we further push the performance to 29.1. Both Uni3D and ULIP2 achieve competitive performance on 3D-to-text retrieval, but fail to maintain the same performance on text-to-3D retrieval. In contrast, due to our entailment loss, paired 3D shapes are enforced to lie inside the entailment cone, and vice versa for unpaired 3D shapes, making text-to-3D retrieval more effective while also enhancing 3D-to-text retrieval performance.

Table 4: Zero-shot cross-modal 3D retrieval results. For a fair comparison, when evaluating other methods, we use their sampled point clouds from 3D meshes. Recall@K is used as evaluation metric. The best is bold, second-best is underlined. P → T denote 3D-to-text retrieval, T → P is text-to-3D retrieval. Note Uni3D-L and Uni3D-G are 3.6× and 11.9× more parameters than ours.

Method	Data	P → T			T → P		
		R@1	R@3	R@5	R@1	R@3	R@5
ULIP2 Xue et al. (2024)	ShapeNet	1.0	2.4	3.6	1.0	2.7	3.9
TAMM Zhang et al. (2024)	ShapeNet	<u>3.9</u>	<u>9.4</u>	<u>12.8</u>	<u>4.9</u>	<u>10.5</u>	<u>14.4</u>
HyperCL (Ours)	ShapeNet	**6.0**	**12.1**	**17.1**	**7.0**	**14.7**	**19.2**
OpenShape Liu et al. (2023)	Ensembled (w/o LVIS)	19.6	35.2	45.5	20.4	37.1	46.0
Uni3D-B Zhou et al. (2024)	Ensembled (w/o LVIS)	22.0	39.3	48.8	18.5	34.1	42.7
Uni3D-L Zhou et al. (2024)	Ensembled (w/o LVIS)	21.9	40.1	49.3	18.7	34.2	44.4
Uni3D-G Zhou et al. (2024)	Ensembled (w/o LVIS)	<u>23.5</u>	<u>40.9</u>	<u>50.4</u>	21.1	36.6	45.6
TAMM Zhang et al. (2024)	Ensembled (w/o LVIS)	19.8	37.2	46.6	<u>21.9</u>	<u>38.5</u>	<u>48.0</u>
HyperCL (Ours)	Ensembled (w/o LVIS)	**24.6**	**42.0**	**50.9**	**25.2**	**43.4**	**52.4**
ULIP2 Xue et al. (2024)	Objaverse + ShapeNet	26.3	45.2	<u>55.1</u>	24.3	43.3	52.7
OpenShape Liu et al. (2023)	Ensembled	22.2	39.8	49.3	23.6	42.5	52.3
Uni3D-B Zhou et al. (2024)	Ensembled	24.9	43.2	53.2	21.7	39.1	49.0
Uni3D-L Zhou et al. (2024)	Ensembled	26.0	44.9	54.8	22.1	39.4	48.1
Uni3D-G Zhou et al. (2024)	Ensembled	<u>26.6</u>	<u>45.6</u>	**55.5**	22.3	40.5	50.2
TAMM Zhang et al. (2024)	Ensembled	23.1	41.2	51.9	<u>25.5</u>	<u>44.1</u>	<u>53.4</u>
HyperCL (Ours)	Ensembled	**27.6**	**45.9**	54.8	**29.1**	**48.4**	**57.4**

4.4 Ablation Study

We thoroughly investigate our proposed HyperCL in this section. To reduce training cost, the number of epochs is set to 60.

Table 5: Ablation study on loss components. $\mathcal{L}_E^{P\leftrightarrow T}$ is the baseline. it does not contrast with rendered images, but adopts cosine similarity instead of negative lorentz distance.

	$\mathcal{L}_E^{P\leftrightarrow T}$	$\mathcal{L}_H^{P\leftrightarrow T}$	$\mathcal{L}_H^{P\leftrightarrow T} + \mathcal{L}_H^{ent}$
Acc@1			
OLVIS	52.3	53.2$_{\uparrow 0.9}$	53.4$_{\uparrow 1.1}$
c-OLVIS	82.9	84.2$_{\uparrow 1.3}$	84.6$_{\uparrow 1.7}$
MN40	85.7	86.3$_{\uparrow 0.6}$	87.9$_{\uparrow 2.2}$
R@1			
P→T	25.2	26.0$_{\uparrow 0.8}$	27.0$_{\uparrow 1.8}$
T→P	25.8	27.1$_{\uparrow 1.3}$	28.4$_{\uparrow 2.6}$
R@5			
P→T	53.4	54.2$_{\uparrow 0.8}$	55.6$_{\uparrow 2.2}$
T→P	54.7	55.4$_{\uparrow 0.7}$	56.3$_{\uparrow 1.6}$

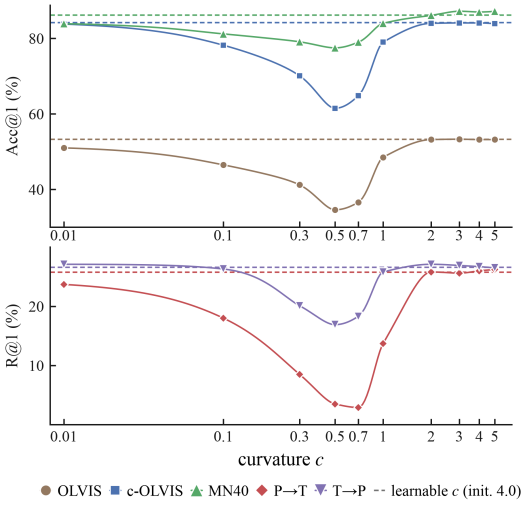


Figure 3: Effect of curvature c on loss $\mathcal{L}_H^{P\leftrightarrow T}$. Dashed lines denote learnable curvature.

Effect of individual loss. Table 5 shows the effect of each loss. We first compare against euclidean baseline $\mathcal{L}_E^{P\leftrightarrow T}$ in Equation. 3. The hyperbolic loss $\mathcal{L}_H^{P\leftrightarrow T}$ alone surpasses baseline by a large margin. As argued in Section 1 and 3.3, this proves that the underlying geometry is tree-like structure and is suitable to be modeled via hyperbolic geometry. The entailment loss further improves the performance. Together the proposed two losses enable discriminative recognition and partial-order relation between texts and 3D instantiations.

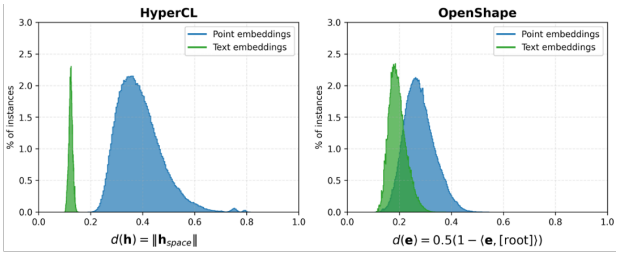


Figure 4: Embedding distances from `[root]`. Text embeddings consistently lie closer to `[root]` than 3D shape embeddings encoded by our model.

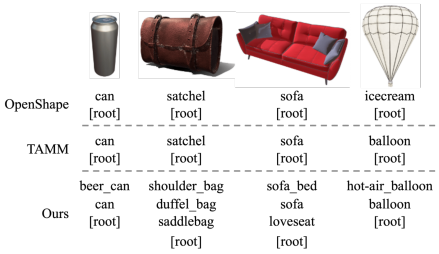


Figure 5: Shape traversal, moving toward `[ROOT]`, our model retrieves categories at multiple levels of abstraction.

Effect of curvature c**.** The curvature measures how hyperbolic space deviates from flat Euclidean space. When it goes to zero, the hyperbolic space gradually degrades to Euclidean space. As shown in Figure 3, both classification and retrieval performance increase from $c = 1$ and stabilizes at $c \geq 2$. As c decreases from 1, performance initially deteriorates, but gradually recovers when it approaches 0 as the underlying geometry becoming increasingly euclidean. However, setting $c = 0.01$ still lags behind $c \geq 2$. This v-shaped curve suggests that moderate curvature may provide neither the strong hierarchical modeling capacity of highly curved hyperbolic space nor the near-Euclidean behavior. Making c a learnable parameter (initialized from 4) is roughly the same as fixed $c = 4$ since c is decreasing during training and the model has to constantly adapt to this changing geometry. Hence, we set c to 4 and it remains fixed.

4.5 Qualitative Results

If the partial-order relation of language entails 3D shapes is imposed, h_i^T should lie closer to `[ROOT]` than h_i^P, since text denotes generic concepts that subsume specific 3D instantiations. Following Desai et al. (2023), the origin of lorentz model is used as `[ROOT]`. As shown in Figure 4, texts and 3D shapes are clearly separated, with text concentrated near the origin and 3D shapes scattered toward the hyperbolic boundary where the distance becomes larger to encompass fine-grained concepts. Conversely, OpenShape Liu et al. (2023) inevitably overlaps two modalities with a large portion. To further examine this vision-language hierarchy, we perform shape traversal. Between a query 3D representation and origin `[ROOT]`, we interpolate evenly spaced points along the hyperbolic geodesic connecting them, and use each intermediate point to retrieve its nearest category. The retrieved categories should become progressively more abstract as we move toward `[ROOT]`. As shown in Figure 5, euclidean baselines Liu et al. (2023); Zhang et al. (2024) tend to directly jump to `[ROOT]`, whereas our model yields a much smoother trajectory of meaningful intermediate concepts, revealing multiple levels of abstraction.

5 Conclusion

In this paper, we present HyperCL for learning transferable 3D representations. We demonstrate that learning from language supervision only effectively bypasses the inability of VLMs to encode reliable semantics from rendered images. We leverage hyperbolic geometry to enhance discrimination and model the inherent vision-language hierarchy that language entails 3D shapes. The proposed hyperbolic contrastive and batchwise entailment losses effectively preserves this hierarchical structure. Furthermore, our manually verified c-OLVIS benchmark serves as a rigorous evaluation. Experimental results show that our method achieves state-of-the-art performance on both 3D shape classification and cross-modal retrieval tasks.

Limitations Though the language descriptions are relatively robust than rendered images, the problems caused by self-occlusion and rendering artifacts may still be propagated into captions. 3D captioning is best performed without rendered images as proxy via 3D LLMs that directly take 3D shapes as input. However this is beyond our paper's scope and we leave this to future work.

AI USE STATEMENT

In this work, we used generative AI tools for supporting qualitative and thematic data analysis. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for sourcing/searching for information and editing a research paper to improve readability. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. Hyperbolic image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4443–4452. IEEE, 2022.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Chih-Yao Chen, Tun Min Hung, Yi-Li Hsu, and Lun-Wei Ku. Label-aware hyperbolic embeddings for fine-grained emotion classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10947–10958, 2023.

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21094–21104. IEEE, 2022.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13142–13153, 2023.

Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *International Conference on Machine Learning*, pp. 7694–7731. PMLR, 2023.

Thang Doan, Xin Li, Sima Behpour, Wenbin He, Liang Gou, and Liu Ren. Hyp-ow: Exploiting hierarchical structure learning with hyperbolic distance enhances open world object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1555–1563, 2024.

Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7399–7409. IEEE, 2022.

Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024.

Hervé Fournier, Anas Ismail, and Antoine Vigneron. Computing the gromov hyperbolicity of a discrete metric space. *Information Processing Letters*, 115(6-8):576–579, 2015.

Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129 (12):3313–3337, 2021.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *International conference on machine learning*, pp. 1646–1655. PMLR, 2018a.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *Advances in neural information processing systems*, 31, 2018b.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International journal of computer vision*, 132(2):581–595, 2024a.

Yipeng Gao, Zeyu Wang, Wei-Shi Zheng, Cihang Xie, and Yuyin Zhou. Sculpting holistic 3d representation in contrastive language-image-3d pre-training. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22998–23008. IEEE, 2024b.

Songwei Ge, Shlok Mishra, Simon Kornblith, Chun-Liang Li, and David Jacobs. Hyperbolic contrastive learning for visual representations beyond objects. In *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 6840–6849. IEEE, 2023.

Yunhui Guo, Xudong Wang, Yubei Chen, and Stella X Yu. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–10. IEEE, 2022.

Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22100–22110. IEEE, 2023.

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 6417–6427. IEEE, 2020.

Fanjie Kong, Yanbei Chen, Jiarui Cai, and Davide Modolo. Hyperbolic learning with synthetic captions for open-world detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16762–16771. IEEE, 2024.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PmLR, 2023.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7061–7070. IEEE, 2023.

Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. *Advances in neural information processing systems*, 36:44860–44879, 2023.

Yingjie Liu, Pengyu Zhang, Ziyao He, Mingsong Chen, Xuan Tang, and Xian Wei. Hyperbolic contrastive learning for hierarchical 3d point cloud embedding. *arXiv preprint arXiv:2501.02285*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pre-trained models. *Advances in Neural Information Processing Systems*, 36:75307–75337, 2023.

Tiange Luo, Justin Johnson, and Honglak Lee. View selection for 3d captioning via diffusion ranking. In *European Conference on Computer Vision*, pp. 180–197. Springer, 2024.

Paolo Mandica, Luca Franco, Konstantinos Kallidromitis, Suzanne Petryk, and Fabio Galasso. Hyperbolic learning with multimodal large language models. In *European Conference on Computer Vision*, pp. 382–398. Springer, 2024.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017.

Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International conference on machine learning*, pp. 3779–3788. PMLR, 2018.

Avik Pal, Max Van Spengler, Guido D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. In *International Conference on Learning Representations*, volume 2025, pp. 87371–87399, 2025.

Zelin Peng, Zhengqin Xu, Zhilin Zeng, Changsong Wen, Yu Huang, Menglin Yang, Feilong Tang, and Wei Shen. Understanding fine-tuning clip for open-vocabulary semantic segmentation in hyperbolic space. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4562–4572. IEEE, 2025.

Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*, pp. 28223–28243. PMLR, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, and Ajanthan Thalaiyasingam. Accept the modality gap: An exploration in the hyperbolic space. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27253–27262. IEEE, 2024.

John G Ratcliffe. *Foundations of hyperbolic manifolds*. Springer, 2006.

Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. Hyperbolic neural networks++. *arXiv preprint arXiv:2006.08210*, 2020.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

Huiqun Wang, Yiping Bao, Panwang Pan, Zeming Li, Xiao Liu, Ruijie Yang, and Di Huang. Multi-modal relation distillation for unified 3d representation learning. In *European Conference on Computer Vision*, pp. 364–381. Springer, 2024.

Jiaze Wang, Yi Wang, Ziyu Guo, Renrui Zhang, Donghao Zhou, Guangyong Chen, Anfeng Liu, and Pheng-Ann Heng. Mm-mixing: Multi-modal mixing alignment for 3d understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7744–7752, 2025.

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*, pp. 1912–1920, 2015.

Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1179–1189. IEEE, 2023.

Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27081–27091. IEEE, 2024.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19291–19300. IEEE, 2022.

Zhihao Zhang, Shengcao Cao, and Yu-Xiong Wang. Tamm: Triadapter multi-modal learning for 3d shape understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21413–21423. IEEE, 2024.

Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. In *International Conference on Learning Representations*, volume 2024, pp. 46766–46782, 2024.

A Appendix

A.1 More Implementation Details

As listed in Table 6, we mainly follow previous practices Liu et al. (2023); Zhou et al. (2024); Xue et al. (2024) in multi-modal 3D contrastive pre-training for data augmentation and model training hyperparameters.

Table 6: Hyperparameters used in our HyperCL.

Hyperparameters	Value	Descriptions
Normalization	✓	normalize xyz coordinates of point cloud into [-1, 1]
Point dropout	0.875	the maximum ratio of dropped points is 0.875
Random scale	[0.8, 1.25]	uniformly scale points
Random shift	[-0.1, 0.1]	shift x, y, z coordinates independently
Random rotation	✓	small rotation jittering around x, y, z axes
Random rotation around gravity-axis	✓	rotate around gravity-axis for learning rotation invariance
Random RGB drop	✓	replace the RGB values of points with 0.4
CLIP model	ViT-E/14+	CLIP model used to compute offline text embeddings
Number of points	10,000	number of sampled points from 3D mesh
Hyperbolic temperature τ	0.06	
Learning rate	0.001	
Learning rate schedule	WarmUpStepLR with decaying factor 0.967	
Optimizer	AdamW with betas = (0.9, 0.98) and eps=1e-6	
Weight decay	0.01	no weight decay on temperature τ, curvature c
Batch Size per GPU	256	
Epochs	120	
AMP	✓	use FP16 for training
GPUs	8 × NVIDIA A800 (PCIe, 80GB)	

A.2 Training Cost

Since both our hyperbolic contrastive loss and entailment loss use the already well-optimized (inverse) hyperbolic triangle functions provided by PyTorch, and the main computation stems from point cloud encoder whose parameters still reside in Euclidean space and can be optimized via AdamW, hence both the memory footprint and FLOPs maintain the same as euclidean baselines as shown in Table 7. The speed slightly decreases to 3.36 it/s compared with $\mathcal{L}_E^{P\leftrightarrow T}$, however, this is because the host-side kernel-launch overhead from a chain of fine-grained primitives such as `norm`, `acosh`, `clamp`, `sinh` rather than computational burden.

Table 7: Comparison of training cost.

	Speed (it/s)	Peak Memory (GB)	GFLOPs
$\mathcal{L}_E^{P\leftrightarrow I} + \mathcal{L}_E^{P\leftrightarrow T}$	3.30	46.65	72.05
$\mathcal{L}_E^{P\leftrightarrow T}$	3.43	46.65	72.05
$\mathcal{L}_H^{P\leftrightarrow T} + \mathcal{L}_H^{ent}$	3.36	46.65	72.05

A.3 More Details on Hyperbolicity

Gromov hyperbolicity quantifies how closely a metric space resembles the structure of a tree. Let $(\mathcal{M}, d)$ be a metric space, where $\mathcal{M}$ is the set of representations and $d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is the distance measure. In our experiments, each representation is a 3D shape embedding $\tilde{e}_i \in \mathbb{R}^D$ produced by Uni3D (Zhou et al., 2024), and d is euclidean metric $d(\tilde{e}_i, \tilde{e}_j) = \|\tilde{e}_i - \tilde{e}_j\|_2$.

For a base point $\tilde{e}_b \in \mathcal{M}$, the Gromov product of $\tilde{e}_i$ and $\tilde{e}_j$ is defined as

$$(\tilde{e}_i \mid \tilde{e}_j)_{\tilde{e}_b} = \frac{1}{2}\left(d(\tilde{e}_i, \tilde{e}_b) + d(\tilde{e}_j, \tilde{e}_b) - d(\tilde{e}_i, \tilde{e}_j)\right). \tag{9}$$

In a rooted tree, it equals the length of the initial path shared by the paths from $\tilde{e}_b$ to $\tilde{e}_i$ and $\tilde{e}_j$. Thus, a larger Gromov product indicates a deeper branching point, whereas a smaller value indicates branching closer to the base point.

Given three representations $\tilde{e}_i$, $\tilde{e}_j$, and $\tilde{e}_k$, a tree satisfies the following Gromov product inequality:

$$(\tilde{e}_i \mid \tilde{e}_k)_{\tilde{e}_b} \geq \min\left\{(\tilde{e}_i \mid \tilde{e}_j)_{\tilde{e}_b}, (\tilde{e}_j \mid \tilde{e}_k)_{\tilde{e}_b}\right\}. \tag{10}$$

The deviation from this tree condition is quantified by the base-point Gromov hyperbolicity:

$$\delta_{\tilde{e}_b} = \max_{\tilde{e}_i, \tilde{e}_j, \tilde{e}_k \in \mathcal{M}} \left[\min\left\{(\tilde{e}_i \mid \tilde{e}_j)_{\tilde{e}_b}, (\tilde{e}_j \mid \tilde{e}_k)_{\tilde{e}_b}\right\} - (\tilde{e}_i \mid \tilde{e}_k)_{\tilde{e}_b}\right]. \tag{11}$$

A tree metric has $\delta_{\tilde{e}_b} = 0$, since the above inequality holds for every triple of points. A positive value measures the largest violation of this condition among sampled representations. Therefore, a smaller value indicates a more tree-like structure, whereas a larger value indicates the opposite. To account for the overall scale of the embedding space, we report the relative hyperbolicity:

$$\delta_{\text{rel}} = \frac{2\delta_{\tilde{e}_b}}{\operatorname{diam}(\mathcal{M})}, \qquad \operatorname{diam}(\mathcal{M}) = \max_{\tilde{e}_i, \tilde{e}_j \in \mathcal{M}} d(\tilde{e}_i, \tilde{e}_j). \tag{12}$$

In practice, we follow Khrulkov et al. (2020); Fournier et al. (2015) to compute the hyperbolicity.

A.4 More Details on c-OLVIS Construction

Based on OLVIS, we construct a cleaned version of it through manual inspection and refinement. During inspection, for each object we utilize the official Sketchfab website stored in the metadata. The viewer supports rotation and zooming in of the 3D shape, allowing labelers to inspect it from multiple viewpoints and reduce errors caused by self-occlusion. To ensure consistent criteria, each 3D shape will display its assigned category name, associated synonyms, along with its LVIS definition. This information helps reviewers distinguish intended category from polysemy.

We develop a lightweight annotation tool that provides three possible options: accepted (a), rejected (r), and unsure (u). Annotators can submit decision by pressing corresponding key, and the tool automatically opens the online URL using chromium. We evenly split OLVIS into three disjoint subsets and assign each subset to one labeler. Candidates marked as accepted during initial screening phase are passed for a second round of review by other independent labelers. Only candidates accepted in both stages are kept in our c-OLVIS benchmark. This two stage procedure reduces the chances that an individual oversight or inconsistent interpretation propagates into the final benchmark. We encourage annotators to label it as unsure rather than accepted to avoid being deterministically filtered out in second round. Objects marked as unsure in the first round are collected and sent for committee screening as detailed next.

We find that many categories in OLVIS are either intrinsically ambiguous or related with each other via parent-child and synonym relationships. For example, table is the parent concept of dining table and coffee table, while in the meantime, it may also be used as a synonym for desk. We first organize such ambiguous concepts into groups via WordNet and instruct reviewers to press unsure button whenever a 3D shape lies in these groups. Then all reviewers are shown the same unsure object, when the group contains parent-children relationship, we apply a strict rule that it should be assigned to the most specific subclass unless it can not be classified into any subclass in that group, then it will be assigned as the parent class. Only the vote for a concept receives more than half of the reviewers then the object will be assigned to that class, otherwise, the object is discarded.

We asked three graduate student to independently assess their given split using the same criteria and it takes around 15 days to annotate our c-OLVIS benchmark. The time is mostly spent on waiting for online rendering and committee review for unsure objects. Examples are given in Figure 6.

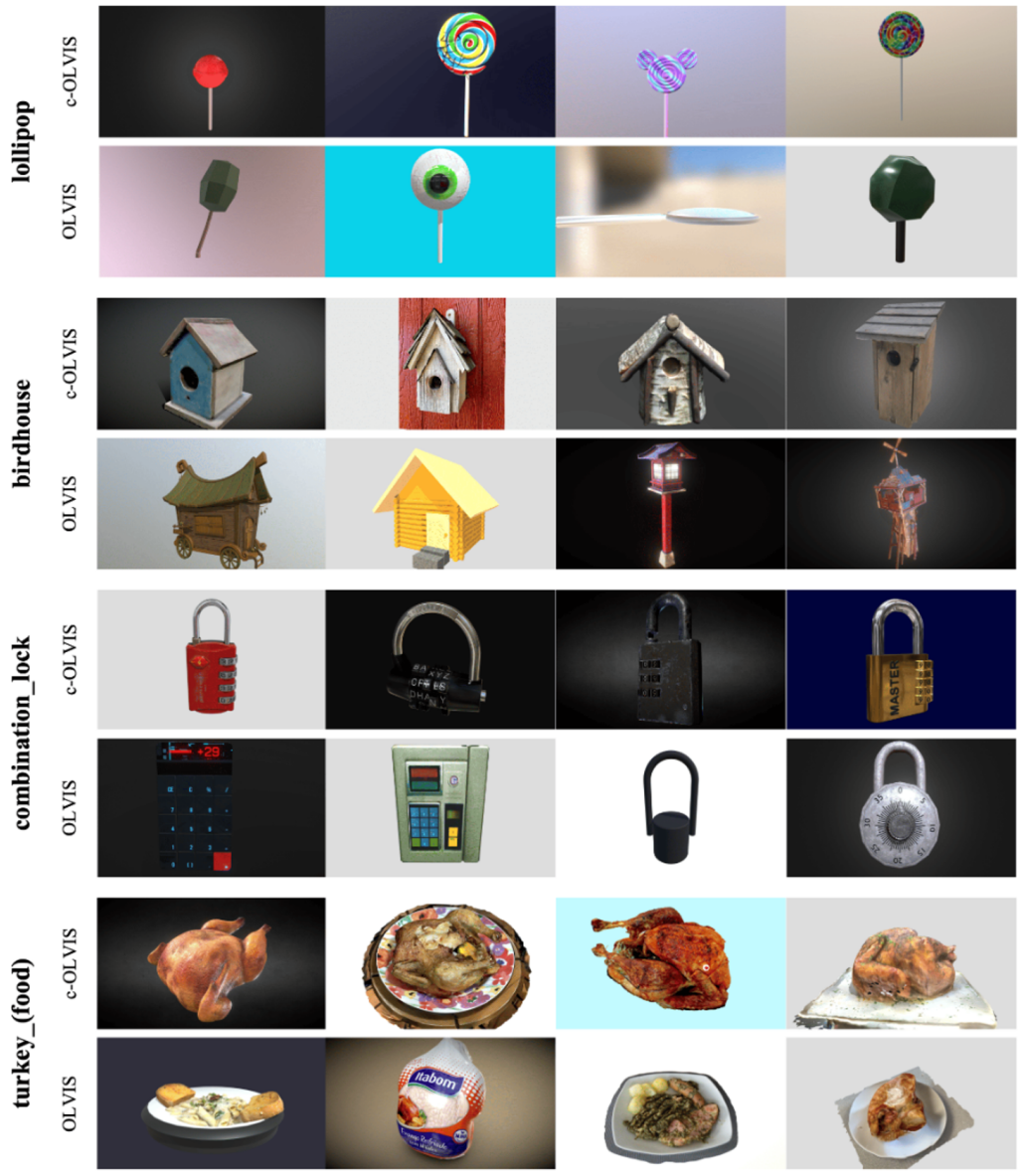


Figure 6: Examples from the manual verification. For four categories lollipop, birdhouse, combination_lock and turkey_(food), the upper row shows 3D shapes that were correctly labelled and kept in our c-OLVIS, while the lower row shows OLVIS samples that were mislabelled and removed.

A.5 More Statistics of c-OLVIS

Compared to OLVIS, c-OLVIS has 7,678 unique 3D shapes from 846 categories, covering 73.2% original categories of OLVIS. This diverse coverage of real-world concepts ensures that our proposed c-OLVIS benchmark remains representative for zero-shot 3D understanding. After rigorous filtering and correction, we remove substantial annotation noise, therefore it provides a more reliable evaluation, allowing the improvement to be more likely to reflect true semantic recognition rather than false exploitation on noisy labels. While at the meantime, even though the accuracy on our c-OLVIS benchmark is typically higher than that on OLVIS, the resulting c-OLVIS benchmark is not merely an easy subset that contains only frequent categories from OLVIS. As shown in Figure 7, categories that contain less 3D shapes are well preserved, the main annotation noise comes from several categories with a large amount of objects (the rightmost bar in right subfigure). Thus, the

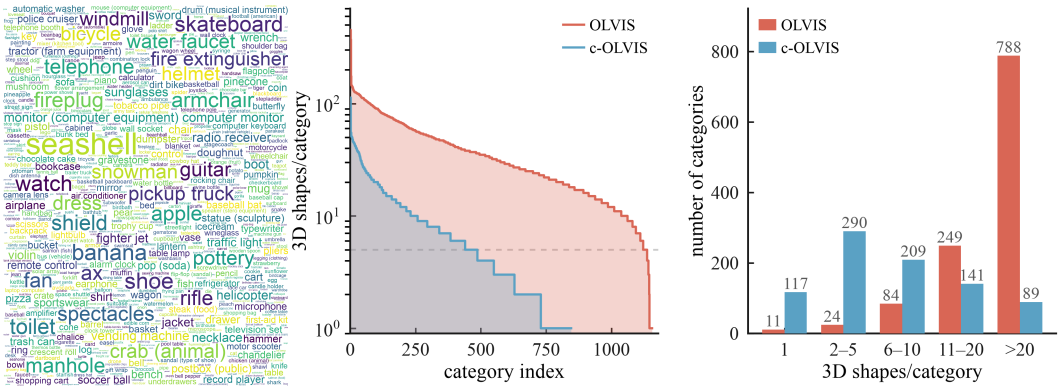


Figure 7: Statistics of our c-OLVIS. **Left:** category vocabulary of c-OLVIS, 846 categories in total. **Middle:** the rank–frequency distribution of class sizes on category-index axis. **Right:** the number of categories per class-size bin.

large accuracy gap between our c-OLVIS and OLVIS can not be explained simply by retaining a large portion of easily recognizable categories. Instead, it is consistent with our manual verification process that removes many noisy or ambiguous instances while preserving low-frequency concepts. This is exactly why annotation quality matters: a reliable benchmark should reward correct semantic recognition rather than penalize it because of incorrect labels.



Figure 8: Qualitative visualization of retrieval results. Top: 3D → text retrieval (top-1 caption). Bottom: text → 3D retrieval (top-5 shapes). Similarities decrease from left to right.

A.6 More Qualitative Results

We visualize cross-modal retrieval results in Figure 8. Euclidean geometry based models often produce partially correct retrievals, they may match coarse semantics but confuse fine-grained concepts such as car *v.s.* truck *v.s.* pickup truck, or miss certain attributes in the query, *e.g.*, failing to recognize the orange color of the scooter. This behavior is consistent with the fact that euclidean geometry embeds tree that has a large amount of concepts with distortion, highlighting the benefit of hyperbolic space for preserving fine-grained semantics.