Hyperbolic Learning With Supervision From Any Granularity

Mina Ghadimi Atigh
University of Amsterdam

Max van Spengler
University of Amsterdam

Teng Long
University of Amsterdam

Melika Ayoughi
University of Amsterdam

Tejaswi Kasarla
University of Amsterdam

Pascal Mettes
University of Amsterdam

Abstract

Supervised classification commonly follows a one-vs-rest paradigm where each sample belongs to one category from a set of independent classes. In real-world settings, classes are typically not independent, but organized hierarchically from coarse-grained to fine-grained. More pressingly, people naturally annotate at different levels of granularity, depending on their expertise, biases, or data quality. What should be the correct label of a picture of a bird? Is it *animal*, *bird*, *albatross*, or *Laysan albatross*? What if one annotator is an ornithologist and the other has little bird knowledge? Similarly, if two pictures of a *Laysan albatross* differ in blurriness, we tend to annotate blurry ones more generically, as we are unsure of details that differentiate classes at the finest levels. Currently, many annotations are removed, ignored, or reassigned because they do not match the required granularity. Instead of viewing the world as a flat, independent collection of concepts, this paper strives to perform supervised learning with labels at any granularity. We propose a hyperbolic embedding space, where classes are hierarchically organized as prototypes. We introduce a coarse-to-fine Busemann approach, where images are optimized to the correct region of the hyperbolic embedding space by projecting their labels – which can be as precise or generic as desired – to ideal prototypes on the boundary of the Poincaré ball. Experiments show that our approach improves multi-granular classification and beats the current state-of-the-art, which views different granularities as independent, instead of a connected tree.[1].

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

One-vs-rest optimization is the backbone of conventional supervised learning. By assigning a label to each sample, machine learning models can be directly optimized towards that label through *e.g.,* a softmax cross-entropy or contrastive objective. While a sensible approach, many real-world classification problems involve classes that are related to each other, commonly through a tree-like structure. Not only are such hierarchical relations common, in many domains they provide important information about generalization to new classes (Li et al., 2022; Ramzi et al., 2022; Atigh et al., 2022; Liu et al., 2020; Dhall et al., 2020; Yu et al., 2025; Liu et al., 2025) and error severity(Garg et al., 2022; Atigh et al., 2022; Liu et al., 2020; Yu et al., 2025; Ma et al., 2021). This paper focuses on a real but often overlooked issue when it comes to data annotation: people do not naturally annotate all samples at the level of the leaves of the hierarchy. Consider Figure 1 as an example, which highlights annotation variations due to image distortions, occlusions, or annotation biases. If we want to make use of the multi-granular nature of the labelling process, we need to look beyond classes as independent entities and model their hierarchical organization.

A wide range of works have shown that for modelling the hierarchical structure of classification tasks, hyperbolic space is superior over the default Euclidean space(Nickel & Kiela, 2017, 2018; Long et al., 2020; Atigh et al., 2022; van Spengler & Mettes, 2025). Li et al. (2024b) investigate action recognition datasets and use hyperbolic space to embed the hierarchical

[1]Code available at: `https://github.com/MinaGhadimiAtigh/hyperbolic-granular-learning`

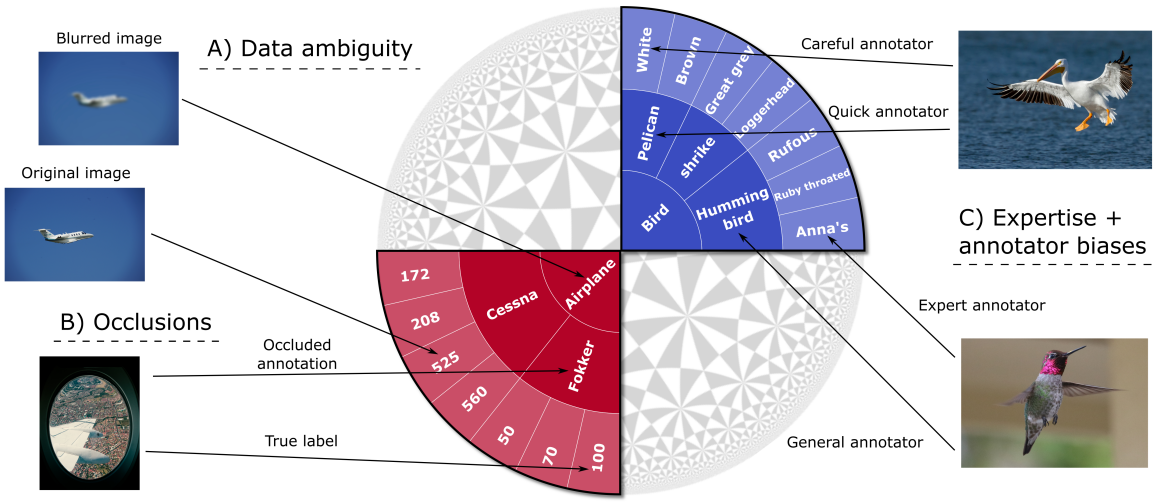

Figure 1: Supervision naturally happens at *any granularity*. In real-world setups, annotations are rarely confined to leaf labels: ambiguity, occlusion, or annotator expertise often result in labels from coarser levels. We embrace this reality and design a model that learns effectively from supervision at any granularity.

label structure and achieve a uniform action learning system in hyperbolic space. Atigh et al. (2022) propose a hyperbolic image segmentation framework in which hierarchical pixel labels are embedded into hyperbolic space, followed by classification performed in the same space. Liu et al. (2020) use the label hierarchy embeddings in hyperbolic space to generalize from seen classes to unseen classes in hyperbolic space. While these works follow the hierarchical structure of the classes, they are ultimately only interested in the leaf classes of the hierarchy. More specifically, each sample will always have a label from the set of leaf classes (Atigh et al., 2022; Liu et al., 2020; Long et al., 2020). The internal nodes are only used to help define the similarity between the leaf classes. In this work, we go beyond hierarchical learning focused solely on leaf classes and investigate how to train models when labels are given at any level of the hierarchy.

This paper introduces coarse-to-fine Busemann optimization for multi-granular hyperbolic learning. We first embed the class hierarchy in a hyperbolic embedding space in the form of prototypes. For each sample, we define a target by projecting it as an ideal prototype on the boundary of the Poincaré ball, *i.e.*, the unit-norm point lying in the same direction as the class prototype on the boundary of the Poincaré ball, which defines the direction to which the sample must be optimized. Our loss integrates a Busemann objective with a set-based contrastive objective and hierarchical entailment to allow for learning at all levels of granularity. Our loss follows the intuition that the more specific a sample's label, the more narrowly defined the region in the embedding space should be for optimization. For example, a sample labelled *Laysan albatross* contains more information than one labeled *bird* and should therefore exert a stronger influence on model training. At the same time, samples with only coarse labels such as *bird* still provide useful supervisory signals that the model can exploit. Experiments on well-known vision datasets show that our approach is capable of handling labels at all levels of granularity. Moreover, our approach results in better classification performance compared to baseline and state-of-the-art approaches, which structurally view labels or different levels of granularity as independent, ignoring the underlying hierarchical structure. Furthermore, our model leverages incomplete supervision to strengthen its internal representations, performing well in filling missing fine-grained labels, laying the groundwork for applications in active and semi-supervised learning.

2 RELATED WORK

2.1 Hierarchical classification

A large body of work in hierarchical classification has leveraged taxonomic structures such as WordNet (Fellbaum, 2010), domain-specific ontologies (Kasarla et al., 2025a), or reorganized hierarchies (Ayoughi et al., 2025b; Zhou et al., 2022; Zheng et al., 2019). Such approaches typically exploit hierarchical relations between classes to enforce consistency in predictions or to transfer information across related categories (Silla Jr & Freitas, 2011). For example, BIOCLIP is a vision foundation model that learns hierarchical representations conforming to the tree of life

ontology and excels at fine-grained biology classification tasks (Stevens et al., 2024). Similarly, Kim et al. (2025) apply transfer learning for hierarchical classification of Amazon parrot species, while Giunchiglia & Lukasiewicz (2020) and Park et al. (2025) exploit hierarchical constraints to ensure prediction coherence and visual grounding consistency across hierarchy levels. While effective, the hierarchical organization primarily serves to define relations between leaf classes, with internal nodes serving as structural anchors rather than independent prediction targets. As a result, intermediate concepts encoded in the hierarchy remain underutilized, even though they could provide semantically meaningful abstractions and improve interpretability.

2.2 Multi-granular classification

Multi-granular classification explicitly considers predictions at multiple levels of abstraction. Such problems arise in diverse domains, including text (Mayne & Perry, 2009; Chen et al., 2020; Giunchiglia & Lukasiewicz, 2020; Mekala et al., 2021), genomics (Schietgat et al., 2010; Barutcuoglu et al., 2006; Romero et al., 2023), and medical imaging (Dimitrovski et al., 2011; Jin et al., 2024; Dimitrovski et al., 2012). Methods to address this challenge often map the label hierarchy into network architectures (Cerri et al., 2014, 2016; Wehrmann et al., 2018) or incorporate it into loss functions that impose hierarchical constraints (Giunchiglia & Lukasiewicz, 2020; Deng et al., 2014). Chang et al. (2021) propose a hierarchical coarse-to-fine label setup, where the model also generates set of coarse-to-fine-grained labels instead of the fine-grained label. Jiang et al. (2024) integrate a hierarchical network with self-supervised fine-grained training and transfer knowledge from fine to coarse labels via soft-label generation. While both approaches incorporate multi-granular supervision, they assume that annotations at all levels of granularity are available during training, which limits their applicability in more realistic settings where only partial annotations may be present. Chen et al. (2022a) propose a hierarchical residual network with a combinatorial loss that aggregates information across hierarchy levels.While most closely related to our work, the proposed architecture in Chen et al. (2022a) is constrained by a fixed design tied to the number of hierarchy levels and is limited to at most three levels. Moreover, their setup restricts supervision to either leaf labels or their immediate parents. In contrast, our method imposes no restrictions on the depth of the hierarchy, and supervision can be provided at arbitrary levels of granularity.

2.3 Hyperbolic classification

When embedding hierarchies, the geometry of the embedding space is a crucial consideration. Hyperbolic space has the key advantage of embedding tree-like structures with minimal distortion (Sarkar, 2011). This property has made hyperbolic space a powerful tool in performing hierarchical tasks or representing data with a hierarchical structure and has motivated a growing line of work that leverages hyperbolic geometry to represent semantic hierarchies across different data types, including language (He et al., 2025a; Dhingra et al., 2018; Tifrea et al., 2018; Zhu et al., 2020; Chen et al., 2022b), graphs (Chlenski & Pe'er, 2025; Li et al., 2024a; Liu et al., 2019; Chami et al., 2019; Zhang et al., 2021; Yang et al., 2022), biological data (Yang et al., 2025; Wang et al., 2025), and vision (Khrulkov et al., 2020; Wang et al., 2024). We refer to surveys on hyperbolic learning for a deeper discussion He et al. (2025b); Mettes et al. (2024); Fang et al. (2023); Peng et al. (2021); Yang et al. (2022).

In computer vision, hyperbolic space has been explored for a wide range of tasks including classification (Woo et al., 2025; Khrulkov et al., 2020; van Spengler et al., 2023a), visual question answering (Chen et al., 2025), continual learning (Ayoughi et al., 2025a), segmentation (Hindel et al., 2024; Sur et al., 2025; Atigh et al., 2022; Chen et al., 2023), zero-shot recognition (Han et al., 2025; Atigh et al., 2025), and vision–language modeling (Pal et al., 2025; Poppi et al., 2025; Ibrahimi et al., 2024; Mandica et al., 2024). A common approach in hyperbolic learning is to embed any prior knowledge regarding the classes directly into the representation space through the use of class prototypes (Ayoughi et al., 2025a; Berg et al., 2025; Ghadimi Atigh et al., 2021; Yu et al., 2022; Mettes et al., 2024; Kasarla et al., 2025b). Images can then be classified by extracting hyperbolic image features and determining the similarity of these features to each of the class prototypes, similar in spirit to Snell et al. (2017). In this work, we follow the prototype-based perspective in hyperbolic learning to enable learning at all granularities of a semantic hierarchy.

A limitation of current hyperbolic classification methods is that samples are labelled at the same level of granularity. In practice, datasets often contain annotations at multiple levels of granularity, ranging from coarse to fine categories. Our work addresses this gap by extending hyperbolic classification to the multi-granular setting, enabling models to learn effectively from both coarse and fine-grained supervision.

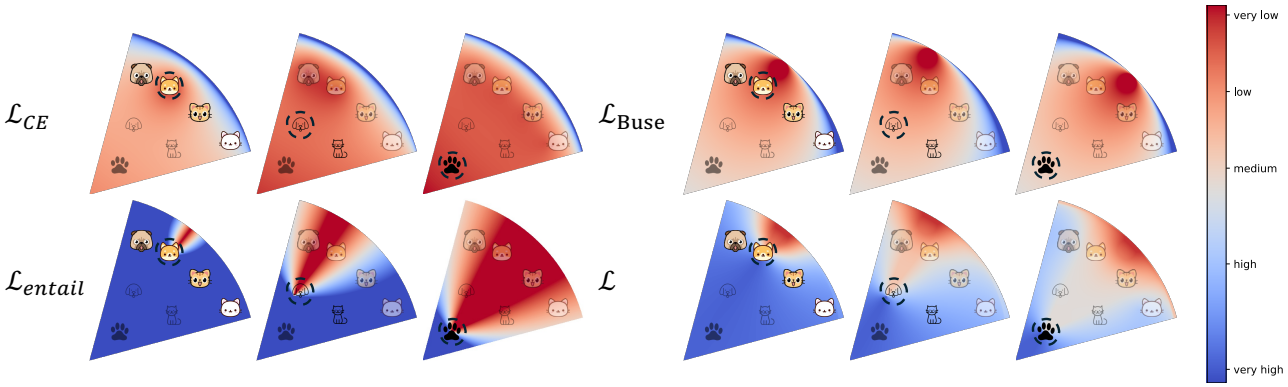

Figure 2: Visualization of our proposed losses. Our contrastive objective ($\mathcal{L}_{\text{CE}}$) pushes samples to the correct leaf prototype. For internal nodes, all corresponding leafs become the prototypes. Our partial order loss ($\mathcal{L}_{\text{entail}}$) forces angular alignment between images and the cone spanned by the class label at hand. Finally, our Busemann loss ($\mathcal{L}_{\text{Buse}}$) projects the label at hand to an ideal prototype to force samples towards the boundary with a desirable angular direction.

3 METHOD

3.1 Problem setup

In the usual classification setting, we are given a dataset $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{N}$ of N training samples, with the i^{th} sample consisting of input x_i and label y_i, and the objective is to predict y_i from x_i. In hierarchical learning (Weber et al., 2024; Long et al., 2020; Liu et al., 2020; Atigh et al., 2022), the labels are part of a hierarchy that can be represented in the form of a directed tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$. This hierarchy is typically separated into leaf nodes $\mathcal{V}_{\text{leaf}}$ and internal nodes $\mathcal{V}_{\text{internal}}$. For each sample i, the label y_i lies in $\mathcal{V}_{\text{leaf}}$, which means that the internal classes only serve to provide the hierarchical relations between leaf classes. The internal classes are not classification targets themselves. Note that standard supervised classification is a special case of hierarchical classification with $\mathcal{E} = \emptyset$ and $\mathcal{V}_{\text{internal}} = \emptyset$.

In this work, we aim for multi-granular classification, where we have multi-granular labels $y \in \mathcal{V} \setminus v_0$, with v_0 being the root of the hierarchy which inherently cannot be used as label since it is fully non-discriminative. In other words, in this setting we have annotations at any granularity and the objective is to leverage these multi-granular annotations to train a model capable of accurately predicting labels at the highest granularity, *i.e.,* $\mathcal{V}_{\text{leaf}}$. For label y, let $\mathcal{F}(y, \mathcal{T})$ denote a function that returns the set of all leaf classes under label y. If $y \in \mathcal{V}_{\text{internal}}$, the function returns all classes at the end of $\mathcal{T}$ starting from y, (i.e., $\{v \in \mathcal{V}_{\text{leaf}} : v \text{ is a descendant of } y\}$). If $y \in \mathcal{V}_{\text{leaf}}$, the function simply maps to $\{y\}$. During inference, each sample is assigned to one of the leaf classes. Below, we outline how a model can be trained on multi-granular data to maximize the performance on leaf classes during inference.

3.2 Coarse-to-fine Busemann learning

Our goal is to learn a model that can leverage supervision at any granularity by embedding both label hierarchy and image representations in a shared hyperbolic space. To achieve this goal, there are three components: (i) class representations obtained by embedding the hierarchy into hyperbolic space, (ii) corresponding ideal prototypes, and (iii) an optimization procedure that aligns image features with their corresponding class prototypes under multi-granular supervision.

Preliminaries. We formulate our method in the Poincaré ball model of hyperbolic space. The d-dimensional Poincaré ball with constant negative curvature -1 is defined as the Riemannian manifold $(\mathbb{B}^d, \mathfrak{g})$, where

$$\mathbb{B}^d = \{\mathbf{x} \in \mathbb{R}^d : ||\mathbf{x}||^2 < 1\}, \tag{1}$$

and where

$$\mathfrak{g} = \lambda_{\mathbf{x}}^2 I_d, \quad \lambda_{\mathbf{x}} = \frac{2}{1 - ||\mathbf{x}||^2}, \tag{2}$$

where I_d is the d-dimensional identity matrix.

A point $\mathbf{v} \in \mathbb{R}^d$ can be mapped to the Poincaré ball by considering it to be a tangent vector at the origin and using the corresponding exponential map,

$$\exp_{\mathbf{0}}(\mathbf{v}) = \tanh(||\mathbf{v}||)\frac{\mathbf{v}}{||\mathbf{v}||}. \tag{3}$$

The hyperbolic distance between two points $\mathbf{x}, \mathbf{y} \in \mathbb{B}^d$ is given by

$$d_{\mathbb{B}}(\mathbf{x}, \mathbf{y}) = \operatorname{arcosh}\big(1 + 2\frac{\|\mathbf{x} - \mathbf{y}\|^2}{(1 - \|\mathbf{x}\|^2)(1 - \|\mathbf{y}\|^2)}\big). \quad (4)$$

For a more comprehensive introduction to hyperbolic geometry, we refer to (Ramsay & Richtmyer, 2013; Mettes et al., 2024).

Hyperbolic prototype embeddings. We first embed the hierarchy $\mathcal{T}$ into a d-dimensional Poincaré ball $\mathbb{B}^d$. Rather than initializing hierarchy nodes with a Gaussian distribution in hyperbolic space(Nickel & Kiela, 2017), we propose to position leaf nodes $\mathcal{V}_{\text{leaf}}$ first with maximum separation (Kasarla et al., 2022). This ensures that leaf classes will not collapse when optimized hierarchically in the next step. Let $c = |\mathcal{V}_{\text{leaf}}|$ denote the number of leaf classes. Then the maximally separated initialization is given in $(c-1)$-dimensional space as:

$$P_{c-1} = \begin{pmatrix} 1 & -\frac{1}{c-1}\mathbf{1}^T \\ \mathbf{0} & \sqrt{1 - \frac{1}{(c-1)^2}}\, P_{c-2} \end{pmatrix} \in \mathbb{R}^{c \times (c-1)}, \quad (5)$$

where $P_1 = (1 \quad -1) \in \mathbb{R}^{1\times 2}$, following Kasarla et al. (2022). Starting from these maximally separated representations for $\mathcal{V}_{\text{leaf}}$, the final node representations for all nodes $\mathcal{V} = \mathcal{V}_{\text{leaf}} \cup \mathcal{V}_{\text{internal}}$ are learned by minimizing the distortion through stochastic gradient descent (Sala et al., 2018). The distortion is given by

$$D(f) = \frac{1}{\binom{|\mathcal{V}|}{2}} \left(\sum_{u,v \in V : u \neq v} \frac{|d_{\mathbb{B}}(f(u), f(v)) - d_{\mathcal{T}}(u, v)|}{d_{\mathcal{T}}(u, v)} \right), \quad (6)$$

where $f : \mathcal{V} \to \mathbb{B}^d$ is our embedding which maps nodes into hyperbolic space and where $d_{\mathcal{T}}$ is the tree metric induced from the undirected version of $\mathcal{T}$. Intuitively, minimizing the distortion ensures that distances between the embedded nodes closely resemble the distances between the nodes in the original tree, leading to minimal information loss of the embedding. We find that internal nodes do not require maximum separation for initialization, this step is only beneficial for the leaf nodes. Internal nodes can instead be uniformly distributed as $\mathcal{U}(-0.001, 0.001)$.

Optimization with multi-granular labels. To extract features of the input image, a base network $\phi_\theta(\cdot)$ maps an input image to a hyperbolic feature representation

$$\mathbf{z}_i = \exp_0(\phi_\theta(x_i)). \quad (7)$$

Given this image representation $\mathbf{z}_i$ and class embeddings $\{V_j\}_{j=1}^c = f(\mathcal{V}_{\text{leaf}})$, the model predicts a distribution over classes by comparing $\mathbf{z}_i$ to each class embedding using the negative of the hyperbolic distance $-d_{\mathbb{B}}(\cdot, \cdot)$ and softmaxing over these similarities. Specifically, we define

$$\hat{p}_{i,j} = \frac{\exp\big(-d_{\mathbb{B}}(\mathbf{z}_i, V_j)\big)}{\sum_{k=1}^{c} \exp\big(-d_{\mathbb{B}}(\mathbf{z}_i, V_k)\big)}. \quad (8)$$

Given image x_i with label y_i, the target probability t_i is defined by

$$t_{i,j} = \begin{cases} 1, & \text{if } y_i \in \mathcal{V}_{\text{leaf}} \text{ and } j = y_i, \\ \dfrac{1}{|\mathcal{F}(y_i, \mathcal{T})|}, & \text{if } y_i \notin \mathcal{V}_{\text{leaf}} \text{ and } j \in \mathcal{F}(y_i, \mathcal{T}), \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

This definition states that the likelihood should be 0 for an irrelevant label, and 1 if the ground truth label is a leaf class and the prediction is a match. For labels that are internal nodes, the likelihood is given as a uniform distribution that smooths the likelihood over all leaf classes that fall under the leaf label. With this definition our first part of the optimization is given as a conventional cross-entropy loss over prototype logits, as visualized in Figure 2:

$$\mathcal{L}_{\text{CE}} = -\sum_{j=1}^{c} t_{i,j} \log \hat{p}_{i,j}. \quad (10)$$

This loss encourages the predicted distribution $\hat{p}_i$ to match the target distribution t_i.

Alongside $\mathcal{L}_{\text{CE}}$, which optimizes for predicting the target distribution, our second loss optimizes for partial order in the label hierarchy through entailment cones. Hyperbolic entailment cones embed partially ordered sets, such as hierarchical structures, into hyperbolic space (Ganea et al., 2018). Given a class prototype representation $f(y_i)$ and an image representation $\mathbf{z}_i$, the goal is to learn $\mathbf{z}_i$ such that it is inside the entailment cone of $f(y_i)$. The entailment cone at $f(y_i)$ is defined as the geodesic cone with $f(y_i)$ as its apex, the spoke segment from $f(y_i)$ perpendicular onto the boundary of the manifold as its axis of symmetry, and half its aperture given by

$$\psi(f(y_i)) = \arcsin\Big(K\frac{1 - ||f(y_i)||^2}{||f(y_i)||}\Big), \quad (11)$$

where K is a constant parameter set to 0.1. Given this definition, the smallest angle of a rotation around $f(y_i)$ that would bring $\mathbf{z}_i$ into the cone can be computed as

$$\mathcal{L}_{entail}(f(y_i), \mathbf{z}_i) = \max\big(0, \Xi(f(y_i), \mathbf{z}_i) - \psi(f(y_i))\big), \quad (12)$$

also visualized in Figure 2. With $l_i := f(y_i)$ for clarity,

$$\Xi(l_i, \mathbf{z}_i) = \arccos\left(\frac{\langle l_i, \mathbf{z}_i\rangle(1+\|l_i\|^2) - \|l_i\|^2(1+\|\mathbf{z}_i\|^2)}{\|l_i\| \cdot \|l_i - \mathbf{z}_i\|\sqrt{(1+\|l_i\|^2)\|\mathbf{z}_i\|^2 - 2\langle l_i, \mathbf{z}_i\rangle}}\right), \tag{13}$$

denotes the angle between the geodesic segment from $f(y_i)$ to $\mathbf{z}_i$ and the cone's axis of symmetry. As $\mathcal{L}_{entail}$ is the value that Ganea et al. (2018) propose for quantifying how far $\mathbf{z}_i$ is removed from occurring within the cone, we propose to use this as part of our loss.

The cross-entropy and entailment losses guide the samples to the correct part of the space, but might be biased towards internal classes if they occur often in the training set. To help guide samples towards the edge of the Poincaré ball, where leaf classes are positioned, we propose to guide the model optimization by computing the corresponding *ideal prototype* $\mathbb{I}_{y_i}$. Ideal prototypes are defined as the normalized prototype projected onto the boundary of the hyperbolic space:

$$\mathbb{I}_{y_i} = \frac{f(y_i)}{\|f(y_i)\|}. \tag{14}$$

Since the space is organized such that general concepts lie closer to the origin and more specific concepts closer to the boundary, the goal is to learn image representations that lie beyond their corresponding class embedding, as these images can be seen as highly specific instances of their class. Ideal prototypes can serve as counterbalances. Since geodesic distances to boundary points are infinite, we instead adopt the *Busemann loss* (Ghadimi Atigh et al., 2021), which provides a tractable distance-to-infinity measure. Specifically, for image representation $\mathbf{z}_i$ and corresponding ideal prototype $\mathbb{I}_{y_i}$, the penalized Busemann loss is defined as

$$\mathcal{L}_{\text{Buse}}(\mathbb{I}_{y_i}, \mathbf{z}_i) = \log \frac{\|\mathbb{I}_{y_i} - \mathbf{z}_i\|^2}{1 - \|\mathbf{z}_i\|^2} - v_0(d) \cdot \log(1 - \|\mathbf{z}_i\|^2), \tag{15}$$

where $v_0(d)$ is a dimension-dependent scaling factor. Figure 2 shows the cost landscape for this loss. The loss pulls embeddings towards the ideal prototype $\mathbb{I}_{y_i}$ while regularizing against collapse at the boundary. The final objective adds the three loss terms from equations 10, 12 and 15:

$$\mathcal{L}(\mathcal{D}_{train}) = \frac{1}{N}\sum_{i=1}^{N}\Big(\mathcal{L}_{\text{CE}}(t_i, \hat{p}_i) + \mathcal{L}_{entail}(f(y_i), \mathbf{z}_i) + \lambda_{Buse}\mathcal{L}_{\text{Buse}}(\mathbb{I}_{y_i}, \mathbf{z}_i)\Big). \tag{16}$$

What makes the three losses complementary is that all have different targets and perspectives: the cross-entropy loss optimizes for hierarchically smoothed likelihoods of leaf classes, the entailment loss optimizes for hierarchical consistency to cones derived from the prototypes, and the Busemann loss optimizes towards an ideal prototype derived from the original prototypes. For fine-grained datasets (e.g., Stanford Cars and FGVC-Aircraft), we set $\lambda_{\text{Buse}} = 1$ for all samples. For CIFAR-100, λ_{Buse} is set to 1 only for samples that satisfy following conditions: (i) they lack fine-grained labels, and (ii) their representations are sufficiently distant from the corresponding class prototype, *i.e.,* samples for which the distance to the prototype exceeds d_B and the angular difference exceeds Ξ_B.

4 SETUP

4.1 Datasets

We evaluate our proposed method on three widely used benchmarks: CIFAR100 (Krizhevsky et al., 2009), FGVC-Aircraft (Maji et al., 2013), and Stanford Cars (Krause et al., 2013). CIFAR100 consists of 60,000 images from 100 fine-grained categories grouped into 20 superclasses. The dataset has a 4-level hierarchy with 128 nodes in total. FGVC-Aircraft contains 10,000 images across 100 aircraft variants, hierarchically organized into families and manufacturers. Its hierarchy spans 4 levels with 201 nodes. Stanford Cars includes 16,185 images covering 196 car models, which are grouped into higher-level categories. The corresponding hierarchy has 3-levels with 206 nodes in total. All hierarchies are from Kasarla et al. (2025a).

4.2 Multi-granular experimental setup

To simulate the multi-granular supervision setting, we construct training sets in which a fixed proportion of examples are annotated with non-leaf labels. Concretely, given the training set $\mathcal{D}_{\text{train}}$ and a ratio parameter $\rho \in [0, 1]$, we randomly select exactly $\rho \cdot |\mathcal{D}_{\text{train}}|$ samples whose fine-grained labels are replaced with internal nodes of the hierarchy $\mathcal{T}$. For each selected sample (x_i, y_i) with $y_i \in \mathcal{V}_{\text{leaf}}$, we uniformly sample an ancestor $\tilde{y}_i \in \text{anc}(y_i)$ from $\mathcal{V}_{\text{internal}} \setminus \phi$, excluding the root, and assign it as the new label. This is a new and challenging setting, which goes beyond the setup of Chen et al. (2022a), where $\tilde{y}_i$ can only be the parent label. Our generalized setup enables learning when the hierarchy $\mathcal{T}$ has arbitrary size and depth. The remaining $(1-\rho) \cdot |\mathcal{D}_{\text{train}}|$ samples preserve their original fine-grained labels. By varying ρ, we systematically evaluate the robustness of our method under increasingly challenging supervision regimes. At test time, all predictions are evaluated against leaf labels to enforce consistency of the evaluation protocol.

Table 1: Classification and hierarchical accuracy (%) for our main ablation study on our three loss components on CIFAR100. A, S, and C denote accuracy, sibling, and cousin accuracy, respectively. The setting without any of our losses corresponds to a hyperbolic prototype approach, which ignores samples without leaf annotations. All our losses matter for multi-granular learning, especially under the most challenging circumstances. All numbers are means over three runs; the full table with mean ± standard deviation is given in the appendix.

$\mathcal{L}_{CE}$	$\mathcal{L}_{entail}$	$\mathcal{L}_{Buse}$	$\rho = 0.5$			$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.95$			$\rho = 0.975$		
			A	S	C	A	S	C	A	S	C	A	S	C	A	S	C
			64.57	75.03	81.83	49.01	61.71	71.81	32.23	45.55	58.06	20.22	32.84	46.79	9.62	20.55	35.72
✓			67.21	80.10	86.45	62.09	79.2	86.50	54.15	77.87	85.88	49.25	75.70	85.49	41.66	75.85	85.83
✓	✓		**67.57**	**80.26**	**86.80**	62.29	**79.4**	**86.84**	55.55	77.30	85.92	50.42	76.51	86.10	42.15	75.14	**86.16**
✓		✓	65.39	78.42	85.43	62.03	77.70	85.46	59.33	78.02	86.01	55.05	77.06	85.82	49.66	76.73	85.86
✓	✓	✓	65.80	78.96	85.86	**63.31**	78.52	85.83	**59.93**	**78.18**	**86.19**	**56.13**	**77.42**	**86.37**	**50.10**	**77.13**	85.79

4.3 Implementation details and evaluation

We employ a ResNet50 backbone trained from scratch as the feature extractor $\phi_\theta(\cdot)$ across all datasets. The embedding space dimension is set to $c - 1$, aligning with the dimensionality of the class prototype embeddings. Hyperbolic operations are implemented using the HypLL library(van Spengler et al., 2023b). Optimization is performed using stochastic gradient descent (SGD) with a learning rate of 0.1, momentum of 0.9, and weight decay of 5×10^{-4}. Models are trained for 200 epochs with batch sizes of 128, 32, and 8 for CIFAR100, FGVC-Aircraft, and Stanford-Cars, respectively. The learning rate is decayed by a factor of 0.2 at epochs 110, 160, and 190. For CIFAR100, we set d_B and Ξ_B to 0.4 and 50°.

We evaluate all models using standard classification accuracy, defined as the fraction of test samples for which the predicted label exactly matches the ground-truth label. To evaluate hierarchical consistency, we consider hierarchical evaluation metrics to account for semantic relationships between classes. Specifically, we measure sibling and cousin accuracy(Hascoet et al., 2019; Long et al., 2020), which reflect predictions that belong to the same parent or grandparent category as the ground-truth label, respectively. These metrics provide additional insight into the model's ability to make semantically reasonable predictions even when the exact class is not correctly predicted.

5 EXPERIMENTS

5.1 Effects of the loss components

In the first experiment, we perform an ablation study on three loss components. We perform this experiment on CIFAR100 across all multi-granular ratios ρ ranging from 0.5 to 0.975. The baseline for this experiment has the same backbone and hyperbolic embedding head as our approach, including the hyperbolic embedding of the class hierarchy. The baseline, however, does not make use of the multi-granular nature of the data, excluding all samples without leaf class annotations.

The standard classification accuracy and the hierarchical accuracy of the first experiment is shown in Table 1. For standard accuracy, we find that the baseline performs reasonable, albeit suboptimal, when 50% of the training set is annotated with internal classes. When this ratio increases, however, the baseline drops rapidly in performance. In comparison, our losses all positively benefit the multi-granular setting. The smoothed likelihood objective of $\mathcal{L}_{CE}$ already provides a major boost, as we are no longer ignoring training samples. The addition of the partial order loss $\mathcal{L}_{entail}$ and ideal prototype loss $\mathcal{L}_{Buse}$ further improves the results. For $\rho = 0.95$, the baseline is stuck at 20.22%, while our losses progressively improve to 49.25%, 50.42%, and 56.13%. We find similar patterns for hierarchical classification accuracy. We conclude that our three losses are all required and beneficial for multi-granular learning in hyperbolic space.

To complement the quantitative findings, we include qualitative results in Figure 3. For instance, when the model misclassifies *willow tree*, it predicts *maple tree*, which is a sibling in the hierarchy. This shows that errors tend to remain semantically consistent and, therefore, less severe.

5.2 Comparative analysis

In the second experiment, we compare our approach to both a naive baseline, which mimics how we currently deal with multi-granular data, and the current state-of-the-art by Chen et al. (2022a). For the naive baseline, we train a model with the same backbone as our approach, but with a standard final fully-connected layer, optimized with cross-entropy on the leaf classes. For this baseline, any sample annotated with an internal class label will have to be ignored. We show the comparison to the naive baseline in top part of the Table 2. Not surprisingly, we find that this baseline performs incredibly poor when a significant portion of the data is no longer annotated with leaf classes. This

Table 2: Comparative results with respect to the naive baseline and Chen et al. (2022a) for hierarchical accuracy. Akin to standard accuracy, our approach is the preferred method for multi-granular learning. All numbers are means over three runs; the full table with mean ± standard deviation is given in the appendix.

	FGVC-Aircraft															Stanford-Cars	
	$\rho = 0.5$			$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.95$			$\rho = 0.975$			$\rho = 0.5$	
	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S
Fully hierarchical																	
Naive	69.42	74.17	78.13	42.66	47.2	55.69	7.83	10.08	19.47	4.14	5.64	12.93	3.12	5.07	13.02	42.30	64.32
Ours	**76.15**	**82.69**	**87.97**	**67.12**	**76.99**	**86.02**	**62.62**	**75.25**	**85.96**	**62.55**	**77.56**	**88.09**	**58.39**	**74.71**	**86.65**	**65.03**	**83.80**
Parent-only																	
Chen et al. (2022a)	65.85	72.33	75.30	49.72	57.47	61.65	23.20	35.16	43.79	15.65	29.58	38.75	11.35	30.28	42.12	19.10	80.87
Ours	**79.15**	**85.03**	**87.61**	**74.23**	**84.97**	**87.85**	**67.54**	**84.58**	**87.61**	**65.89**	**83.92**	**86.92**	**64.63**	**84.88**	**88.03**	**71.47**	**91.36**

Table 3: Classification and sibling accuracy (%) for reclassifying the training set with the model trained on CIFAR100 with various ρ ratios. All numbers are means over three runs; the full table with mean ± standard deviation is given in the appendix.

	$\rho = 0.5$		$\rho = 0.75$		$\rho = 0.90$		$\rho = 0.95$		$\rho = 0.975$	
	A	S	A	S	A	S	A	S	A	S
Naive	83.68	88.95	65.01	74.45	33.66	45.28	20.30	32.14	12.36	22.96
Ours	**87.03**	**95.91**	**80.47**	**94.47**	**74.22**	**93.58**	**69.88**	**93.39**	**62.50**	**93.38**

Figure 3: Success and failure cases of our model trained on CIFAR100 with $\rho = 0.975$. Even when the model misclassifies, *e.g.,* predicting an *otter* as a *beaver*, it is hierarchically consistent and less severe.

result serves to highlight the severity of the problem: we cannot afford to simply ignore samples if we don't like the annotations, we have to make use of them.

We also compare with the approach of Chen et al. (2022a). This approach assumes a fixed three-level hierarchy, where each level has a separate head of the network. The more detailed the annotations, the greater the loss contribution during backpropagation. We are aware that this approach is already a few years old, yet to our knowledge remains the best-performing method for multi-granular learning, highlighting the need for more studies into this challenging problem. Note that in general, the approach of Chen et al. (2022a) cannot handle deeper hierarchies, while our method is agnostic to network depth. The code of their approach only works for the setting where leaf labels are re-assigned to their parent label based on the threshold ρ, a more restrictive setting than for our method. To show that our approach also works in this restrictive setting, we adjust our method to their setup. The results are shown in the bottom parts of the Table 2. We again find that our method obtains superior results. We conclude that our method is the best performing method for multi-granular learning.

5.3 Reclassifying the training sets

In the third experiment, we study whether models trained under different ρ can *reclassify* their own training data, *i.e.,* predict and recover fine-grained labels that were missing during training. This task evaluates how well the model leverages hierarchical information to compensate for incomplete supervision, while also testing whether it overfits to coarse annotations.

Table 3 compares our method against a baseline that ignores all non-leaf labels. The baseline quickly collapses under limited supervision, reaching only 33.66% accuracy at $\rho = 0.90$ and dropping to 20.30% at $\rho = 0.95$, showing signs of overfitting to existing fine-grained labels. In contrast, our method consistently leverages the hierarchy to recover missing annotations, reaching 69.88% at $\rho = 0.95$ and 62.50% at $\rho = 0.975$, compared to just 20.30% and 12.36% for the baseline.

These findings highlight a key strength of multi-granular supervision: beyond improving test-time generalization, it enables models to fill incomplete training labels. This property is particularly appealing for real-world annotation pipelines, where leaf-level labels are costly, and suggests promising directions for active or semi-supervised learning.

6 CONCLUSIONS

This paper shows how hyperbolic space is a natural space for learning from any granularity. Classical supervised learning treats classes as independent, typically optimized with contrastive or cross-entropy objectives. In real-world settings, however, classes are

not independent. Consider, for example, the classification of objects on the road in autonomous driving or the classification of skin lesions in medical imaging. Not only are real-world tasks organized hierarchically, but also annotations are not uniform. Annotators can vary in expertise and bias, while samples can differ in quality and ambiguity. As a consequence, humans naturally annotate at different levels of granularity. Such a multi-granular setting, unfortunately, remains underexplored. Since multi-granular learning is inherently hierarchical, modeling the problem in hyperbolic space is a promising direction, due to the hierarchical properties of the space itself. We propose a new method for learning from any granularity in hyperbolic space. Our approach combines a prototypical loss to hierarchical embeddings with an entailment loss to preserve partial order and a Busemann loss to ensure specificity. Experiments show that all three components matter for multi-granular learning, and our final approach not only improves over the current state-of-the-art, but is also more flexible and opens new opportunities for label completion, relevant for semi-supervised and active learning settings.

7 ACKNOWLEDGMENTS

Mina Ghadimi Atigh acknowledges support from the ClickNL project SustainAV: Sustainable Archives for Future Creation (CI25012). This work was also partially supported by the EU's Horizon Europe research and innovation programme through the ENEXA project (grant agreement no. 101070305). Max van Spengler acknowledges support from the University of Amsterdam Data Science Centre. Teng Long acknowledges support from the VI.Vidi.223.166 project of the NWO Talent Programme, partly financed by the Dutch Research Council (NWO).

References

Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. Hyperbolic image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4453–4462, 2022.

Mina Ghadimi Atigh, Stephanie Nargang, Martin Keller-Ressel, and Pascal Mettes. Simzsl: Zero-shot learning beyond a pre-defined semantic embedding space. *IJCV*, 2025.

Melika Ayoughi, Mina Ghadimi Atigh, Mohammad Mahdi Derakhshani, Cees GM Snoek, Pascal Mettes, and Paul Groth. Continual hyperbolic learning of instances and classes. *arXiv preprint arXiv:2506.10710*, 2025a.

Melika Ayoughi, Max van Spengler, Pascal Mettes, and Paul Groth. Designing hierarchies for optimal hyperbolic embedding. In *European Semantic Web Conference*, pp. 362–382. Springer, 2025b.

Zafer Barutcuoglu, Robert E Schapire, and Olga G Troyanskaya. Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.

Paul Berg, Léo Buecher, Björn Michele, Minh-Tan Pham, Laetitia Chapel, and Nicolas Courty. Multi-prototype hyperbolic learning guided by class hierarchy. *International Journal of Computer Vision*, pp. 1–16, 2025.

Ricardo Cerri, Rodrigo C Barros, and André CPLF De Carvalho. Hierarchical multi-label classification using local neural networks. *Journal of Computer and System Sciences*, 80(1):39–56, 2014.

Ricardo Cerri, Rodrigo C Barros, André C PLF de Carvalho, and Yaochu Jin. Reduction strategies for hierarchical multi-label classification in protein function prediction. *BMC bioinformatics*, 17(1):373, 2016.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

Dongliang Chang, Kaiyue Pang, Yixiao Zheng, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Your" flamingo" is my" bird": Fine-grained, or not. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11476–11485, 2021.

Bike Chen, Wei Peng, Xiaofeng Cao, and Juha Röning. Hyperbolic uncertainty aware semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 25(2):1275–1290, 2023.

Boli Chen, Xin Huang, Lin Xiao, Zixin Cai, and Liping Jing. Hyperbolic interaction model for hierarchical multi-label classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7496–7503, 2020.

Jingzhou Chen, Peng Wang, Jian Liu, and Yuntao Qian. Label relation graphs enhanced hierarchical residual network for hierarchical multi-granularity classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4858–4867, 2022a.

Shangyu Chen, Pengfei Fang, Mehrtash Harandi, Trung Le, Jianfei Cai, and Dinh Phung. Hvq-vae: Variational auto-encoder with hyperbolic vector quantisation. *Computer Vision and Image Understanding*, pp. 104392, 2025.

Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully

hyperbolic neural networks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5672–5686, 2022b.

Philippe Chlenski and Itsik Pe'er. Even faster hyperbolic random forests: A beltrami-klein wrapper approach. *arXiv preprint arXiv:2506.04360*, 2025.

Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *European conference on computer vision*, pp. 48–64. Springer, 2014.

Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. Hierarchical image classification using entailment cone embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 836–837, 2020.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. Embedding text in hyperbolic spaces. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pp. 59–69, 2018.

Ivica Dimitrovski, Dragi Kocev, Suzana Loskovska, and Sašo Džeroski. Hierarchical annotation of medical images. *Pattern Recognition*, 44(10-11):2436–2449, 2011.

Ivica Dimitrovski, Dragi Kocev, Suzana Loskovska, and Sašo Džeroski. Hierarchical classification of diatom images using ensembles of predictive clustering trees. *Ecological Informatics*, 7(1):19–29, 2012.

Pengfei Fang, Mehrtash Harandi, Trung Le, and Dinh Phung. Hyperbolic geometry in computer vision: A survey. *arXiv preprint arXiv:2304.10764*, 2023.

Christiane Fellbaum. Wordnet. In *Theory and applications of ontology: computer applications*. Springer, 2010.

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

Ashima Garg, Depanshu Sani, and Saket Anand. Learning hierarchy aware features for reducing mistake severity. In *European Conference on Computer Vision*, pp. 252–267. Springer, 2022.

Mina Ghadimi Atigh, Martin Keller-Ressel, and Pascal Mettes. Hyperbolic busemann learning with ideal prototypes. *Advances in neural information processing systems*, 34:103–115, 2021.

Eleonora Giunchiglia and Thomas Lukasiewicz. Coherent hierarchical multi-label classification networks. *Advances in neural information processing systems*, 33:9662–9673, 2020.

Linbin Han, Zhi Qiao, Xiantong Zhen, Jiahong Gao, and Zhen Qian. Knowledge-enhanced hyperbolic language-image pretraining for zero-shot learning. In *International Conference on Information Processing in Medical Imaging*, pp. 203–217. Springer, 2025.

Tristan Hascoet, Yasuo Ariki, and Tetsuya Takiguchi. On zero-shot recognition of generic objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Neil He, Rishabh Anand, Hiren Madhu, Ali Maatouk, Smita Krishnaswamy, Leandros Tassiulas, Menglin Yang, and Rex Ying. Helm: Hyperbolic large language models via mixture-of-curvature experts. *arXiv preprint arXiv:2505.24722*, 2025a.

Neil He, Hiren Madhu, Ngoc Bui, Menglin Yang, and Rex Ying. Hyperbolic deep learning for foundation models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 6021–6031, 2025b.

Julia Hindel, Daniele Cattaneo, and Abhinav Valada. Taxonomy-aware continual semantic segmentation in hyperbolic spaces for open-world perception. *IEEE Robotics and Automation Letters*, 2024.

Sarah Ibrahimi, Mina Ghadimi Atigh, Nanne Van Noord, Pascal Mettes, and Marcel Worring. Intriguing properties of hyperbolic embeddings in vision-language models. *TMLR*, 2024.

Juan Jiang, Jingmin Yang, Wenjie Zhang, and Hongbin Zhang. Hierarchical multi-granularity classification based on bidirectional knowledge transfer. *Multimedia Systems*, 30(4):207, 2024.

Cheng Jin, Luyang Luo, Huangjing Lin, Jun Hou, and Hao Chen. Hmil: Hierarchical multi-instance learning for fine-grained whole slide image classification. *IEEE Transactions on Medical Imaging*, 2024.

Tejaswi Kasarla, Gertjan Burghouts, Max Van Spengler, Elise Van Der Pol, Rita Cucchiara, and Pascal Mettes. Maximum class separation as inductive bias in one matrix. *Advances in neural information processing systems*, 35:19553–19566, 2022.

Tejaswi Kasarla, Ruthu Hulikal Rooparaghunath, Stefano D'Arrigo, Gowreesh Mago, Abhishek Jha, Melika Ayoughi, Swasti Shreya Mishra, Ana Manzano Rodríguez, Teng Long, Mina Ghadimi Atigh, et al. Hiervision: Standardized and reproducible hierarchical sources for vision datasets. In *2nd Beyond*

Euclidean Workshop: Hyperbolic and Hyperspherical Learning for Computer Vision, 2025a.

Tejaswi Kasarla, Max van Spengler, and Pascal Mettes. Balanced hyperbolic embeddings are natural out-of-distribution detectors. *arXiv preprint arXiv:2506.10146*, 2025b.

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6418–6428, 2020.

Jung-Il Kim, Jong-Won Baek, and Chang-Bae Kim. Hierarchical image classification using transfer learning to improve deep learning model performance for amazon parrots. *Scientific Reports*, 15 (1):3790, 2025.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1246–1257, 2022.

Yancong Li, Xiaoming Zhang, Ying Cui, and Shuai Ma. Hyperbolic graph neural network for temporal knowledge graph completion. In *LREC-COLING*, 2024a.

Yong-Lu Li, Xiaoqian Wu, Xinpeng Liu, Zehao Wang, Yiming Dou, Yikun Ji, Junyi Zhang, Yixing Li, Xudong Lu, Jingru Tan, et al. From isolated islands to pangea: Unifying semantic space for human action understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16582–16592, 2024b.

Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic visual embedding learning for zero-shot recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9273–9281, 2020.

Yuanpei Liu, Zhenqi He, and Kai Han. Hyperbolic category discovery. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9891–9900, 2025.

Teng Long, Pascal Mettes, Heng Tao Shen, and Cees GM Snoek. Searching for actions on the hyperbole. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1141–1150, 2020.

Avery Ma, Aladin Virmaux, Kevin Scaman, and Juwei Lu. Improving hierarchical adversarial robustness of deep neural networks. *arXiv preprint arXiv:2102.09012*, 2021.

S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.

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.

Andrew Mayne and Russell Perry. Hierarchically classifying documents with multiple labels. In *2009 IEEE symposium on computational intelligence and data mining*, pp. 133–139. IEEE, 2009.

Dheeraj Mekala, Varun Gangal, and Jingbo Shang. Coarse2fine: Fine-grained text classification on coarsely-grained annotated data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 583–594, 2021.

Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. Hyperbolic deep learning in computer vision: A survey. *International Journal of Computer Vision*, 132(9):3484–3508, 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 Maria D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Seulki Park, Youren Zhang, Stella X Yu, Sara Beery, and Jonathan Huang. Visually consistent hierarchical image classification. 2025.

Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic deep neural networks: A survey. *IEEE Transactions on pattern analysis and machine intelligence*, 44(12): 10023–10044, 2021.

Tobia Poppi, Tejaswi Kasarla, Pascal Mettes, Lorenzo Baraldi, and Rita Cucchiara. Hyperbolic safety-aware vision-language models. In *CVPR*, 2025.

Arlan Ramsay and Robert D Richtmyer. *Introduction to hyperbolic geometry*. Springer Science & Business Media, 2013.

Elias Ramzi, Nicolas Audebert, Nicolas Thome, Clément Rambour, and Xavier Bitot. Hierarchical average precision training for pertinent image retrieval. In *European Conference on Computer Vision*, pp. 250–266. Springer, 2022.

Miguel Romero, Felipe Kenji Nakano, Jorge Finke, Camilo Rocha, and Celine Vens. Leveraging class hierarchy for detecting missing annotations on hierarchical multi-label classification. *Computers in Biology and Medicine*, 152:106423, 2023.

Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International conference on machine learning*, pp. 4460–4469. PMLR, 2018.

Rik Sarkar. Low distortion delaunay embedding of trees in hyperbolic plane. In *International symposium on graph drawing*, pp. 355–366. Springer, 2011.

Leander Schietgat, Celine Vens, Jan Struyf, Hendrik Blockeel, Dragi Kocev, and Sašo Džeroski. Predicting gene function using hierarchical multi-label decision tree ensembles. *BMC bioinformatics*, 11(1):2, 2010.

Carlos N Silla Jr and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22 (1):31–72, 2011.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Samuel Stevens, Jiaman Wu, Matthew J Thompson, Elizabeth G Campolongo, Chan Hee Song, David Edward Carlyn, Li Dong, Wasila M Dahdul, Charles Stewart, Tanya Berger-Wolf, et al. Bioclip: A vision foundation model for the tree of life. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19412–19424, 2024.

Tanuj Sur, Samrat Mukherjee, Kaizer Rahaman, Subhasis Chaudhuri, Muhammad Haris Khan, and Biplab Banerjee. Hyperbolic uncertainty-aware few-shot incremental point cloud segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11810–11821, 2025.

Alexandru Tifrea, Gary Becigneul, and Octavian-Eugen Ganea. Poincare glove: Hyperbolic word embeddings. In *International Conference on Learning Representations*, 2018.

Max van Spengler and Pascal Mettes. Low-distortion and gpu-compatible tree embeddings in hyperbolic space. *arXiv preprint arXiv:2502.17130*, 2025.

Max van Spengler, Erwin Berkhout, and Pascal Mettes. Poincare resnet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5419–5428, 2023a.

Max van Spengler, Philipp Wirth, and Pascal Mettes. Hypll: The hyperbolic learning library. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 9676–9679, 2023b.

Jianhui Wang, Wenyu Zhu, Bowen Gao, Xin Hong, Ya-Qin Zhang, Wei-Ying Ma, and Yanyan Lan. Learning protein-ligand binding in hyperbolic space. *arXiv preprint arXiv:2508.15480*, 2025.

Ziwei Wang, Sameera Ramasinghe, Chenchen Xu, Julien Monteil, Loris Bazzani, and Thalaiyasingam Ajanthan. Learning visual hierarchies in hyperbolic space for image retrieval. *arXiv preprint arXiv:2411.17490*, 2024.

Simon Weber, Bar Zöngür, Nikita Araslanov, and Daniel Cremers. Flattening the parent bias: Hierarchical semantic segmentation in the poincaré ball. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28223–28232, 2024.

Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. Hierarchical multi-label classification networks. In *International conference on machine learning*, pp. 5075–5084. PMLR, 2018.

Suhan Woo, Seongwon Lee, Jinwoo Jang, and Euntai Kim. Hypevpr: Exploring hyperbolic space for perspective to equirectangular visual place recognition. *arXiv preprint arXiv:2506.04764*, 2025.

Jingjing Yang, Puyu Han, Qian Liu, Yuhang Pan, Liwenfei He, Lingqiong Zhang, Lingyun Dai, Yongcheng Wang, and Jie Tao. Riemann-gnn: Causal reasoning on hyperbolic riemannian manifolds for interpretable drug-disease prediction. *bioRxiv*, pp. 2025–05, 2025.

Menglin Yang, Min Zhou, Zhihao Li, Jiahong Liu, Lujia Pan, Hui Xiong, and Irwin King. Hyperbolic graph neural networks: A review of methods and applications. *arXiv preprint arXiv:2202.13852*, 2022.

Zhen Yu, Toan Nguyen, Yaniv Gal, Lie Ju, Shekhar S Chandra, Lei Zhang, Paul Bonnington, Victoria Mar, Zhiyong Wang, and Zongyuan Ge. Skin lesion recognition with class-hierarchy regularized hyperbolic embeddings. In *International conference on medical image computing and computer-assisted intervention*, pp. 594–603. Springer, 2022.

Zhen Yu, Toan D Nguyen, Lie Ju, Yaniv Gal, Maithili Sashindranath, Paul Bonnington, Lei Zhang, Victoria Mar, and Zongyuan Ge. Hierarchical skin lesion image classification with prototypical decision tree. *npj Digital Medicine*, 8(1):26, 2025.

Yiding Zhang, Xiao Wang, Chuan Shi, Xunqiang Jiang, and Yanfang Ye. Hyperbolic graph attention network. *IEEE Transactions on Big Data*, 8 (6):1690–1701, 2021.

Yu Zheng, Jianping Fan, Ji Zhang, and Xinbo Gao. Exploiting related and unrelated tasks for hierarchical metric learning and image classification. *IEEE Transactions on Image Processing*, 29:883–896, 2019.

Yu Zhou, Xiaoni Li, Yucan Zhou, Yu Wang, Qinghua Hu, and Weiping Wang. Deep collaborative multitask network: A human decision process inspired model for hierarchical image classification. *Pattern Recognition*, 124:108449, 2022.

Yudong Zhu, Di Zhou, Jinghui Xiao, Xin Jiang, Xiao Chen, and Qun Liu. Hypertext: Endowing fasttext with hyperbolic geometry. *arXiv preprint arXiv:2010.16143*, 2020.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. **Yes**
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Not Applicable
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. All source code with all specifications will be made public for the camera-ready version.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Not Applicable
   (b) Complete proofs of all theoretical results. Not Applicable
   (c) Clear explanations of any assumptions. Not Applicable

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Code and data for the figures will be included in the source code release.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes
   (b) The license information of the assets, if applicable. Not Applicable
   (c) New assets either in the supplemental material or as a URL, if applicable. Yes
   (d) Information about consent from data providers/curators. Not Applicable
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

Table 4: Classification, sibling, and cousin accuracy (%) for reclassifying the training set with a pretrained model under various (ρ) ratios.

	$\rho = 0.90$	$\rho = 0.95$	$\rho = 0.975$
pretrained baseline	55.18	37.8	28.63
Ours	**59.32**	**55.09**	**48.83**

Table 5: Classification, sibling, and cousin accuracy (%) for reclassifying the training set with Euclidean prototypes versus hyperbolic prototypes under various (ρ) ratios.

	$\rho = 0.90$	$\rho = 0.95$	$\rho = 0.975$
Euclidean prototypes	47.8	36.5	28.45
Ours	**59.93**	**56.13**	**50.10**

A APPENDIX

A.1 Additional experiments

In this section, we report the standard deviations for the experiments presented in the main paper. Tables 6, 7, and 8 provide mean ± standard deviation values for the corresponding results in Tables 1, 2, and 3 of the main paper (computed over three independent runs).

A.2 Experiments on CUB200 dataset

We report new results on the CUB200 dataset in Table 9. We follow the same hierarchical evaluation protocol as in the main paper and measure classification (A), sibling (S), and cousin (C) accuracies under different ρ ratios. The results confirm that our method consistently improves over the baseline on CUB200 as well, particularly in the high-ρ regime where supervision is most limited, demonstrating that our approach generalizes to fine-grained datasets.

A.3 Experiments with pretrained backbone

We additionally evaluate our method using a pretrained backbone. The training protocol and hierarchical evaluation remain unchanged from the main setup, and we compare our method against a pretrained baseline in Table 4. Across all ρ values, our method consistently outperforms the pretrained baseline, with gains that become more pronounced as ρ increases. This suggests that our approach provides complementary benefits beyond those offered by pretrained representations.

A.4 Experiments with Euclidean prototypes

To further assess the importance of the underlying geometry, we include an additional Euclidean prototype baseline. This baseline uses the same backbone as our method and replaces the hyperbolic prototypes with Euclidean prototypes, trained using the manifold-agnostic embedding loss of Nickel & Kiela (2017). Since this problem setting is relatively underexplored, there are few established external baselines, and this comparison helps contextualize the benefits of our hyperbolic formulation.

Table 5 reports classification, sibling, and cousin accuracies for the Euclidean prototype baseline and for our hyperbolic approach under different ρ values. Hyperbolic prototypes clearly and consistently outperform their Euclidean counterparts (e.g., 59.93 vs. 47.8 at $\rho = 0.90$), indicating that a Euclidean geometry is not well suited to this hierarchical, multi-granular setting. This observation is consistent with previous findings comparing hyperbolic and Euclidean representations in related tasks (Ganea et al., 2018; Sala et al., 2018).

Table 6: Classification and hierarchical accuracy (%) for our main ablation study on our three loss components on CIFAR100. A, S, and C denote accuracy, sibling, and cousin accuracy, respectively. The setting without any of our losses corresponds to a hyperbolic prototype approach, which ignores samples without leaf annotations. All our losses matter for multi-granular learning, especially under the most challenging circumstances.

$\mathcal{L}_{CE}$	$\mathcal{L}_{entail}$	$\mathcal{L}_{Busc}$	$\rho = 0.5$			$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.95$			$\rho = 0.975$		
			A	S	C	A	S	C	A	S	C	A	S	C	A	S	C
			64.57 ± 0.29	75.03 ± 0.19	81.83 ± 0.16	49.01 ± 1.22	61.71 ± 1.10	71.81 ± 0.72	32.23 ± 0.79	45.55 ± 0.91	58.06 ± 1.04	20.22 ± 0.96	32.84 ± 1.07	46.79 ± 1.25	9.62 ± 0.1	20.55 ± 0.93	35.72 ± 0.93
✓			67.21 ± 0.17	80.10 ± 0.32	86.45 ± 0.22	62.09 ± 0.5	79.2 ± 0.34	86.50 ± 0.4	54.15 ± 0.37	77.87 ± 0.93	85.88 ± 0.46	49.25 ± 0.91	75.70 ± 0.59	85.49 ± 0.32	41.66 ± 0.86	75.85 ± 0.57	85.83 ± 0.28
✓	✓		$\mathbf{67.57 \pm 0.23}$	$\mathbf{80.26 \pm 0.14}$	$\mathbf{86.80 \pm 0.27}$	62.29 ± 0.45	$\mathbf{79.4 \pm 0.34}$	$\mathbf{86.84 \pm 0.29}$	55.55 ± 0.73	77.30 ± 0.66	85.92 ± 0.2	50.42 ± 0.28	76.51 ± 0.31	86.10 ± 0.31	42.15 ± 0.28	75.14 ± 00.00	$\mathbf{86.16 \pm 00.00}$
✓		✓	65.39 ± 0.28	78.42 ± 0.43	85.43 ± 0.40	62.03 ± 0.76	77.70 ± 0.78	85.46 ± 0.52	59.33 ± 0.43	78.02 ± 0.46	86.01 ± 0.38	55.05 ± 0.44	77.06 ± 0.56	85.82 ± 0.47	49.66 ± 0.39	76.73 ± 0.11	85.86 ± 0.29
✓	✓	✓	65.80 ± 0.40	78.96 ± 0.26	85.86 ± 0.18	$\mathbf{63.31 \pm 0.14}$	78.52 ± 0.50	85.83 ± 0.54	$\mathbf{59.93 \pm 0.1}$	$\mathbf{78.18 \pm 0.27}$	$\mathbf{86.19 \pm 0.15}$	$\mathbf{56.13 \pm 0.45}$	$\mathbf{77.42 \pm 0.71}$	$\mathbf{86.37 \pm 0.13}$	$\mathbf{50.10 \pm 0.41}$	$\mathbf{77.13 \pm 0.24}$	85.79 ± 0.24

Table 7: Comparative results with respect to the naive baseline and Chen et al. (2022a) for hierarchical accuracy. Akin to standard accuracy, our approach is the preferred method for multi-granular learning.

Method	$\rho = 0.5$			$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.95$			$\rho = 0.975$			Stanford-Cars $\rho = 0.5$	
	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S
Fully hierarchical																	
Naive	69.42 ± 0.63	74.17 ± 0.41	78.13 ± 0.52	42.66 ± 1.37	47.2 ± 1.18	55.69 ± 0.94	7.83 ± 0.89	10.08 ± 1.12	19.47 ± 0.76	4.14 ± 0.71	5.64 ± 0.95	12.93 ± 1.08	3.12 ± 0.58	5.07 ± 0.83	13.02 ± 0.69	42.30 ± 1.24	64.32 ± 0.88
Ours	$\mathbf{76.15 \pm 0.47}$	$\mathbf{82.69 \pm 0.33}$	$\mathbf{87.97 \pm 0.28}$	$\mathbf{67.12 \pm 0.59}$	$\mathbf{76.99 \pm 0.51}$	$\mathbf{86.02 \pm 0.35}$	$\mathbf{62.62 \pm 0.68}$	$\mathbf{75.25 \pm 0.44}$	$\mathbf{85.96 \pm 0.39}$	$\mathbf{62.55 \pm 0.53}$	$\mathbf{77.56 \pm 0.62}$	$\mathbf{88.09 \pm 0.24}$	$\mathbf{58.39 \pm 0.71}$	$\mathbf{74.71 \pm 0.37}$	$\mathbf{86.65 \pm 0.31}$	$\mathbf{65.03 \pm 0.56}$	$\mathbf{83.80 \pm 0.42}$
Parent-only																	
Chen et al. 2022	65.85 ± 0.74	72.33 ± 0.58	75.30 ± 0.49	49.72 ± 1.16	57.47 ± 0.97	61.65 ± 0.83	23.20 ± 1.05	35.16 ± 0.88	43.79 ± 1.14	15.65 ± 0.92	29.58 ± 1.07	38.75 ± 0.79	11.35 ± 0.86	30.28 ± 0.73	42.12 ± 0.96	19.10 ± 1.33	80.87 ± 0.45
Ours	$\mathbf{79.15 \pm 0.38}$	$\mathbf{85.03 \pm 0.29}$	$\mathbf{87.61 \pm 0.26}$	$\mathbf{74.23 \pm 0.52}$	$\mathbf{84.97 \pm 0.34}$	$\mathbf{87.85 \pm 0.23}$	$\mathbf{67.54 \pm 0.46}$	$\mathbf{84.58 \pm 0.41}$	$\mathbf{87.61 \pm 0.27}$	$\mathbf{65.89 \pm 0.61}$	$\mathbf{83.92 \pm 0.38}$	$\mathbf{86.92 \pm 0.29}$	$\mathbf{64.63 \pm 0.54}$	$\mathbf{84.88 \pm 0.35}$	$\mathbf{88.03 \pm 0.22}$	$\mathbf{71.47 \pm 0.49}$	$\mathbf{91.36 \pm 0.19}$

Table 8: Classification and sibling accuracy (%) for reclassifying the training set with the model trained on CIFAR100 with various ρ ratios.

Method	$\rho = 0.5$		$\rho = 0.75$		$\rho = 0.90$		$\rho = 0.95$		$\rho = 0.975$	
	A	S	A	S	A	S	A	S	A	S
Naive	83.68 ± 0.67	88.95 ± 0.53	65.01 ± 1.04	74.45 ± 0.82	33.66 ± 1.19	45.28 ± 1.07	20.30 ± 0.93	32.14 ± 1.15	12.36 ± 0.88	22.96 ± 0.96
Ours	$\mathbf{87.03 \pm 0.42}$	$\mathbf{95.91 \pm 0.28}$	$\mathbf{80.47 \pm 0.54}$	$\mathbf{94.47 \pm 0.37}$	$\mathbf{74.22 \pm 0.63}$	$\mathbf{93.58 \pm 0.41}$	$\mathbf{69.88 \pm 0.58}$	$\mathbf{93.39 \pm 0.49}$	$\mathbf{62.50 \pm 0.71}$	$\mathbf{93.38 \pm 0.44}$

Table 9: Classification, sibling, and cousin accuracy (%) for reclassifying the training set with the model trained on CUB200 with various ρ ratios.

Method	$\rho = 0.90$			$\rho = 0.95$			$\rho = 0.975$		
	A	S	C	A	S	C	A	S	C
Baseline	27.72 ± 1.08	52.77 ± 1.40	82.27 ± 0.48	17.77 ± 0.39	43.65 ± 1.43	77.82 ± 1.79	11.74 ± 0.82	37.82 ± 0.96	73.59 ± 0.71
Our method	$\mathbf{29.82 \pm 0.91}$	$\mathbf{78.25 \pm 0.67}$	$\mathbf{96.38 \pm 0.31}$	$\mathbf{21.20 \pm 1.04}$	$\mathbf{80.12 \pm 0.59}$	$\mathbf{96.03 \pm 0.29}$	$\mathbf{18.21 \pm 0.89}$	$\mathbf{80.26 \pm 0.54}$	$\mathbf{96.27 \pm 0.27}$