Learning Intrinsic Hierarchy for Generalized Category Discovery

Yu Duan[1], Junzhi He[2], Zhanxuan Hu[3], Mengda Ji[4], Rong Wang[5], Quanxue Gao[1]*

[1]School of Telecommunications Engineering, Xidian University, Xi'an, China
[2]School of Computer Science, Northwestern Polytechnical University, Xi'an, China
[3]School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an, China
[4]School of Information Science and Technology, Yunnan Normal University, Kunming, China
[5]Unmanned System Research Institute, Northwestern Polytechnical University, Xi'an, China
duanyuee@gmail.com, hejunzhi@mail.nwpu.edu.cn, zhanxuanhu@gmail.com,
jimengda@mail.nwpu.edu.cn, wangrong07@tsinghua.org.cn, qxgao@xidian.edu.cn

Abstract

Generalized Category Discovery (GCD) aims to classify unlabeled data by leveraging knowledge from labeled categories. While existing methods have achieved remarkable progress, they often treat images as *flat feature sets*, neglecting the *intrinsic hierarchy*: where key objects dominate meaning and backgrounds serve as context. For instance, in images of a dog either standing on grass or lying on a bed, the dog remains the central semantic element, whereas the background varies. Motivated by this, we propose **LEA**rning Intrinsic **H**ierarchy (**LEAH**), a lightweight module designed to model hierarchical structure within images. LEAH consists of two components: a *pruner* that filters task irrelevant tokens to extract key objects, and a *constructor* that embeds key objects and full images into hyperbolic space using adaptive entailment cones to capture compositional semantics. LEAH can be easily integrated into existing GCD frameworks with minimal modification. When applied to SimGCD, it achieves up to 13.2% accuracy improvement on fine-grained benchmarks, demonstrating its effectiveness in discovering subtle inter-class differences through hierarchical modeling.

Introduction

In recent years, machine learning has made remarkable progress in recognizing objects within a closed-set setting, where all categories are predefined during training. However, real-world scenarios often exists unseen novel categories during model training, needing more flexible and adaptive learning frameworks. Generalized Categories Discovery (GCD) (Vaze et al. 2022) addresses this challenge by enabling models to not only classify instances from known categories but also discover and categorize unseen classes in an open-world setting (Cao, Brbic, and Leskovec 2021). By bridging the gap between supervised learning and unsupervised clustering, GCD aims to enhance the scalability and practicality in dynamic environments.

Existing GCD methods generally follow a two-step pipeline: 1) *representation learning*: training a robust feature extraction backbone such as Vision Transformer (ViT),

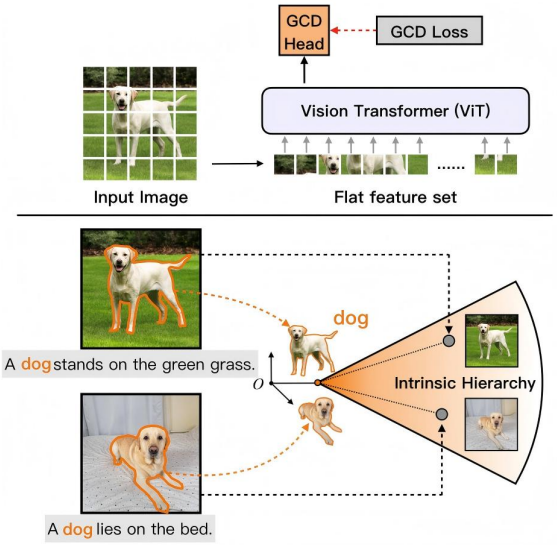

Figure 1: Comparison between traditional flat feature representations and our proposed hierarchical modeling. Unlike previous methods that treat all regions equally, LEAH distinguishes key objects from the background and models their hierarchical relationship with the full image in hyperbolic space to better capture semantic structure.

and 2) *classification*: mainstream methods integrate a parametric classifier with the `[CLS]` token output for end-to-end prediction (Chiaroni et al. 2023; Cao, Brbic, and Leskovec 2021; Rizve, Kardan, and Shah 2022), while others apply heuristic clustering methods directly to `[CLS]` features (Zhang et al. 2023; Pu, Zhong, and Sebe 2023; Miao, Fan, and Xiao 2024; Yin et al. 2023). Recently, several methods have leveraged attention mechanisms to filter image tokens, focusing on salient regions to improve performance. However, all these methods share a fundamental limitation: *they commonly treat input images as flat feature sets, ignoring the intrinsic hierarchical structure within images.* Specifically, the key objects, which carry primary semantic meaning, are treated equally with the background con-

*Corresponding author.

text, missing the critical hierarchical relationship between the two. As shown in Figure 1, while the background context may vary, humans still recognize these images as dogs because the key object, the dog itself, dominates the semantic content. In the context of GCD setting, by establishing a semantic relationship between key objects and the full image, it is better able to utilize context information from similar images, aiding in the recognition and transfer of novel classes with limited labeled data.

Based on this observation, we propose **LEA**rning Intrinsic **H**ierarchy (**LEAH**) for GCD. Intuitively, to model the intrinsic hierarchical structure within images, LEAH mainly consists of two components: the *pruner* and the *constructor*. The *pruner* learns a query vector in each ViT block to filter out task-irrelevant tokens. Next, the *constructor* maps the features into hyperbolic space, and designs adaptive entailment cones to build hierarchical relationships between key object and full images. Finally, in this well-learned geometric representation, key objects are embedded near the origin of the metric space, and the more complex full images progressively positioned towards the boundary, forming a tree-like hierarchy that naturally captures semantic relationships.

Unlike existing methods, LEAH builds clear hierarchical relationships based on image content. It works as a plug-and-play module, which can be plugged into existing GCD models with little extra work. Although easy to use, LEAH significantly improves performance across both general and fine-grained datasets. In summary, we make the following contributions in this paper:

- To the best of our knowledge, we are the first to introduce an intrinsic hierarchical structure within images in GCD tasks. It allows for the separation of key objects from irrelevant backgrounds, providing a new perspective for understanding and processing fine-grained image data.
- We propose a concise method, LEAH, which dynamically learns the hierarchical relationships between key objects and complex backgrounds by utilizing adaptive entailment cones in hyperbolic space.
- Through extensive experiments on public GCD benchmarks, LEAH consistently demonstrates its effectiveness and superiority compared to baseline and state-of-the-art methods across multiple datasets, highlighting its potential for application in open-world scenarios.

Related Works

Generalized Categories Discovery

Generalized Categories Discovery (GCD) builds upon the framework of Novel Categories Discovery (NCD) by employing contrastive learning to fine-tune the representation on the target data, and followed by a well-designed classifier to assign unlabeled data into known and unknown categories. From the types of classifier, we could roughly divide GCD methods into two parts: 1) *non-parametric methods* and 2) *parametric methods*.

After learning a good representations, *non-parametric methods* always leverage a heuristic clustering methods to obtain the final label assignments. For instance, GCD (Vaze et al. 2022) employs semi-supervised k-means on all the `[CLS]` features to obtain the final results. DCCL (Pu, Zhong, and Sebe 2023) adopts an alternating approach that implements a dual-layer contrastive learning strategy that operates at both the instance and concept levels. Finally, it also employs semi-supervised k-means to obtain the final clustering results. Moreover, CMS (Choi, Kang, and Cho 2024) incorporates mean shift for contrastive learning to encourage pull similar samples together. It is important to note that while these types of methods can yield competitive results, they pose challenges in practical scenarios and large-scale datasets due to the high computational complexity of non-parametric post-processing methods.

In contrast to *non-parametric methods*, *parametric methods* jointly optimize the parametric classifier with the backbone, directly obtain predictions end-to-end. SimGCD (Wen, Zhao, and Qi 2023) assigns pseudo labels for unlabeled data, reducing the high complexity of the clustering algorithm. Based on SimGCD, PIM (Chiaroni et al. 2023) maximizes mutual information from an information theory perspective to discover novel categories. ProtoGCD (Ma et al. 2025) designs a dual-level adaptive pseudo-labeling mechanism together with two regularization terms to learn more suitable representations for GCD.

Existing methods treat images as flat token sets, ignoring the inherent hierarchical structure within. To address this, we propose LEAH, a framework that explicitly captures hierarchical dependencies within images. By identifying key semantic regions and modeling their intrinsic relationships, LEAH enables improved representation learning for fine-grained categories, leading to better classification performance.

Hyperbolic Representation Learning

Hyperbolic space has gained great improvement in representation learning due to its ability to naturally encode hierarchical structures. Traditional Euclidean spaces often struggle to represent data that exhibit exponential growth patterns, such as trees and nested relationships (Nickel and Kiela 2017). In contrast, hyperbolic geometry provides a framework where distances and volumes can grow exponentially, making it ideal for hierarchical data representation (Gromov 1987).

Recent advancements have leveraged hyperbolic embeddings for various modalities, including language and vision. For instance, Ganea *et al.* (Ganea, Bécigneul, and Hofmann 2018) introduced hyperbolic entailment cones, enabling the modeling of hierarchical relationships among concepts by capturing parent-child dynamics effectively. MERU (Desai et al. 2023) have utilized hyperbolic embeddings to enhance cross-modal relationships, facilitating better alignment between visual and textual data. Furthermore, RegionCLIP (Zhong et al. 2022) and Ge *et al.* (Ge et al. 2023) have explored object-centric hierarchies in images, integrating local information to refine understanding and representation in hyperbolic settings. These methods highlight the contextual relationships in enhancing model performance across various downstream tasks.

Methods

We present LEAH to capture the intrinsic hierarchical structure within images for GCD. In this part, we begin by briefly introducing the necessary background on hyperbolic geometry, followed by a detailed description of our method.

Lorentz Model of Hyperbolic Geometry

Hyperbolic geometry is a non-Euclidean geometry characterized by a constant negative curvature. The resulting space has the desirable property that volumes of subsets can grow exponentially as a function of their radius, making it an ideal choice for learning representations of data with an inherent hierarchical or tree-like structure.

Here, we adopt the Lorentz model $\mathcal{L}$ of hyperbolic geometry to develop our proposed method.

$$\mathcal{L}^n = \left\{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = -1/c, c > 0\right\}. \quad (1)$$

This model represents a hyperbolic space of n dimensions situated on the upper half of a two-sheeted hyperboloid in $\mathbb{R}^{n+1}$ Minkowski space time. Each vector $\mathbf{x} \in \mathbb{R}^{n+1}$ can be expressed as $[\mathbf{x}_s, x_t]$, where $\mathbf{x}_s \in \mathbb{R}^n$ and $x_t \in \mathbb{R}$. Where $\mathbf{x}_s$ is *space*-coordinates and the last dimension x_t is taken as the *time*-axis (Brown and Pooley 2006). For any two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{n+1}$, their inner product is computed as follows:

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = \langle \mathbf{x}_s, \mathbf{y}_s \rangle - x_t \cdot \mathbf{y}_t, \quad (2)$$

where $\langle \cdot, \cdot \rangle$ is Euclidean inner product. The induced *Lorentzian Norm* is defined as $||\mathbf{x}||_{\mathcal{L}} = \sqrt{|\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}}|}$. The Lorentzian distance between points $\mathbf{x}$ and $\mathbf{y}$ in $\mathcal{L}^n$ is given by:

$$d_{\mathcal{L}}(\mathbf{x}, \mathbf{y}) = \sqrt{1/c} \cdot \cosh^{-1}\left(-c \cdot \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}\right). \quad (3)$$

The *exponential map*, denoted as $\exp_{\mathbf{z}} : \mathcal{T}_{\mathbf{z}}\mathcal{L}^n \rightarrow \mathcal{L}^n$, moves all points $\mathbf{z} \in \mathcal{T}_z\mathcal{L}$ to the manifold $\mathcal{L}$. It is defined as following:

$$\mathbf{x} = \exp_{\mathbf{z}}(\mathbf{v}) = \cosh\left(\sqrt{c}||\mathbf{v}||_{\mathcal{L}}\right)\mathbf{z} + \frac{\sinh\left(\sqrt{c}||\mathbf{v}||_{\mathcal{L}}\right)}{\sqrt{c}||\mathbf{v}||_{\mathcal{L}}} \cdot \mathbf{v}. \quad (4)$$

In this work, we focus on these mappings with $\mathbf{z}$ set as the origin of the hyperboloid ($\mathbf{O} = [\mathbf{0}, \sqrt{1/c}]$). The more discussions about Hyperbolic Geometry can be found in the *supplementary materials*.

Problem Formulations and Overview of LEAH

Given a dataset $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$, where $\mathcal{D}_l = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ is a labeled set with known classes $\mathcal{Y}_l$, and $\mathcal{D}_u = \{\mathbf{x}_j\}_{j=1}^M$ is an unlabeled set containing samples from both known and unknown classes. Let $\mathcal{Y}_u$ denote the set of all classes, where $\mathcal{Y}_l \subset \mathcal{Y}_u$. GCD aims to partition $\mathcal{D}_u$ into meaningful clusters corresponding to the unknown categories, guided by supervision from $\mathcal{D}_l$. We assume that the number of unknown classes is known during evaluation.

Next, we provide an overview of the proposed method, LEAH. LEAH consists of two main components: the *pruner* and the *constructor*. The *pruner* estimates the importance of each patch token and filters out irrelevant ones to extract key objects from the image. The *constructor* then builds a hierarchical structure between the extracted key objects and the full image, enhancing performance.

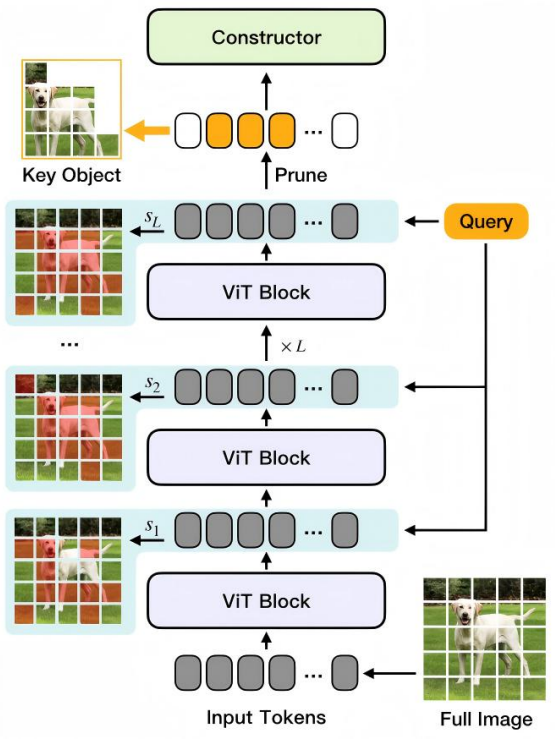

Figure 2: The pipeline of LEAH's pruner. A learnable query is used to compute similarity with the output tokens from each ViT block across multiple layers. Based on these similarities, we identify key object regions in the image. The selected key object tokens, along with the full image tokens, are then passed to a *constructor*.

Pruner: Extract the Key Object

As shown in Figure 2, we adopt a modified version of Cropr (Bergner, Lippert, and Mahendran 2025) as our *pruner* and integrate it into all transformer blocks except the final one. Specifically, in each transformer block, the *pruner* consists of three components: a scorer, an aggregator, and an auxiliary head. Let $\mathbf{s}_i \in \mathbb{R}^{1\times N}$ denote the similarity scores at the i-th block, where N is the number of patch tokens. The scorer computes the similarity between a learnable query $\mathbf{Q}$ and the input tokens $\mathbf{X}$ using scaled cross-attention:

$$\mathbf{s}_i = \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D}}, \quad (5)$$

where $\mathbf{K} = \mathbf{X}$, and $\sqrt{D}$ is a scaling factor. These similarity scores are designed to reflect the semantic importance of each token relative to the learnable query, thereby highlighting regions that are likely to correspond to key objects.

To better guide the query toward attending to these key object regions, the aggregator computes a weighted representation of the input tokens based on $\mathbf{s}_i$:

$$\mathbf{a} = \text{Softmax}(\mathbf{s}_i) \cdot \mathbf{X}. \quad (6)$$

This output is then enhanced via a feed-forward module to capture more discriminative patterns:

$$\mathbf{a}' = \text{MLP}(\text{LN}(\mathbf{a})) + \mathbf{a}. \quad (7)$$

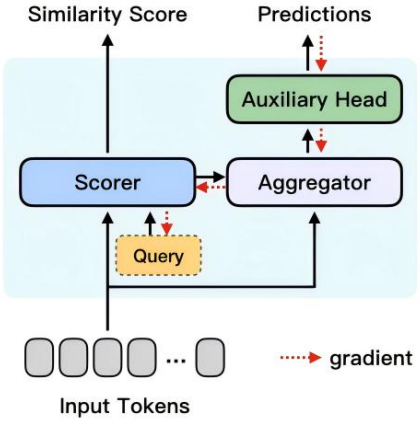

Figure 3: Illustration of the pruner in each ViT block.

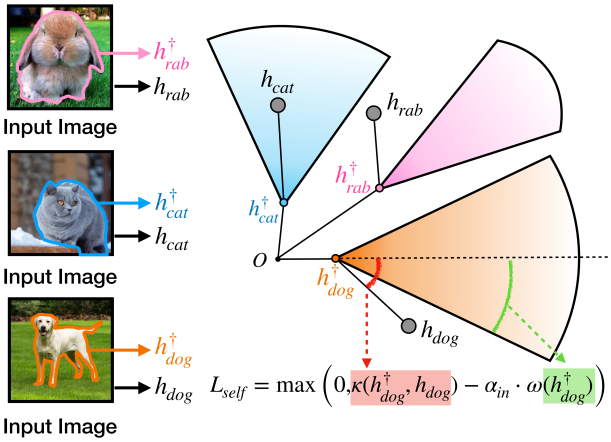

Figure 4: Illustration of the Adaptive Entailment Cone. The *constructor* is designed to enforce that the embeddings of full images from the same class (e.g., dog) lie within the entailment cone defined by the key object. The loss function ensures that intra-class samples are contained within the cone while samples from other classes remain outside, enhancing class separation and hierarchical structures in hyperbolic space.

The resulting representation $\mathbf{a}'$ is passed to an auxiliary head for label prediction, supervised by a cross-entropy loss.

At final ViT block, we average all similarity scores to obtain a final score vector $\hat{\mathbf{s}}$:

$$\hat{\mathbf{s}} = \frac{1}{L-1} \sum_{i=1}^{L-1} \text{Softmax}(\mathbf{s}_i), \tag{8}$$

where L is the total number of transformer blocks. Based on these scores, the pruner selects the top-K tokens $\mathbf{X}^r \in \mathbb{R}^{1\times K}$ via:

$$\mathbf{X}^r = \text{Top-K}(\mathbf{X}|\hat{\mathbf{s}}). \tag{9}$$

These selected tokens correspond to the most semantically informative patches and are thus regarded as the key object representations. The output tokens are aggregated via average pooling to yield the final embedding of the key object.

During training, a stop-gradient is applied before the scorer and aggregator to prevent back-propagation into the backbone. This decouples the auxiliary loss from the main representation learning, allowing the pruner to independently specialize in extracting discriminative object-level regions.

Constructor: Learn the Intrinsic Hierarchy

Here, we will talk about the *constructor*, which maps the features into hyperbolic space, utilizing *entailment cone* build hierarchical relationships between key object and full images. We first introduce the *entailment cone*, and then propose entailment adaptive hierarchal objective functions in hyperbolic space to optimize the hierarchical structures.

Ganea et al. (Ganea, Bécigneul, and Hofmann 2018) introduced hyperbolic *entailment cones* which defines a region $\Gamma_{\mathbf{x}}$, for every possible point $\mathbf{x}$ in the space, all points $\mathbf{y}$ are semantically linked to $\mathbf{x}$ as its child concepts. Formally, considering the Lorentz model $\mathcal{L}^n$, the half-aperture of these regions $\Gamma_{\mathbf{x}}$, is formulated as

$$\omega(\mathbf{x}) = \sin^{-1}\left(\frac{2G}{\sqrt{c}||\mathbf{x}_s||}\right), \tag{10}$$

where a constant $G = 0.1$ is used for setting boundary conditions near the origin. We can see that $\omega(\mathbf{x})$ is inversely correlated with $||\mathbf{x}_s||$. The $\Gamma_{\mathbf{x}}$ becomes larger when $\mathbf{x}$ is close to the origin. In other words, points near the origin represent broader, more general concepts, while those farther from the origin correspond to more specific concepts. Therefore, we need to position key object embeddings closer to the origin and distribute their child concepts (whole images) within these regions. To quantify this hierarchical relationship, we measure the exterior angle $\kappa(\mathbf{x}, \mathbf{y}) = \pi - \angle\mathbf{Oxy}$ as illustrated in Figure 4, which is computed as

$$\kappa(\mathbf{x}, \mathbf{y}) = \cos^{-1}\left(\frac{\mathbf{y}_t + x_t \cdot c\langle \mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}}}{||\mathbf{x}_s||\sqrt{(c\langle \mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}})^2 - 1}}\right). \tag{11}$$

When the exterior angle is smaller than the aperture angle, the relationship between $\mathbf{x}$ and $\mathbf{y}$ is considered satisfied and requires no penalty. Conversely, if the exterior angle exceeds the aperture angle, we must reduce this angular value. Based on this observation, we introduce two objective functions to achieve this goal, which mainly consists of two parts, namely a self-entailment adaptive loss and supervised entailment adaptive loss.

Formally, given the hyperbolic space embeddings of i-th key objects and whole images as $\mathbf{h}^{\dagger}_i$ and $\mathbf{h}_i$ respectively, we define self entailment loss as follow:

$$\mathcal{L}^*_{self} = \max\left(0, \kappa(\mathbf{h}^{\dagger}_i, \mathbf{h}_i) - \omega(\mathbf{h}^{\dagger}_i)\right) \tag{12}$$

Intuitively, $\mathcal{L}_{self}$ only pushes points outside the region to the boundary. However, the *constructor*'s goal is to ensure whole images are distributed within the region's interior. To achieve this, we introduce a learnable threshold $\alpha_{in} \in \mathbb{R}$ and reformulate Eq. (12) as

$$\mathcal{L}_{self} = \max\left(0, \kappa(\mathbf{h}^{\dagger}_i, \mathbf{h}_i) - \alpha_{in} \cdot \omega(\mathbf{h}^{\dagger}_i)\right). \tag{13}$$

Methods	CUB			Stanford Cars			FGVC-Aircraft			CIFAR-10			CIFAR-100			ImageNet-100		
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New
RS+	33.3	51.6	24.2	28.3	61.8	12.1	26.9	36.4	22.2	46.8	19.2	60.5	58.2	77.6	19.3	37.1	61.6	24.8
UNO+	35.1	49.0	28.1	35.5	70.5	18.6	40.3	56.4	32.2	68.6	**98.3**	53.8	69.5	80.6	47.2	70.3	95.0	57.9
ORCA	35.3	45.6	30.2	23.5	50.1	10.7	22.0	31.8	17.1	81.8	86.2	79.6	69.0	77.4	52.0	73.5	92.6	63.9
GCD	51.3	56.6	48.7	39.0	57.6	29.9	45.0	41.1	46.9	91.5	**97.9**	88.2	73.0	76.2	66.5	66.3	89.8	74.1
DCCL	63.5	60.8	64.9	43.1	55.7	36.2	-	-	-	96.3	96.5	96.9	75.3	76.8	70.2	76.2	90.5	**80.5**
GPC	55.4	58.2	53.1	42.8	59.2	32.8	46.3	42.5	47.9	90.6	97.6	87.0	75.4	**84.6**	60.1	66.7	93.4	75.3
PIM	62.7	75.7	56.2	43.1	66.9	31.6	-	-	-	94.7	97.4	93.3	78.3	84.2	66.5	83.1	**95.3**	77.0
InfoSieve	**69.4**	**77.9**	**65.2**	55.7	74.8	46.4	56.3	**63.7**	52.5	-	-	-	**82.3**	85.7	75.5	81.3	**95.6**	74.2
CMS	68.2	**76.5**	64.0	**56.9**	**76.1**	**47.6**	56.0	63.4	52.3	94.8	97.7	93.4	78.3	82.2	70.5	80.5	93.8	73.8
ProtoGCD	63.2	68.5	60.5	53.8	73.7	44.2	**56.8**	62.5	**53.9**	**97.3**	95.3	**98.2**	81.9	82.9	**80.0**	**84.0**	92.2	**79.9**
SimGCD	60.3	65.6	57.7	53.8	71.9	45.0	54.2	59.1	51.8	97.1	95.1	98.1	80.1	81.2	**77.8**	83.0	93.1	77.9
+LEAH	**69.4**	76.3	**65.9**	**65.5**	**80.4**	**58.2**	**58.3**	**64.7**	**55.1**	**97.5**	95.6	**98.5**	**82.1**	**87.0**	72.5	**84.6**	95.1	79.4

Table 1: Comparison results with state-of-the-art methods on six benchmark datasets. The best and second best results are **red bold** and **bold** respectively.

Building on the available labeled data in GCD tasks, we extend $\mathcal{L}_{self}$ to propose a supervised entailment loss. Our key intuition is twofold: *1)* whole images sharing the same label (positive pairs) should be distributed inside the key object region; *2)* whole images from different classes should be outer the region. Formally, this gives rise to the following supervised loss for positive pairs:

$$\mathcal{L}_{pos} = \frac{1}{|\mathcal{P}_i|} \sum_{j \in \mathcal{P}_i} \max\left(0, \kappa(\mathbf{h}_i^{\dagger}, \mathbf{h}_j) - \alpha_{in} \cdot \omega(\mathbf{h}_i^{\dagger})\right), \tag{14}$$

where $|\mathcal{P}|$ denotes the set of positive samples sharing the same label with $\mathbf{h}_i$. Correspondingly, for negative sample pairs, we introduce $\mathcal{L}_{neg}$ as follow:

$$\mathcal{L}_{neg} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \max\left(0, \alpha_{out} \cdot \omega(\mathbf{h}_i^{\dagger}) - \kappa(\mathbf{h}_i^{\dagger}, \mathbf{h}_j)\right), \tag{15}$$

where $\mathcal{N}_i = \{\mathcal{B} \backslash \mathcal{P}_i\}$ is negative sets that shares the different labels with $\mathbf{h}_i$. Finally, We integrate entailment loss into existing GCD works, yielding the final objective function as:

$$\mathcal{L} = \mathcal{L}_{GCD} + \lambda_1 \mathcal{L}_{self} + \lambda_2 \mathcal{L}_{sup} \tag{16}$$

where $\mathcal{L}_{GCD}$ is existing GCD loss function, $\mathcal{L}_{sup} = \mathcal{L}_{pos} + \mathcal{L}_{neg}$, λ_1 and λ_2 are trade-off hyperparameters.

Experiments

Experimental Settings

Datasets. We demonstrate the effectiveness of LEAH on three widely used generic datasets: *CIFAR-10*,*CIFAR-100* (Krizhevsky, Hinton et al. 2009) and *ImageNet-100* (Deng et al. 2009), as well as three challenging fine-grained image classification datasets, including *CUB* (Welinder et al. 2010), *Stanford Cars* (Krause et al. 2013), and *FGVC-Aircraft* (Maji et al. 2013). Each dataset is divided into labeled and unlabeled subsets. Following standard GCD (Vaze et al. 2022) settings, we classify certain categories within each dataset as known and others as unknown. Detailed statistics and dataset separation are summarized in *supplementary materials*.

Metrics. Following standard GCD (Vaze et al. 2022) settings, we compute accuracy (ACC) using the Hungarian algorithm to compare the ground-truth labels with the model's cluster assignments. For clarity and convenience, the accuracy metrics are reported for *All* unlabeled data, along with the subsets corresponding to known and unknown classes, labeled as *Old* and *New* in the tables, respectively.

Baseline Comparisons. We compare our method with a comprehensive set of recent methods for GCD. Specifically, we include the following baselines: RS+ (Han et al. 2021), UNO+ (Fini et al. 2021), ORCA (Cao, Brbic, and Leskovec 2021), GCD (Vaze et al. 2022), DCCL (Pu, Zhong, and Sebe 2023), GPC (Zhao, Wen, and Han 2023), PIM (Chiaroni et al. 2023), InfoSieve (Rastegar, Doughty, and Snoek 2023), CMS (Choi, Kang, and Cho 2024), ProtoGCD (Ma et al. 2025), and SimGCD (Wen, Zhao, and Qi 2023).

Main Results

Results on fine-grained datasets. As shown in Table 1, we highlight the best and second best results in bold. As an excellent baseline method, SimGCD has already achieved strong performance on three datasets. After introducing LEAH into SimGCD, a significant performance improvement can be observed. For instance, on the CUB and Stanford Cars, the *All* performance improves by 9.1% and 11.7%, respectively. In terms of discovering new categories, LEAH achieves an impressive 13.2% improvement on the Stanford Cars. It directly demonstrates the effectiveness of LEAH, sine it constructs hierarchical relationships between key objects and full images, enhancing the discriminative of representations. On the other hand, compared to other state-of-the-art (SOTA) methods, LEAH also conducts remarkable performances. For example, on the CUB dataset, LEAH achieves similar challenging results to InfoSieve. However, LEAH intentionally sacrifices some performance on old categories to focus more on the discovery of novel categories, highlighting the superior generalization.

Results on generic datasets. Meanwhile, we also conducted comparison results on three generic datasets in Table 1, where LEAH also achieved excellent results on the

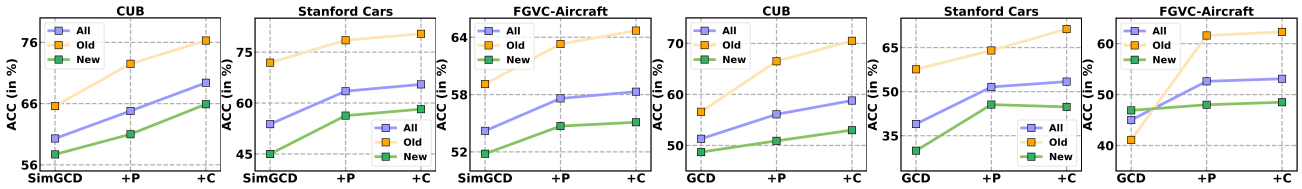

Figure 5: LEAH ablations. The **+P** and **+C** represents the *pruner* and *constructor*, respectively.

Method	CUB		
	All	Old	New
SimGCD+LEAH	69.4	76.3	65.9
fixed α_{in}	68.6$_{-0.8}$	75.6$_{-0.7}$	65.1$_{-0.8}$
fixed α_{out}	68.7$_{-0.7}$	75.9$_{-0.4}$	65.1$_{-0.8}$
fixed all	68.5$_{-0.9}$	76.5$_{+0.2}$	64.5$_{-1.4}$

Table 2: Effectiveness of adaptive entailment cones. By default, we set the fixed parameters $\alpha_{in} = 0.9$ and $\alpha_{out} = 1.5$ for the experiments.

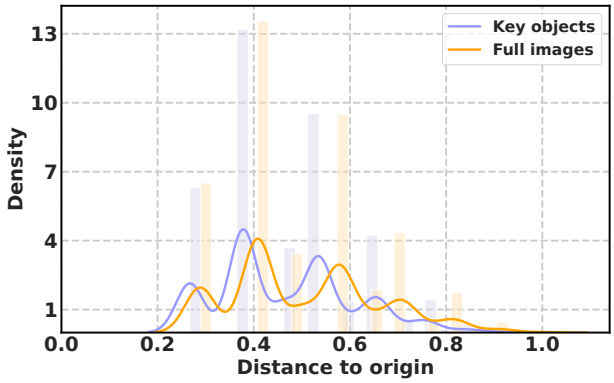

Figure 6: Distribution of embedding distances to origin. LEAH embeds key objects closer to the origin *w.r.t.* full images.

All categories. However, compared to fine-grained datasets, the improvements on these three datasets are relatively limited. We believe that this phenomenon is due to the following reasons: *1)* SimGCD has already performed exceptionally well on these datasets. For example, on CIFAR-10, the *New* has reached 98.1%, leaving little room for LEAH to improve. This also indicates that the different categories in these datasets are well-separated. *2)* Due to their low resolution, CIFAR-10/100 images hinder accurate object localization and hierarchical modeling, degrading LEAH's results.

Ablation Study

As previously mentioned, LEAH is a plug-and-play module. Here, we demonstrate its effectiveness by integrating LEAH with different baselines, and the results are summarized in Figure 5. LEAH consists of two key components: the *pruner* and the *constructor*. Experimental results show that incorporating the *pruner* leads to significant performance gains across all metrics, which strongly validates the importance of focusing on key objects. By filtering out task-irrelevant regions, the pruner effectively enhances the quality of visual representations, especially in fine-grained recognition tasks. This is particularly evident in Figure 5-(f), where the *Old* categories show remarkable improvement. Furthermore, when the *constructor* is added, the model achieves additional performance gains. This suggests that the hierarchical structure built upon the selected key objects not only improves class separation but also enhances generalization to novel categories. It indicates that building hierarchical relationships within images is both reasonable and effective. Together, the *pruner* and *constructor* form a complementary pipeline: one identifies what to focus on, and the other organizes it in a semantically meaningful way.

In-depth Studies

In the following, we provide additional experiments to explore the key features of LEAH in a more detailed manner.

Static v.s. Adaptive Entailment Cone As the most crucial component in hierarchy construction, the entailment cone plays a significant role in LEAH. To this end, we explore the effectiveness of static vs. adaptive entailment cones by fixing the learnable thresholds α_{in} and α_{out} respectively, to highlight the importance of adaptive cones. As shown in Table 2, the best performance is achieved when both dynamic cones are utilized. This observation can be explained by two key factors: *1)* In $\mathcal{L}_{self}$, α_{in} enables the sample distribution to lie within the cones, rather than at the aperture boundary. *2)* In $\mathcal{L}_{sup}$, α_{in} and α_{out} allow samples from the same class to push together within the cones formed by the key objects, while samples from different classes are pushed further away, enhancing class separation.

Additionally, when all parameters are fixed (last row in Table 2), we observe a slight performance improvement on the *Old* categories, but a significant drop in performance on the *New* categories. This can be attributed to the fact that under supervised context, full images tend to be distributed at the boundaries of the cones, degrading the generalization performance.

Visualization of Learned Hyperbolic Space We visualize the learned hyperbolic space to see if the full images and corresponding key objects embeddings are distributed in a

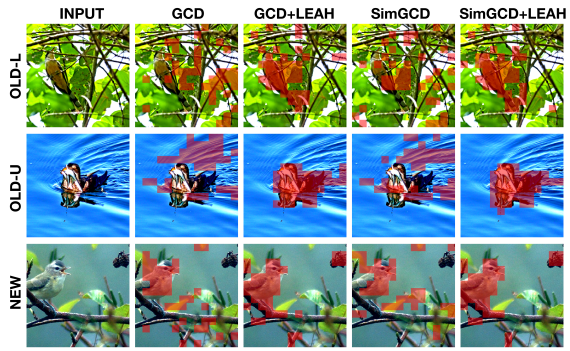

Figure 7: Comparison between [CLS] token and learnable query for key object extraction. We highlight key object regions based on token similarity. LEAH, using learnable queries, more accurately extracts key objects, such as birds, compared to the [CLS] token.OLD-U and OLD-L denote old-unlabeled and old-labeled samples, respec

proper semantic hierarchy. We visualize the learned hyperbolic space to analyze whether the embeddings of full images and their corresponding key objects are distributed in a semantically meaningful hierarchy. Specifically, we computed the distances from both the full images and their key objects to the origin in the hyperbolic space.

As shown in Figure 6, the distribution of key objects and full images is similar, but the key objects tend to be embedded closer to the origin. It indicates that key objects capture more fundamental or essential features of the overall image. Meanwhile, the broader distribution of full images reflects the complexity and diversity of their content. It is consistent with the intuitive understanding of key objects as foundational components within an image's semantic framework.

Why We Use Learnable Queries to Extract Key Objects? As mentioned earlier, the main idea of the *pruner* is to extract key objects from the entire image. Instead of using the traditional [CLS] token, we employ a learnable query to average the tokens of the key objects. To demonstrate the effectiveness, we conduct a visual experiment using two baseline methods (GCD and SimGCD), and compare the similarity between the [CLS] token, learnable query, and other patch tokens. We mark the top 40 tokens with the highest similarity. As shown in Figure 7, LEAH significantly outperforms the [CLS] token in extracting key objects. For instance, in the first row, the [CLS] token fails to accurately locate the bird, whereas LEAH precisely identifies the bird's position. This improvement is crucial for the enhanced performance of LEAH. In the second row, the [CLS] token focuses more on the water surface rather than the key object. Thus, the use of a learnable query allows for more accurate location of key objects, improving the performance and providing more precise input for constructing of hierarchical relationships.

How Many Patch Tokens Should be Retained? As discussed earlier, LEAH selects a subset of informative patch tokens to represent key objects in the full image. One critical factor in this process is determining *how many tokens should be retained?* Since the size and proportion of key objects vary across datasets, the choice of K may significantly affect the model's performance. To explore this, we conduct experiments on three fine-grained datasets—CUB, Stanford Cars, and FGVC-Aircraft—by varying the number of retained tokens $K = \{20, 40, 60\}$. The results are presented in Table 3.

	CUB			Stanford Cars			FGVC-Aircraft		
	All	**Old**	**New**	**All**	**Old**	**New**	**All**	**Old**	**New**
GCD	51.3	56.6	48.7	39.0	57.6	29.9	45.0	41.1	46.9
K=20	54.9	62.2	51.3	46.2	54.4	42.1	45.8	52.0	42.7
K=40	58.8	70.5	53.0	48.1	61.0	41.8	48.8	54.7	45.2
K=60	55.8	64.1	51.6	53.4	71.3	44.8	53.1	62.3	48.5
SimGCD	60.3	65.6	57.7	53.8	71.9	45.0	54.2	59.1	51.8
K=20	55.0	61.6	51.7	62.9	80.4	51.0	58.3	64.7	55.1
K=40	69.4	76.3	65.9	61.6	79.3	53.1	57.6	63.3	54.7
K=60	67.7	74.7	64.2	65.5	80.4	58.2	56.9	66.7	52.1

Table 3: Ablation study on the number of retained patch tokens for key object extraction. The results on *CUB*, *Stanford Cars*, and *FGVC-Aircraft* show that an appropriate choice of K is essential for maximizing performance, with optimal values differing by dataset characteristics.

We observe that retaining 40 tokens achieves the best performance on *CUB* and *FGVC-Aircraft*, while *Stanford Cars* benefits more from retaining 60 tokens. This can be attributed to the relative size of key objects in each dataset: *1)* In *CUB* and *FGVC-Aircraft*, key objects are generally smaller or more localized, so retaining too few tokens may lose important details, while too many can introduce background noise. *2)* In *Stanford Cars*, the key objects usually occupy a larger portion of the full image, requiring more tokens to preserve its spatial structure. These results suggest that carefully choosing K is essential for balancing object coverage and background suppression. An overly small K risks under-representing the object, while a large K may dilute important features with irrelevant background context.

Conclusion

In this paper, we present LEAH, a simple yet effective plug-and-play module that explores the intrinsic hierarchical structure within images. Unlike many existing GCD methods that treat images as flatten sets, LEAH introduces a novel perspective by capturing intra-image structure, which is particularly beneficial for distinguishing subtle inter-class differences. We demonstrate that LEAH consistently improves performance across multiple baselines and achieves competitive results compared to recent state-of-the-art methods. Despite its promising results, LEAH still has limitations. The key object extraction process relies on a manually tuned *pruner*, which may hinder adaptability across datasets with varying object scales and distributions. In addition, the current hierarchical construction is limited to supervised or self-supervised settings, leaving a large portion of unlabeled data in GCD tasks underutilized. Future work will focus on enhancing the adaptability of the pruner and expanding LEAH to fully utilize unlabeled data, further improving its performance and generalization across diverse settings.

Acknowledgments

This work was supported by the National Natural Science Founda tion of China, Grant No. 62176203 and 62576263; the Natural Science Basic Research Program of Shaanxi Province, Grant No. 2025JC-QYCX-051; the Fundamental Research Funds for the Central Universities, the Innovation Fund of Xidian University, Grant No. YJSJ25007, Basic Research Project of Yunnan Province(Grant No.202501CF070004), and Xingdian Talent Support Program.

References

Bergner, B.; Lippert, C.; and Mahendran, A. 2025. Token Cropr: Faster ViTs for Quite a Few Tasks. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 9740–9750.

Brown, H. R.; and Pooley, O. 2006. Minkowski spacetime: A glorious non-entity. *Philosophy and Foundations of Physics*, 1: 67–89.

Cao, K.; Brbic, M.; and Leskovec, J. 2021. Open-world semi-supervised learning. In *International Conference on Learning Representations*.

Chiaroni, F.; Dolz, J.; Masud, Z. I.; Mitiche, A.; and Ben Ayed, I. 2023. Parametric information maximization for generalized category discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1729–1739.

Choi, S.; Kang, D.; and Cho, M. 2024. Contrastive mean-shift learning for generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23094–23104.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Desai, K.; Nickel, M.; Rajpurohit, T.; Johnson, J.; and Vedantam, S. R. 2023. Hyperbolic image-text representations. In *International Conference on Machine Learning*, 7694–7731. PMLR.

Fini, E.; Sangineto, E.; Lathuiliere, S.; Zhong, Z.; Nabi, M.; and Ricci, E. 2021. A unified objective for novel class discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9284–9292.

Ganea, O.; Bécigneul, G.; and Hofmann, T. 2018. Hyperbolic entailment cones for learning hierarchical embeddings. In *International conference on machine learning*, 1646–1655. PMLR.

Ge, S.; Mishra, S.; Kornblith, S.; Li, C.-L.; and Jacobs, D. 2023. Hyperbolic contrastive learning for visual representations beyond objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6840–6849.

Gromov, M. 1987. Hyperbolic groups. In *Essays in group theory*, 75–263. Springer.

Han, K.; Rebuffi, S.-A.; Ehrhardt, S.; Vedaldi, A.; and Zisserman, A. 2021. Autonovel: Automatically discovering and learning novel visual categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 6767–6781.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Ma, S.; Zhu, F.; Zhang, X.-Y.; and Liu, C.-L. 2025. Protogcd: Unified and unbiased prototype learning for generalized category discovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; and Vedaldi, A. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Miao, L.; Fan, W.; and Xiao, H. 2024. Study on Typical Output Scenario Characteristics of Photovoltaic Power Station Based on Improved FCM Clustering. volume 37, 1–11.

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

Pu, N.; Zhong, Z.; and Sebe, N. 2023. Dynamic Conceptional Contrastive Learning for Generalized Category Discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7579–7588.

Rastegar, S.; Doughty, H.; and Snoek, C. 2023. Learn to categorize or categorize to learn? self-coding for generalized category discovery. *Advances in Neural Information Processing Systems*, 36: 72794–72818.

Rizve, M. N.; Kardan, N.; and Shah, M. 2022. Towards realistic semi-supervised learning. In *European Conference Computer Vision*, 437–455. Springer.

Vaze, S.; Han, K.; Vedaldi, A.; and Zisserman, A. 2022. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7492–7501.

Welinder, P.; Branson, S.; Mita, T.; Wah, C.; Schroff, F.; Belongie, S.; and Perona, P. 2010. Caltech-UCSD birds 200.

Wen, X.; Zhao, B.; and Qi, X. 2023. Parametric classification for generalized category discovery: A baseline study. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16590–16600.

Yin, S.; Xiao, Y.; Xu, X.; Ren, H.; and He, Y. 2023. Clustering Identification Method of Household-transformer Relationship Based on Adaptive Piecewise Aggregation Approximation. *Guangdong Electric Power*, 36(02): 76–83.

Zhang, S.; Khan, S.; Shen, Z.; Naseer, M.; Chen, G.; and Khan, F. S. 2023. Promptcal: Contrastive affinity learning via auxiliary prompts for generalized novel category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3479–3488.

Zhao, B.; Wen, X.; and Han, K. 2023. Learning Semi-supervised Gaussian Mixture Models for Generalized Category Discovery. *arXiv preprint arXiv:2305.06144*.

Zhong, Y.; Yang, J.; Zhang, P.; Li, C.; Codella, N.; Li, L. H.; Zhou, L.; Dai, X.; Yuan, L.; Li, Y.; et al. 2022. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16793–16803.