Solving the Catastrophic Forgetting Problem in Generalized Category Discovery

Xinzi Cao[1,2]*, Xiawu Zheng[2,3]*, Guanhong Wang[4]

Weijiang Yu[1], Yunhang Shen[6], Ke Li[6], Yutong Lu[1†], Yonghong Tian[2,8†]

[1] Sun Yat-sen University, [2] Peng Cheng Laboratory, [3] Xiamen University
[4] Zhejiang University, [5] Tencent Youtu Lab, [6] Peking University

caoxz@mail2.sysu.edu.cn   zhengxiawu@xmu.edu.cn   guanhongwang@zju.edu.cn

{weijiangyu8, shenyunhang01}@gmail.com   tristanli@tencent.com

luyutong@mail.sysu.edu.cn   yhtian@pku.edu.cn

Abstract

*Generalized Category Discovery (GCD) aims to identify a mix of known and novel categories within unlabeled data sets, providing a more realistic setting for image recognition. Essentially, GCD needs to **remember** existing patterns thoroughly to recognize novel categories. Recent state-of-the-art method SimGCD transfers the knowledge from known-class data to the learning of novel classes through debiased learning. However, some patterns are catastrophically **forgot** during adaptation and thus lead to poor performance in novel categories classification. To address this issue, we propose a novel learning approach, **LegoGCD**, which is seamlessly integrated into previous methods to enhance the discrimination of novel classes while maintaining performance on previously encountered known classes. Specifically, we design two types of techniques termed as **L**ocal **E**ntropy **R**egularization (LER) and Dual-views Kullback–Leibler divergence c**o**nstraint (DKL). The LER optimizes the distribution of potential known class samples in unlabeled data, thus ensuring the preservation of knowledge related to known categories while learning novel classes. Meanwhile, DKL introduces Kullback–Leibler divergence to encourage the model to produce a similar prediction distribution of two view samples from the same image. In this way, it successfully avoids mismatched prediction and generates more reliable potential known class samples simultaneously. Extensive experiments validate that the proposed LegoGCD effectively addresses the known category forgetting issue across all datasets, e.g., delivering a **7.74%** and **2.51%** accuracy boost on known and novel classes in CUB, respectively. Our code is available at:* https://github.com/Cliffia123/LegoGCD.

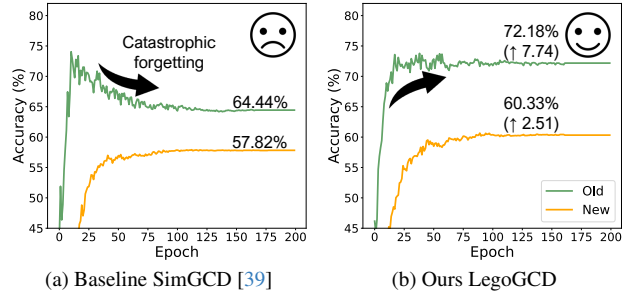*Equal Contribution.
†Joint Corresponding Authors.

Figure 1. Visualization of the accuracy results in unlabeled dataset on CUB dataset [37] during training. (a) shows a decrease in the accuracy of known (Old) classes (green) in the baseline as the accuracy of novel (New) classes (orange) increases. (b) demonstrates that LegoGCD solves the catastrophic forgetting problem and surpasses the baseline by a significant margin of 7.74.

1. Introduction

Deep learning have achieved superior performance on computer vision tasks [4, 11, 24, 25, 30, 34], particularly on image classification [10, 12, 27, 28, 51]. However, conventional methods work in a close-world setting, where all training data comes with pre-defined classes. Consequently, deploying these models in real-world scenarios with potential novel classes becomes a considerable challenge. Furthermore, these achievements rely heavily on large-scale annotated dataset, which is not easily accessible in realistic scenarios. To address these challenges, a new paradigm of *Generalized Category Discovery* (GCD) [1, 7, 9, 23, 36, 39, 45, 46] has been proposed and attracts increasing attention in recent years.

The goal of GCD is to train a classification model capable of recognizing both known and novel categories within unlabeled data. To be clear, GCD distinguishes itself from the Novel Class Discovery (NCD) [8], which relies on an

unrealistic assumption that all unlabeled data exclusively belongs to entirely new classes or patterns. In contrast, GCD adopts a more pragmatic assumption, acknowledging that unlabeled data encompasses a mixture of both known and novel categories. Consequently, GCD is more realistic compared to NCD, especially in real-world scenarios.

Since GCD is partially based on the learned patterns, an intuitive idea is to classify the unlabeled data through a clustering-based approach [36] *i.e.* *k*-means. However, as the scale of datasets increases, the computational costs for clustering in the original GCD grow exponentially. To tackle this issue, Wen *et al.* introduce SimGCD [39], which replaces the clustering approach with a classifier. Specifically, SimGCD trains the classifier using a pseudo-labeling strategy, a technique that has demonstrated remarkable effectiveness in Semi-supervised Learning (SSL) [3, 45]. Nevertheless, the pseudo labels of novel samples tend to be assigned as known classes due to the absence of guidance for novel class samples. In response, Wen *et al.* further propose to adopt class mean entropy to encourage the model to focus on novel categories, consequently generating high-quality pseudo labels for classifier training. As a result, SimGCD has achieved state-of-the-art performance and established itself as a robust baseline solution in the GCD setting.

However, SimGCD [39] still has a significant drawback. It encourages the model to focus more on novel classes by employing an entropy regularization, which unfortunately comes at the cost of known class accuracy, resulting in a **catastrophic forgetting problem** in known categories. To illustrate this issue, we have tracked the classification accuracy of known and novel categories on unlabeled data during each training epoch on CUB [37]. As shown in Fig. 1a, the green curve represents the accuracy of known (Old) classes, while the orange curve represents the accuracy of novel (New) classes. Notably, we can easily observe this phenomenon, with the accuracy of novel classes improving, the accuracy of known categories initially increases to approximately 74% after 20 epochs but then drops to 64.44% in the end. We thus conclude that SimGCD faces catastrophic forgetting in known categories during training.

To address the above issue, we propose a novel *Local Entropy Regularization* (LER) to preserve the knowledge of known categories. In particular, we first identify potential known samples using a threshold on their logits prediction like FixMatch [29]. Then, we employ the information entropy function to encourage the predictions of above selected known samples close to a uniform distribution, thereby maintaining the prediction stability of known classes. Consequently, this LER prevents known samples from being misclassified as novel classes and therefore preserves the knowledge related to known categories during

learning novel categories.

It's worth noting that the model may occasionally miss potential known samples or select incorrect (novel) samples for LER. For example, when we have two augmentation view samples, x_i and x_i', from the same image, and x_i has higher logits than the threshold while x_i' falls below it. In such cases, we can't be certain whether the original image belongs to known classes, and this uncertainty may impact the effectiveness of LER. We argue that the predictions of the two view samples should be correctly aligned to ensure the quality of the chosen known samples. Therefore, we further propose a dual-view alignment scheme called *Dual-views Kullback–Leibler divergence constraint* (DKL), which employs Kullback–Leibler (KL) divergence to encourage the consistency of two views from the same image.

To summarize, we propose a novel approach named LegoGCD, which integrates SimGCD [39] with our proposed LER and DKL to address the problem of catastrophic forgetting. To validate the effectiveness of LegoGCD, we conduct extensive experiments on eight datasets, including generic datasets such as CIFAR10/100 [15], ImageNet-1k [5], and fine-grained datasets CUB [37], Stanford Cars [14], and FGVC-Aircraft [21]. Intuitively, we also visualize the classification accuracy of known and novel categories on CUB [37]. These results are shown in Fig. 1b. The green curve represents the accuracy of known (Old) categories, while the orange curve indicates the accuracy of novel (New) classes. Clearly, LegoGCD effectively prevents the decline in known classes and achieves an accuracy of 72.18%, surpassing SimGCD by a margin of **7.74**. Clearly, the results indicate that LegoGCD solves the catastrophic forgetting problem of known categories effectively. Moreover, our method can be easily placed onto SimGCD like Lego, requiring only a few lines of code on the implementation without introducing any additional parameters or altering the internal network structure of SimGCD.

In summary, our key contributions are as follows:

- We introduce a novel constraint named Local Entropy Regularization (LER), which is designed to mitigate the catastrophic forgetting problem of known classes by preserving the knowledge of known categories during learning novel classes.
- We propose a Dual-views Kullback–Leibler divergence constraint (DKL) that ensures the prediction distribution of one view approximates that of another, maintaining consistency between dual views augmented from the same image.
- The proposed LegoGCD is effective and can be simply integrated from SimGCD without any extra parameter addition. Extensive results demonstrate our method exhibits significant performance improvement on known classes, *e.g.*, a 7.74% increase in CUB [37].

2. Related Work

2.1. Generalized Category Discovery

GCD was first formulated by Vaze *et al*. [36], presents a unique challenge distinct from Semi-supervised Learning (SSL) [3, 17, 22, 32, 47]. While SSL assumes that unlabeled data belongs to the same class set as the labeled data, GCD tackles a more realistic scenario where the unlabeled data may include classes not present in the labeled set, which is the same setting in Novel Category Discovery (NCD) [7–9, 13, 18, 19, 40, 41, 49]. Therefore, GCD can be viewed as an extension of NCD, with the main difference being that GCD seeks to identify specific categories within novel classes, while NCD focuses on grouping novel classes into a single category. The original GCD approach employs contrastive and SSL, which uses clustering during inference and incurs significant computational costs. To address this challenge, Wen *et al*. [39] introduce SimGCD with a classifier to replace clustering, offering a robust baseline for the GCD problem. However, it's important to note that SimGCD introduces a drawback, leading to a decrease in the classification accuracy of known classes during the intense learning of novel categories.

2.2. Entropy regularization

It is a widely used technique in image classification, especially in the context of cross-entropy, which aims to align prediction distributions with the standard label distribution. However, in scenarios such as Semi-supervised Learning (SSL) [3, 17, 22, 32, 47], where true labels are unknown, pseudo labels take the place of actual labels in standard cross-entropy. This form of entropy regularization minimizes output differences between various views of unlabeled data. Notably, Data Augmentations [43, 44] have proven effective, contributing to substantial successes in pseudo-supervised learning. For instance, in SimGCD [39], an augmentation strategy generates two views of data, establishing training targets in one view and enforcing prediction consistency with the other view during unlabeled data training. Another form of entropy, information entropy, measures the amount of information within a set of events. In SimGCD, information entropy is employed to minimize class mean entropy, promoting more uniform class predictions in each iteration to ensure the visibility of novel classes. However, due to the absence of protection for the knowledge of known classes, class mean entropy has led to a degeneration in known categories.

3. Method

In this section, we first formulate the GCD task (Sec. 3.1) and present the overview of the baseline SimGCD [39] (Sec. 3.2). Then, we introduce how to mitigate the degradation of SimGCD by the proposed LegoGCD. At last, we

describe the details of the proposed Local Entropy Regularization (LER) and Dual-views Kullback-Leibler divergence constraint (DKL) in Sec. 3.3 and Sec. 3.4, respectively.

3.1. Problem Formulation

Traditional image classification tasks are typically developed using a labeled dataset, denoted as $\mathcal{D}^l = \{(\boldsymbol{x}_i, y_i)\} \in \mathcal{X} \times \mathcal{Y}_l$. This dataset contains only samples from known classes, represented by $\mathcal{Y}_l$. In contrast, Generalized Category Discovery (GCD) aims to recognize unlabeled data, denoted as $\mathcal{D}^u = \{(\boldsymbol{x}_i, y_i)\} \in \mathcal{X} \times \mathcal{Y}_u$. This dataset comprises both known and novel class samples, where $\mathcal{Y}_l$ is a subset of $\mathcal{Y}_u$. The goal of GCD is to develop a model that can identify both known and novel classes using the labels from known categories ($\mathcal{Y}_l$) and unlabeled data ($\mathcal{D}_u$) without access to class labels. It's important to note that the total number of categories is represented as $K = |\mathcal{Y}_l \cup \mathcal{Y}_u|$. We assume prior knowledge of this total category count, as done in previous works [7, 9, 48, 50].

3.2. Preliminaries

In this section, we introduce the fundamental structure of SimGCD [39] baseline program as shown in Fig. 2. This program primarily consists of two key components: Representation learning and Parametric Classification.

3.2.1 Representation Learning

The representation learning process in our framework follows GCD [36] and SimGCD [39]. It employs a Vision Transformer (ViT-B/16) [6] pretrained with DINO self-supervision [4] on ImageNet [5] as the backbone. This process includes supervised contrastive learning on labeled data and unsupervised contrastive learning on *all* data, encompassing both labeled and unlabeled data.

Formally, given two views (random augmentations) $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ of the same image in a mini-batch B, the unsupervised contrastive loss is written as:

$$\mathcal{L}_{\text{rep}}^u = \frac{1}{|B|} \sum_{i \in B} -\log \frac{\exp\left(\boldsymbol{z}_i^\top \cdot \boldsymbol{z}_i' / \tau_u\right)}{\sum_i^{i \neq n} \exp\left(\boldsymbol{z}_i^\top \cdot \boldsymbol{z}_n' / \tau_u\right)}, \quad (1)$$

where $\boldsymbol{z}_i = \phi\left(f\left(\boldsymbol{x}_i\right)\right)$ and $\boldsymbol{z}_i' = \phi\left(f\left(\boldsymbol{x}_i'\right)\right)$, f is the feature backbone, ϕ is a multi-layer perceptron (MLP) projection head, τ_u is a unsupervised temperature.

The objective of the supervised contrastive loss is to encourage the model to bring samples with the same class label closer in the feature space, formally written as:

$$\mathcal{L}_{\text{rep}}^s = \frac{1}{|B^l|} \sum_{i \in B^l} \frac{1}{|\mathcal{N}_i|} \sum_{q \in N_i} -\log \frac{\exp\left(\boldsymbol{z}_i^\top \cdot \boldsymbol{z}_q' / \tau_c\right)}{\sum_i^{i \neq n} \exp\left(\boldsymbol{z}_i^\top \cdot \boldsymbol{z}_n' / \tau_c\right)}, \quad (2)$$

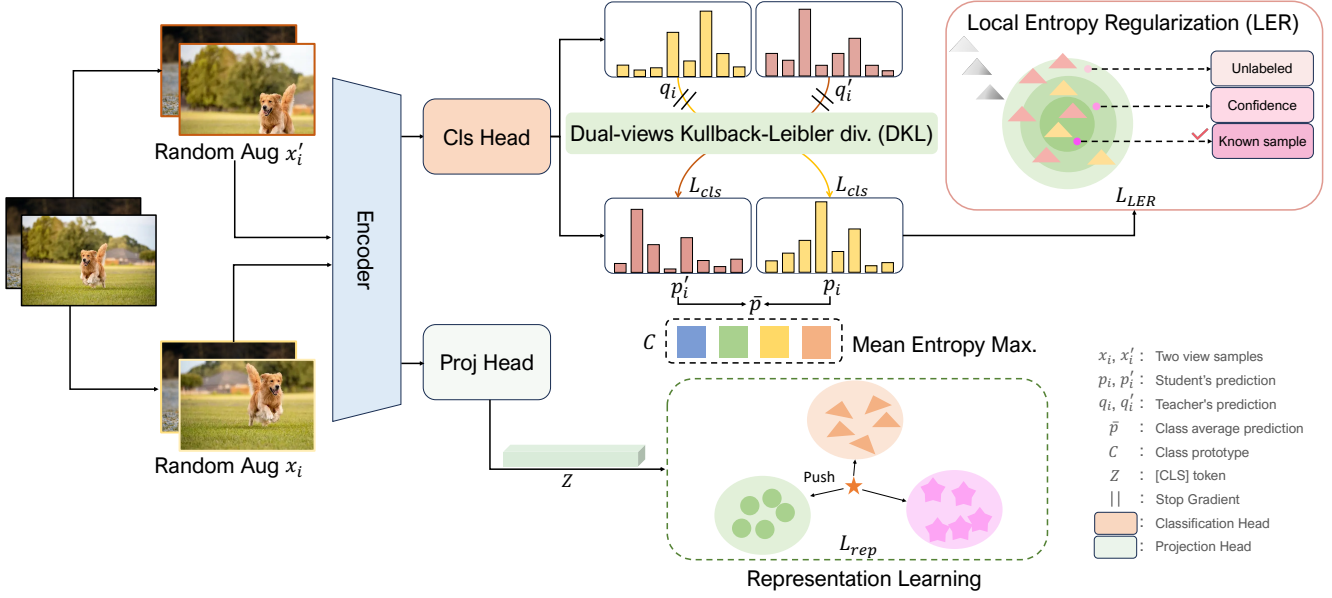where $\mathcal{N}_i$ is the indices of images share the same label with

Figure 2. Illustration of our proposed LegoGCD. LegoGCD is mainly composed of SimGCD and our proposed LER and DKL. (a) Representation learning and Mean Entropy in SimGCD (Sec. 3.2). (b) Local Entropy Regularization (LER) (Sec. 3.3) for discovering potential known samples in unlabeled data and preserving the knowledge of known classes. (c) Dual-views Kullback-Leibler divergence (DKL) (Sec. 3.4) to ensure consistent predictions for two view samples.

x_i in the mini-batch B. Finally, the total loss in representation learning is constructed as:

$$\mathcal{L}_{\text{rep}} = (1 - \lambda)\mathcal{L}_{\text{rep}}^u + \lambda\mathcal{L}_{\text{rep}}^s, \quad (3)$$

where B^l is the labeled subset of B and λ is a weight factor.

3.2.2 Parametric Classification

Different from the GCD [36] that uses k-means, SimGCD [39] employs a more efficient classifier based on self-distillation [2, 4]. Formally, the classifier is denoted as a set of prototypes $\mathcal{C} = \{c_1, \ldots, c_K\}$, where $K = |\mathcal{Y}_l \cup \mathcal{Y}_u|$. During training, the soft label of each view x_i is obtained by applying softmax on cosine similarity between hidden feature $h_i = f(x_i)$ and prototypes $\mathcal{C}$, scaled by $1/\tau_s$:

$$p_i^{(k)} = \frac{\exp\left(\frac{1}{\tau_s}(h_i/\|h_i\|_2)^\top (c_k/\|c_k\|_2)\right)}{\sum_{k'}\exp\left(\frac{1}{\tau_s}(h_i/\|h_i\|_2)^\top (c_{k'}/\|c_{k'}\|_2)\right)}, \quad (4)$$

The soft pseudo label q_i' for view x_i' is similarly generated. Then, it employs a cross-entropy loss $\ell(q', p) = -\sum_k q'^{(k)} \log p^{(k)}$ to supervise the learning of prediction with pseudo labels or ground-truth labels:

$$\mathcal{L}_{\text{cls}}^u = \frac{1}{|B|}\sum_{i\in B}\ell(q_i', p_i), \mathcal{L}_{\text{cls}}^s = \frac{1}{|B|}\sum_{i\in B}\ell(y_i, p_i), \quad (5)$$

where $\mathcal{L}_{\text{cls}}^u$ and $\mathcal{L}_{\text{cls}}^s$ are unsupervised and supervised classification losses for all and labeled data, respectively. To regulate unsupervised learning, SimGCD adopts a class mean entropy regulariser [2]: $H(\overline{p}) = -\sum_k \overline{p}^{(k)} \log \overline{p}^{(k)}$, where $\overline{p}$ is the mean prediction of each class in a batch $\overline{p} = \frac{1}{2|B|}\sum_{i\in B}(p_i + p_i')$. Then the classification objective is: $\mathcal{L}_{\text{cls}} = (1 - \lambda)(\mathcal{L}_{\text{cls}}^u - \varepsilon H(\overline{p})) + \lambda\mathcal{L}_{\text{cls}}^s$. Finally, the overall objective in baseline SimGCD is $\mathcal{L}_{\text{rep}} + \mathcal{L}_{\text{cls}}$.

3.3. Local Entropy Regularization

Motivation. Although the baseline SimGCD [39] has improved accuracy and computational efficiency compared to GCD [36], it faces catastrophic forgetting in known (Old) classes during training, as shown in Fig. 1a. This is due to SimGCD relying on class mean entropy, causing a shift in focus to novel classes and resulting in information loss in known classes. We argue that retaining knowledge of known classes should be prioritized. Fig. 3a displays potential known class samples in each epoch on the FGVC-Aircraft [21] dataset. Evidently, LegoGCD recognizes nearly 10 more potential known samples than SimGCD in the end. Fig. 3b illustrates the maximum number of potential known samples in various datasets, confirming that LegoGCD with LER excels at preserving the recognition ability of known categories. Concretely, we propose a Local Entropy Regularization (LER) to preserve the knowledge of known categories, solving the problem of catastrophic forgetting. In contrast to the class mean entropy, which primarily shifts the network's focus to novel classes, we argue that the network should also maintain its ability to recognize
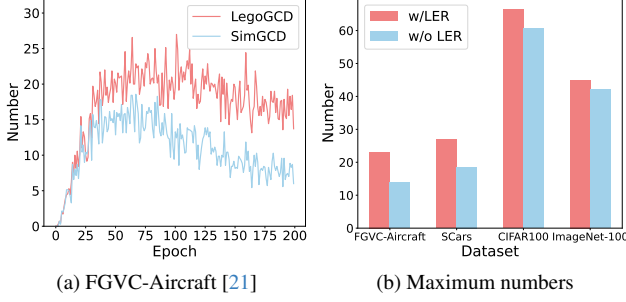
(a) FGVC-Aircraft [21]   (b) Maximum numbers

Figure 3. Comparison of potential known samples in SimGCD [39] and LegoGCD. (a) LegoGCD recognizes almost 10 more high-confidence samples than SimGCD in the end on the FGVC-Aircraft dataset. (b) LegoGCD with LER produces more high-confidence known samples in various generic and fine-grained datasets compared to SimGCD without LER.

known samples as it did before. Specifically, we choose known samples based on a confidence threshold δ in unlabeled data and utilize entropy regularization to ensure the stability of these known classes associated with the selected known samples.

The training dataset, denoted as $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\} \in \mathcal{X} \times \mathcal{Y}$, comprises both labeled and unlabeled samples represented as $\boldsymbol{x}_i$ within a batch B. To distinguish between labeled and unlabeled samples in a batch, we utilize a binary mask vector $M = [m_1, m_2, \ldots, m_i] \subseteq \{0, 1\}$, where each m_i can be either 0 (indicating an unlabeled sample) or 1 (indicating a labeled sample). Consequently, the unlabeled samples in a batch are obtained by applying this mask $M = 0$.

Next, let $\boldsymbol{p}_i = \left[p_i^1, p_i^2, \ldots, p_i^k\right]$ as the prediction vector for sample $\boldsymbol{x}_i$, where K represents the total number of categories. We use $S = [s_1, s_2, \ldots, s_i] \subseteq \{0, 1\}$ as a binary vector to denote the **high-confidence** sample. When $s_i = 1$, it indicates that $\boldsymbol{x}_i$ is a potential high-confidence sample. This can be expressed as follows:

$$s_i = \mathbb{1}\left(\max\left(\boldsymbol{p}_i\right) \geq \delta\right), \quad (6)$$

where δ represents the confidence threshold. Then, we let $\mathcal{Y} = [y_1, y_2, \ldots, y_i] \in \{1, 2, \ldots, K\}$ denotes the potential class label corresponding to $\boldsymbol{x}_i$. y_i is determined by the index of the maximum value in $\boldsymbol{p}_i$:

$$y_i = \arg\max_j \left(\boldsymbol{p}_i\right)_j, \quad i = 1, 2, \ldots, b \quad (7)$$

where b denotes the number of batch sizes. We also introduce $\mathcal{O} = [o_1, o_2, \ldots, o_i] \subseteq \{0, 1\}$ as a binary vector, and $o_i = 1$ signifying potential **known** samples with **high-confidence** in **unlabeled data**. This calculation can be performed as follows:

$$o_i = \underbrace{\mathbb{1}\left(m_i = 0\right)}_{\text{Unlabeled}} \cdot \underbrace{\mathbb{1}\left(s_i = 1\right)}_{\text{High-confidence}} \cdot \underbrace{\mathbb{1}\left(y_i \in \mathcal{Y}_l\right)}_{\text{Known}}, \quad (8)$$

where $\mathcal{Y}_l$ represents the known class set. Next, we use information entropy to assess the stability of the known categories during training, which is expressed as follows:

$$\mathcal{L}_{entropy} = -\frac{1}{B} \sum_{i=1}^{B} \mathbb{1}\left(o_i = 1\right) \cdot H\left(\frac{1}{\tau_o} \boldsymbol{p}(\boldsymbol{x}_i)\right), \quad (9)$$

where τ_o is a temperature, H is an entropy resulariser [2] used in Sec. 3.2.2. Additionally, to further enhance the margins between the categories, we replace the vanilla information loss with a Margin-aware Pattern (MAP) [38, 42], and the final LER loss can be formulated as:

$$\mathcal{L}_{LER} = -\frac{1}{B} \sum_{i=1}^{B} \mathbb{1}\left(o_i = 1\right) \cdot H\left(\frac{1}{\tau_o} \boldsymbol{p}(\boldsymbol{x}_i), \frac{1}{\tau_o} \boldsymbol{p}(\boldsymbol{x}_i) - \Delta_j\right), \quad (10)$$

where $\Delta_j = \lambda_{ler} \log\left(\frac{1}{\tilde{p}_j}\right)$, $j \in \{1, \ldots, K\}$, $\lambda_{ler} = 0.4$, and $\tilde{p}_j$ is the average model prediction updated at each iteration through an exponential moving average.

The total process is divided into three steps: 1) Applying the label mask M to select samples from **unlabeled data**; 2) Using the threshold δ to select **high-confidence** samples from all training data; 3) Identifying high-confidence **known** samples from all training samples. Furthermore, we provide an intuitive representation of the entire process in the top-right corner of Fig. 2.

Fig. 3b compares the quantity of high-confidence known samples across various datasets. Clearly, the numbers increase with the introduction of LER, proving our method with LER retains more knowledge about known classes.

3.4. Dual-views Kullback-Leibler divergence

It's crucial to note that the model might mistakenly identify incorrect samples as known ones when considering two views of the same image for LER. For instance, if $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ are misaligned, and $\boldsymbol{x}_i$ exceeds the confidence threshold δ while $\boldsymbol{x}_i'$ falls below it, uncertainty arises about whether the image of the two views is a potential known one. Thus, we might erroneously select $\boldsymbol{x}_i$ or miss $\boldsymbol{x}_i'$ for LER. In other words, we can confidently select both $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ as known samples only when both $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ exceed the threshold δ.

In general, an ideal approach is to push the alignment of two view samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ and both belong to the known or novel sample set. To achieve this, we propose a dual-view alignment technique named Dual-views Kullback-Leibler divergence constraint (DKL).

Formally, given wo cosine similarity $\boldsymbol{p}_i$ and $\boldsymbol{p}_i'$ from Sec. 3.2.2 of two view samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ in a mini-batch B, the DKL can be formulated as:

$$D_{KL}(\boldsymbol{p}_i \| \boldsymbol{p}_i') = \frac{1}{B/2} \sum_{i=1}^{B/2} \boldsymbol{p}(\boldsymbol{x}_i) \cdot \log \frac{\boldsymbol{p}(\boldsymbol{x}_i)}{\boldsymbol{p}(\boldsymbol{x}_i')}. \quad (11)$$

In summary, DKL aligns the predictions of two view samples, enhancing the creation of more reliable potential known samples for LER. Finally, the ultimate classification loss can be updated as:

$$\mathcal{L}_{\text{cls}} = (1-\lambda)(\mathcal{L}_{\text{cls}}^u - \varepsilon H(\overline{\boldsymbol{p}}) + D_{KL}(\boldsymbol{p}_i \| \boldsymbol{p}_i')) + \lambda \mathcal{L}_{\text{cls}}^s. \quad (12)$$

where λ is a weight factor to control the balance between supervised and unsupervised classification learning.

By simply integrating the Local Entropy Regularization (LER) and Dual-views Kullback-Leibler divergence constraint (DKL) into SimGCD, we propose a new paragram named LegoGCD. The overall loss for training our model can be formulated as:

$$\mathcal{L} = \alpha \cdot (\mathcal{L}_{rep} + \mathcal{L}_{cls}) + \beta \cdot \mathcal{L}_{LER}, \quad (13)$$

where β is a control factor to assign the weight to remember known classes. Aligning with SimGCD, we set α to 1 and simultaneously adjust β (see Tab. 7). Notably, the DKL is plugged into classification loss $\mathcal{L}_{cls}$. Algorithm 1 in appendix describes one training step of LegoGCD.

4. Experiment

4.1. Experimental Setup

Dataset. We evaluate the effectiveness of our approach on eight datasets, consistent with SimGCD [39]. These datasets encompass generic image recognition datasets like CIFAR10/100 [15] and ImageNet-100 [33], as well as Semantic Shit [35] datasets, including CUB [37], Stanford Cars [14], and FGVC-Aircraft [21]. Additionally, we include two more challenging datasets: Herbarium 19 [31] and ImageNet-1k [26]. For each dataset, we follow the GCD [36] and SimGCD [39] protocols by sub-sampling 50% of known class images to form the labeled set $\mathcal{D}^l$ within the training set. The remaining images from known and novel classes constitute the unlabeled data $\mathcal{D}^u$. Tab. 1 provides details of the datasets used in our experiments.

Evaluation protocol. During training, we use dataset $\mathcal{D}$ combined by $\mathcal{D}^l$ and $\mathcal{D}^u$ to train the models. For evaluation, we use clustering accuracy (ACC) [36, 39] to evaluate the model performance. Specifically, ACC is computed as follows: given the ground-truth label y^* and the model's prediction $\hat{y}_i$, $ACC = \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}(y_i^* = p(\hat{y}_i))$, where $M = |\mathcal{D}^u|$, and p is determined using the Hungarian optimal assignment algorithm [16].

Implementation details. Following [36, 39], we conduct our experiments using ViT-B/16 backbone [6], which was pre-trained with DINO [4], and only fine-tune the last attention block of the backbone for all models. We use the [CLS] token output as the image feature and input for classifier training in SimGCD. Our training regimen includes an initial learning rate of 0.1, decay using a cosine schedule. For a fair comparison, we use a batch size of 128 and

Table 1. Overview of the datasets used in our experiments. We list the specific number of labeled and unlabeled images ($\mathcal{D}^l$, $\mathcal{D}^u$) and their corresponding class assignments (Old and New).

Dataset	Balance	Labeled $\mathcal{D}^l$		Unlabeled $\mathcal{D}^u$	
		Images	Old	Images	New
CIFAR10 [15]	✓	12.5k	5	37.5k	10
CIFAR100 [15]	✓	20.0k	80	30.0k	100
ImageNet-100 [33]	✓	31.9k	50	95.3k	100
CUB [37]	✓	1.5k	100	4.5k	200
Stanford Cars [14]	✓	2.0k	98	6.1k	196
FGVC-Aircraft [21]	✓	1.7k	50	5.0k	100
Herbarium 19 [31]	✗	8.9k	341	25.4k	683
ImageNet-1K [26]	✓	321k	500	960k	1000

train the models for 200 epochs, setting $\lambda = 0.35$ in the loss function (see Eq. (5)). Temperature values $\tau_u = 0.07$ and $\tau_c = 1.0$ are employed in representation learning. As for training the classifier in SimGCD, we follow the same settings, including $\tau_s = 0.1$ and initial $\tau_t = 0.07$ which is warmed up to 0.04 with a cosine schedule within the first 30 epochs. In LegoGCD, we set $\tau_o = 0.05$ (see Eq. (9)). All experiments are conducted using PyTorch and trained on Nvidia Tesla V100 GPUs.

4.2. Comparison with the baselines

We compare our approach with Generalized Category Discovery methods like k-means [20], ORCA[19], GCD [36], SimGCD [39], and strong baselines derived from Novel Category Discovery, including RS+ [9] and UNO+[7]. For a fair comparison, we reproduce SimGCD (denoted as SimGCD*) using the same random seed (*i.e.* seed=0) as our method. Tab. 2 shows results on SSB datasets [35] and Herbarium 19 [31], Tab. 3 and Tab. 4 present results on generic recognition datasets, including the challenging ImageNet-1k [26].

Overall, our method effectively mitigates catastrophic forgetting and achieves superior performance compared to the GCD and SimGCD baseline, particularly in recognizing "Old" categories. Specifically, it outperforms the baseline by **3.5%/2.1%** on CIFAR100 and ImageNet-100 and shows significant improvements in fine-grained evaluations with **5.4%** in CUB and **5.1%** in Stanford Cars. Additionally, it surpasses the baseline by **0.4%/0.5%** on the challenging datasets ImageNet-1k and Herbarium 19. In addressing the forgetting problem, our approach also competes well with SimGCD on "New" classes, achieving **3.4%** in Stanford Cars, **1.0%** in CIFAR100, and **3.5%** in ImageNet-100.

4.3. Ablation Study

In this section, we conduct ablation studies to validate the effectiveness of LER and DKL in LegoGCD. The datasets considered include fine-grained datasets like CUB and Stanford Cars, as well as generic image recognition datasets CI-

Table 2. Classification results on Semantic Shift Benchmark [35] datasets and Herbarium 19 [31]. **Bold** represent our results, Δ indicates the margin ahead of the baseline SimGCD, and red signifies improvement in known categories.

|	CUB			Stanford Cars			FGVC-Aircraft			Herbarium 19		
Method	All	Old	New	All	Old	New	All	Old	New	All	Old	New
k-means[20]	34.3	38.9	32.1	12.8	10.6	13.8	16.0	14.4	16.8	13.0	12.2	13.4
RS+[9]	33.3	51.6	24.2	28.3	61.8	12.1	26.9	36.4	22.2	27.9	55.8	12.8
UNO+[7]	35.1	49.0	28.1	35.5	70.5	18.6	40.3	56.4	32.2	28.3	53.7	14.7
ORCA[19]	35.3	45.6	30.2	23.5	50.1	10.7	22.0	31.8	17.1	20.9	30.9	15.5
GCD [36]	51.3	56.6	48.7	39.0	57.6	29.9	45.0	41.1	46.9	35.4	51.0	27.0
SimGCD [39]	60.3	65.6	57.7	53.8	71.9	45.0	54.2	59.1	51.8	44.0	58.0	36.4
SimGCD* [39]	61.9	66.5	59.6	53.4	70.6	45.0	54.6	61.4	51.1	44.9	56.9	38.4
Ours	**63.8**	**71.9**	**59.8**	**57.3**	**75.7**	**48.4**	**55.0**	**61.5**	**51.7**	**45.1**	**57.4**	**38.4**
Δ	**1.9**	**5.4**	**0.2**	**3.9**	**5.1**	**3.4**	**0.4**	**0.1**	**0.6**	**0.2**	**0.5**	**0.0**
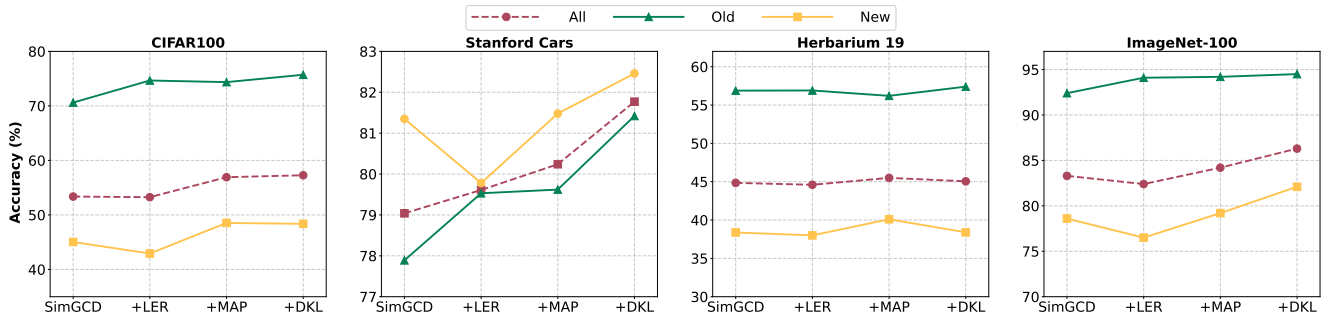
Figure 4. Step by step, we integrate LER and DKL into the baseline SimGCD [39]. Initially, the addition of LER increases accuracy in the "Old" category while decreasing accuracy in the "New" category. Subsequently, the introduction of a Margin-aware Pattern (MAP) widens margins between novel categories, ultimately achieving the best performance when embedding with DKL.

Table 3. Classification results on the generic image recognition datasets, CIFAR10 [15] and CIFAR100 [15].

	CIFAR10			CIFAR100		
Method	All	Old	New	All	Old	New
k-means[20]	83.6	85.7	82.5	52.0	52.2	50.8
RS+[9]	46.8	19.2	60.5	58.2	77.6	19.3
UNO+[7]	68.6	98.3	53.8	69.5	80.6	47.2
ORCA[19]	81.8	86.2	79.6	69.0	77.4	52.0
GCD [36]	91.5	97.9	88.2	73.0	76.2	66.5
SimGCD [39]	97.1	95.1	98.1	80.1	81.2	77.8
SimGCD* [39]	96.9	93.8	98.5	79.0	77.9	81.5
Ours	**97.1**	**94.3**	**98.5**	**81.8**	**81.4**	**82.5**
Δ	**0.2**	**0.5**	**0.0**	**2.8**	**3.5**	**1.0**

Table 4. Classification results on the generic image recognition ImageNet-100 [33] and the challenging ImageNet-1k [26].

	ImageNet-100			ImageNet-1k		
Method	All	Old	New	All	Old	New
k-means[20]	72.7	75.5	71.3	-	-	-
RS+[9]	37.1	61.6	24.8	-	-	-
UNO+[7]	70.3	95.0	57.9	-	-	-
ORCA[19]	73.5	92.6	63.9	-	-	-
GCD [36]	74.1	89.8	66.3	52.5	72.5	42.2
SimGCD [39]	83.0	93.1	77.9	57.1	77.3	46.9
SimGCD* [39]	83.3	92.4	78.6	62.3	79.1	53.8
Ours	**86.3**	**94.5**	**82.1**	**62.4**	**79.5**	**53.8**
Δ	**3.0**	**2.1**	**3.5**	**0.1**	**0.4**	**0.0**

FAR100 and ImageNet-100.

Local Entropy Regularization (LER). We conduct ablation studies on LER as shown in Fig. 4. Firstly, the SimGCD baseline with LER (without Margin-aware Pattern, MAP) notably enhances "Old" Categories (see green curves). However, using raw LER alone may impact accuracy in "New" classes (see orange curves). This is be-

cause it primarily encourages the network to remember known classes, consequently influencing the learning of novel ones. To mitigate this, we incorporate MAP (see Eq. (10)) to encourage the network to simultaneously enhance the margins of all classes, particularly novel classes. Finally, the inclusion of MAP in LER leads to improvements in both the "Old" and "New" categories.

Table 5. Ablation study on different combinations of our algorithms on CUB. **Bold** indicates the best results.

SimGCD	LER	MAP	DKL	CUB All	Old ↑	New
✓				61.9	66.5	59.6
✓			✓	62.5	67.3	**60.2**
✓	✓			63.4	71.0	59.6
✓	✓	✓		63.7	**72.0**	58.5
✓	✓	✓	✓	**63.8**	71.9	59.8

Table 6. Ablation study on α and β was conducted on CUB and CIFAR100. **Bold** indicates the best results. The <u>underline</u> denotes the selected β.

α	β	CUB All	Old ↑	New	CIFAR100 All	Old ↑	New
	0.0	61.9	66.5	59.6	79.0	77.9	81.5
	0.5	63.0	69.5	59.7	80.8	80.4	81.5
	<u>1.0</u>	63.6	69.7	**60.6**	**81.8**	81.4	**82.5**
1.0	1.5	63.4	71.2	59.8	81.4	81.7	80.8
	<u>2.0</u>	**63.8**	**71.9**	59.8	81.7	**82.1**	80.9

Table 7. Ablation study on confidence threshold δ was conducted on Stanford Cars and CIFAR100. **Bold** indicates the best results. The <u>underline</u> denotes the selected threshold.

δ	Stanford Cars All	Old ↑	New	CIFAR100 All	Old ↓	New
0.70	55.3	66.8	**49.8**	80.3	**83.3**	74.3
0.75	55.8	72.3	47.9	80.9	83.3	76.3
0.80	56.5	73.0	48.5	81.3	82.7	78.5
<u>0.85</u>	**57.3**	**75.7**	48.4	**81.8**	81.4	**82.5**
0.90	56.4	75.3	47.3	79.0	80.5	76.0

5. Conclusion

In this paper, we propose LegoGCD, a novel approach to mitigate the issue of catastrophic forgetting in known categories. The core of our design is to preserve knowledge of known classes while maintaining the accuracy of novel classes. To achieve this, we develop two techniques: Local Entropy Regularization (LER) and Dual-views Kullback-Leibler divergence constraint (DKL). LER explicitly regularizes high-confidence potential known class samples to retain the knowledge of known categories. To ensure the accurate selection of these samples, we employ DKL to align the distribution of two view samples for LER. Both LER and DKL can be seamlessly integrated into baseline SimGCD resembling Lego blocks, without introducing new parameters or altering the internal network structure. Extensive experiments demonstrate that LegoGCD significantly enhances performance in known classes, effectively addressing the catastrophic forgetting problem.

6. Acknowledgements

Dual-views Kullback-Leibler divergence (DKL). DKL is designed to improve the quality of known samples for LER by aligning the predictions of two views from the same image. The results in Tab. 5 indicate that incorporating DKL into raw SimGCD led to improvements in both "Old" and "New" category accuracies by 1.8% and 0.6%, respectively (67.3% vs. 66.5%, 60.2% vs. 59.6%). This success proves beneficial for self-distillation learning in SimGCD. Additionally, from Fig. 4 and Tab. 5, we can see that our method becomes more effective with DKL, particularly in "Old" classes, while maintaining performance on "New" classes.

Different α and β. The coefficient β is crucial for balancing knowledge preservation of known classes and facilitating effective learning of novel classes. As depicted in Tab. 6 with $\alpha = 1.0$ aligned to SimGCD, accuracy on "Old" categories consistently surpasses SimGCD ($\beta = 0.0$) across different β values. Optimal equilibrium is achieved with $\beta = 2.0$ in CUB and $\beta = 1.0$ in CIFAR100.

Different confidence threshold δ. We compare accuracy with different δ in Tab. 7. Results show varying responses across datasets. In Stanford Cars, "Old" accuracy initially increases, then slightly decreases, while "New" accuracy decreases but remains above SimGCD at 45.5%. In contrast, in CIFAR100, "Old" accuracy decreases but consistently surpasses SimGCD at 77.9%. Our goal is to prioritize high "Old" accuracy and ensure "New" equals or exceeds SimGCD. Therefore, we choose $\delta = 0.85$ for both Stanford Cars and CIFAR100.

References

[1] Wenbin An, Feng Tian, Qinghua Zheng, Wei Ding, QianYing Wang, and Ping Chen. Generalized category discovery with decoupled prototypical network. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 12527–12535, 2023. 1

[2] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. Masked siamese networks for label-efficient learning. In *Proceedings of the European*

conference on Computer Vision (ECCV), pages 456–473, 2022. 4, 5

[3] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5050–5060, 2019. 2, 3

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 9630–9640, 2021. 1, 3, 4, 6

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 2, 3

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021. 3, 6

[7] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9264–9272, 2021. 1, 3, 6, 7

[8] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8400–8408, 2019. 1

[9] Kai Han, Sylvestre-Alvise Rebuffi, Sébastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Autonovel: Automatically discovering and learning novel visual categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(10):6767–6781, 2022. 1, 3, 6, 7

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2980–2988, 2017. 1

[12] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. 1

[13] K. J. Joseph, Sujoy Paul, Gaurav Aggarwal, Soma Biswas, Piyush Rai, Kai Han, and Vineeth N. Balasubramanian. Novel class discovery without forgetting. In *Proceedings of the European conference on Computer Vision (ECCV)*, pages 570–586, 2022. 3

[14] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization.

In *Proceedings of IEEE/CVF International Conference on Computer Vision Workshops (ICCV)*, pages 554–561, 2013. 2, 6

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 6, 7

[16] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 6

[17] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *Proceedings of International Conference on Learning Representations (ICLR)*. OpenReview.net, 2017. 3

[18] Wenbin Li, Zhichen Fan, Jing Huo, and Yang Gao. Modeling inter-class and intra-class constraints in novel class discovery. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3449–3458, 2023. 3

[19] Jiaming Liu, Yangqiming Wang, Tongze Zhang, Yulu Fan, Qinli Yang, and Junming Shao. Open-world semi-supervised novel class discovery. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI*, pages 4002–4010. ijcai.org, 2023. 3, 6, 7

[20] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pages 281–297, 1967. 6, 7

[21] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 2, 4, 5, 6

[22] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(8):1979–1993, 2019. 3

[23] Nan Pu, Zhun Zhong, and Nicu Sebe. Dynamic conceptional contrastive learning for generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7579–7588, 2023. 1

[24] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 91–99, 2015. 1

[25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015. 1

[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015. 6, 7

[27] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018. 1

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[29] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015. 1

[31] Kiat Chuan Tan, Yulong Liu, Barbara Ambrose, Melissa Tulig, and Serge Belongie. The herbarium challenge 2019 dataset. *arXiv preprint arXiv:1906.05372*, 2019. 6, 7

[32] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1195–1204, 2017. 3

[33] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Proceedings of the European conference on Computer Vision (ECCV)*, pages 776–794, 2020. 6, 7

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017. 1

[35] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2022. 6, 7

[36] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491, 2022. 1, 2, 3, 4, 6, 7

[37] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 1, 2, 6

[38] Xudong Wang, Zhirong Wu, Long Lian, and Stella X. Yu. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14627–14637, 2022. 5

[39] Xin Wen, Bingchen Zhao, and Xiaojuan Qi. Parametric classification for generalized category discovery: A baseline study. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16590–16600, 2023. 1, 2, 3, 4, 5, 6, 7

[40] Muli Yang, Yuehua Zhu, Jiaping Yu, Aming Wu, and Cheng Deng. Divide and conquer: Compositional experts for generalized novel class discovery. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14248–14257, 2022. 3

[41] Muli Yang, Liancheng Wang, Cheng Deng, and Hanwang Zhang. Bootstrap your own prior: Towards distribution-agnostic novel class discovery. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3459–3468, 2023. 3

[42] Zhuoran Yu, Yin Li, and Yong Jae Lee. Inpl: Pseudo-labeling the inliers first for imbalanced semi-supervised learning. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2023. 5

[43] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6022–6031, 2019. 3

[44] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2018. 3

[45] Jie Zhang, Xiaosong Ma, Song Guo, and Wenchao Xu. Towards unbiased training in federated open-world semi-supervised learning. In *Proceedings of International Conference on Machine Learning, (ICML)*, pages 41498–41509, 2023. 1, 2

[46] Sheng Zhang, Salman H. Khan, Zhiqiang Shen, Muzammal Naseer, Guangyi Chen, and Fahad Shahbaz Khan. 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 (CVPR)*, pages 3479–3488, 2023. 1

[47] Wenqiao Zhang, Lei Zhu, James Hallinan, Shengyu Zhang, Andrew Makmur, Qingpeng Cai, and Beng Chin Ooi. Boostmis: Boosting medical image semi-supervised learning with adaptive pseudo labeling and informative active annotation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20634–20644, 2022. 3

[48] Bingchen Zhao and Kai Han. Novel visual category discovery with dual ranking statistics and mutual knowledge distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 22982–22994, 2021. 3

[49] Yuyang Zhao, Zhun Zhong, Nicu Sebe, and Gim Hee Lee. Novel class discovery in semantic segmentation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4330–4339, 2022. 3

[50] Zhun Zhong, Linchao Zhu, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Openmix: Reviving known knowledge for discovering novel visual categories in an open world. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9462–9470, 2021. 3

[51] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8697–8710, 2018. 1

1. Data details

Fig. 5 illustrates the data difference between traditional classification and Generalized Category Discovery (GCD). Unlike traditional classification models trained in a closed set, where both training and test data only come from labeled data, GCD operates in an open set—a more realistic and challenging setting. In GCD, the training data includes unlabeled samples that consist of both known classes (*e.g.* dog and bird) and novel classes (*e.g.* penguin and horse) without annotations. During testing, the model should accurately classify the known class samples and recognize the novel class samples.
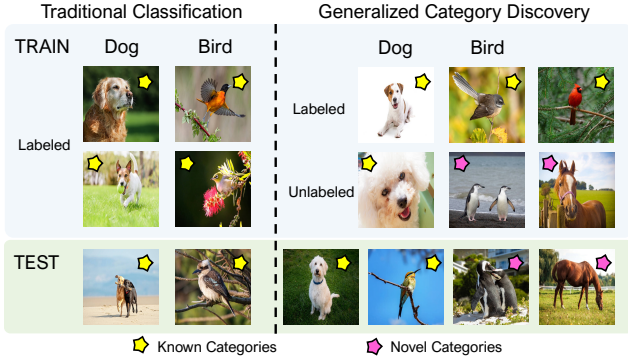

Figure 5. Data details of traditional classification and Generalized Category Discovery.

2. Training visualization

Fig. 6 shows the "Old" accuracy across training epochs for both SimGCD and our LegoGCD, employing the same random seed. Our method (depicted by green curves) consistently outperforms SimGCD (shown by orange curves) across diverse datasets. Notably, LegoGCD effectively addresses the catastrophic forgetting problem, particularly in fine-grained datasets like CUB and Stanford Cars, as well as in generic image recognition datasets CIFAR10/100 and ImageNet-100. Meanwhile, LegoGCD enhances known class accuracies, even in datasets with less pronounced forgetting, such as the unbalanced Herbarium 19. Additionally, improvements are observed in FGVA-Aircraft and ImageNet-1k datasets without forgetting.

3. Representations visualization

In this section, we employ t-distributed stochastic neighbor embedding (t-SNE) to visualize the learned representations of LegoGCD and compare them with the baseline SimGCD. The result of this comparison is presented in Fig. 7. Spectively, we randomly select 10 categories, each composed of 5 known and novel classes, with known and novel samples marked with ● and ✖, respectively. Fig. 7a and Fig. 7b display the visualizations on ImageNet-100 in SimGCD and

---

Algorithm 1 Pseudo code on one step for LegoGCD

```
#x1, x2: two view samples
#s_proj, s_pred, t_pred: projection feature,
    logits (similarities) for student and teacher
#mask: label mask
#x1_pred, x2_pred: logits of two view samples
def training_step(x1, x2):
    s_proj, s_pred = model([x1, x2])
    t_pred = s_pred.detach()
    #(1)Representation learning (unsupervised)
    unsup_con_loss = UnsupConLoss(s_proj)#Eq.(1)
    #(2) Representation learning (supervised)
    sup_con_loss = SupConLoss(s_proj, label=
    target[mask=1]) #Eq.(2)
    #(3) Supervised classification loss uses
    ground-truth labels on labeled data
    sup_loss = cross_entropy(s_pred[mask=1],
    target[mask=1]) #Eq.(5)
    #(4) Unsupervised (Self-distillation)
    classification loss on all data in Eq.(5)
    unsup_loss = cross_entropy(t_pred, s_pred)
    #(5) DKL
    x1_pred, x2_pred.detach() = s_pred.chunk(2)
    unsup_loss += DKL(x1_pred, x2_pred) #Eq.(11)
    #(6) LER in Eq.(10)
    loss_ler = LER(s_pred, s_pred+delta_logits)
    # Total representation learning loss
    loss_rep =  (1-lambda)*unsup_con_loss +
    sup_con_loss
    # Total classification loss
    loss_cls = (1-lambda)*unsup_loss + sup_loss
    # Overall loss
    loss = alpha * (loss_rep + loss_cls ) +beta*
    loss_ler #Eq.(13)
    return loss
```

---

our method, respectively. In Fig. 7a, some representations of novel classes are closer to known classes 24 and 48 than their truth labels, which are circled by red color. On the contrary, the representations of our method in known categories in Fig. 7b exhibit clear margins, indicating our method can more effectively distinguish known samples. Furthermore, Fig. 7c illustrates the logit distribution of known samples in unlabeled data. The predictions of our method exhibit higher logits, indicating enhanced sample discriminability.

4. Experimental supplements

In this section, we give detailed analyses of CIFAR10 and FGVC-Aircraft which improvements are not obvious in "Old" classes.

4.1. Results on CIFAR10

In this section, we conduct an ablation study on the confidence threshold in CIFAR10, as detailed in Tab. 8. Notably, the "Old" accuracy consistently surpasses that of SimGCD when $\delta = 0$. Despite a marginal drop in "New" accuracy ranging from 0.3 to 0.5, significant improvements are observed in "Old" accuracy, effectively mitigating the for-
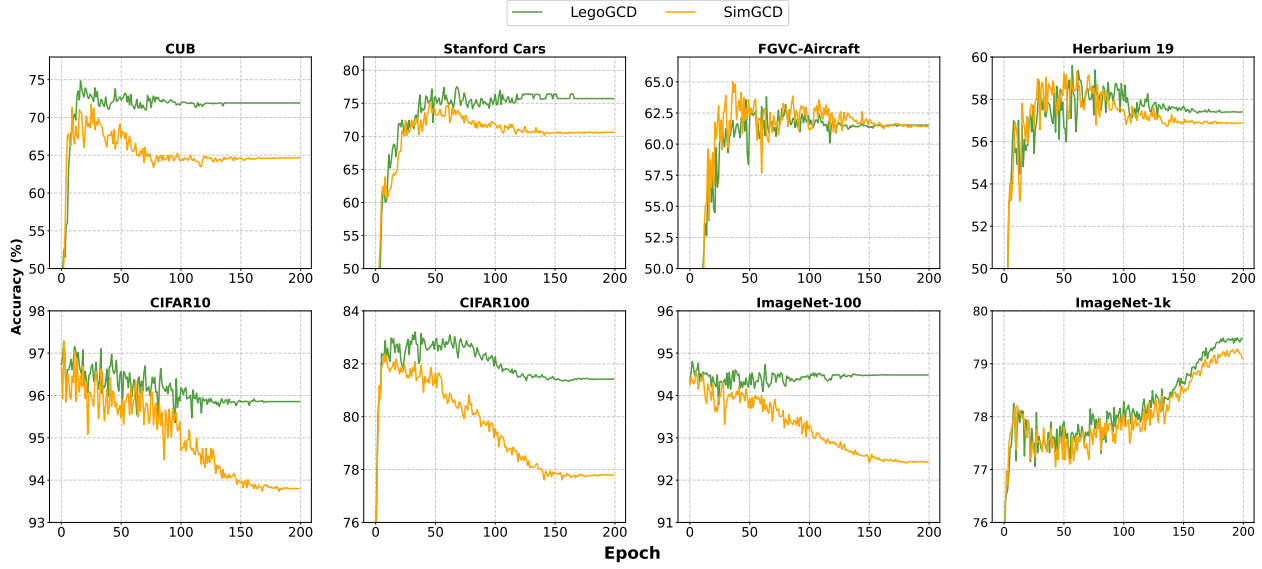
Figure 6. "Old" accuracy in each epoch compared between SimGCD and our LegoGCD. Our method (depicted in green) consistently outperforms SimGCD (shown in orange) across all datasets.

(a) ImageNet-100 on SimGCD   (b) ImageNet-100 on LegoGCD   (c) Logits distributions
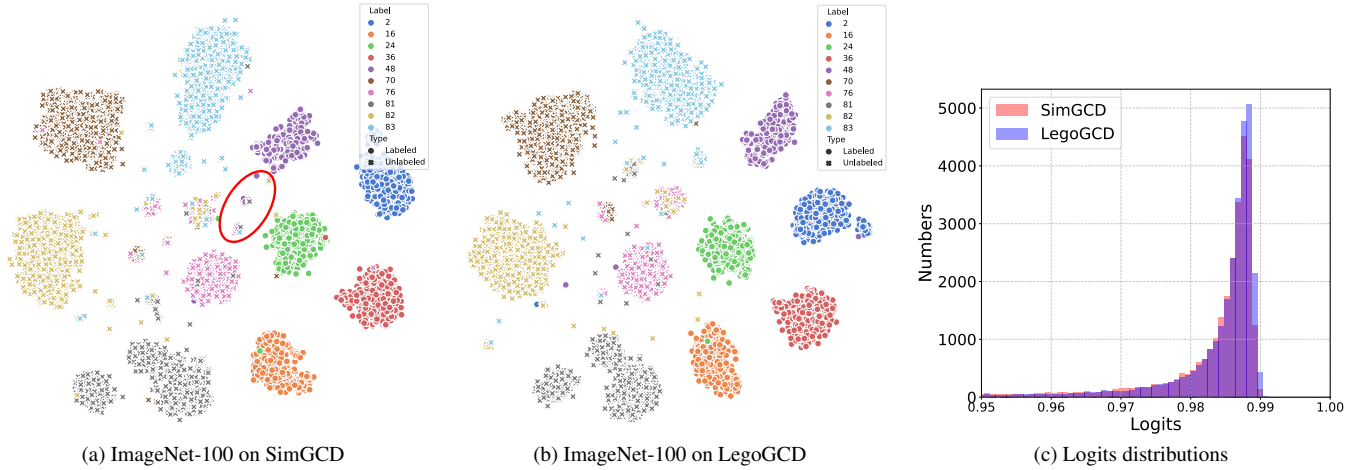
Figure 7. The t-SNE visualization and logit distributions of the unlabeled dataset for SimGCD and LegoGCD ImageNet-100.

getting problem and show significant robustness in "Old" classes. Ultimately, we select $\delta = 0.97$ as the optimal threshold. While this choice results in a 0.3% reduction in "New" accuracy, it boosts "Old" accuracy by 1.9%, leading to an overall improvement of 0.6% in "All" accuracy.

4.2. Results on FGVC-Aircraft

In Tab. 9, we analyze the accuracy in the FGVC-Aircraft dataset under different settings for comprehensive comparisons. Initially, we use the same random seed=0 in both SimGCD and LegoGCD. Subsequently, we conduct 5 training runs across SimGCD and LegoGCD without a fixed random seed and average the results. As depicted in Tab. 9, when utilizing the same random seed=0, our method only

Table 8. Ablation study on confidence threshold δ was conducted on CIFAR10. The green indicates the margins ahead SimGCD (*i.e.* δ=0), while the red donates lagging values.

δ	CIFAR10		
	All	Old	New
0.0	96.9	93.8	98.5
0.85	97.5 +0.6	96.4 +2.6	98.0 -0.5
0.90	97.4 +0.5	96.0 +2.2	98.1 -0.4
0.95	97.1 +0.2	94.3 +0.5	98.5 +0.0
0.97	97.5 +0.6	95.7 +1.9	98.2 -0.3

Table 9. The accuracy of the FGVC-Aircraft dataset compared with SimGCD and LegoGCD in different settings.

Seed	SimGCD			LegoGCD		
	All	Old	New	All	Old	New
Yes	54.6	61.4	51.1	55.0	61.5+0.1	51.7+0.6
	51.8	57.2	49.0	53.5	62.0	49.2
	52.5	58.3	49.6	54.6	60.0	51.9
No	53.8	58.8	51.3	56.1	64.2	52.0
	55.2	61.8	51.9	55.8	61.3	53.0
	56.6	60.9	54.4	56.3	62.7	53.0
Avg.	54.0±1.75	59.4±1.67	51.2±1.94	55.2±1.18	62.0±1.57	51.8±1.57

slightly outperforms SimGCD by 0.1%, as shown in Tab. 2. However, our method achieves a substantial improvement of 2.6% in "Old" classes after 5 runs. Additionally, the standard deviation of our method is 1.57 while 1.67 in SimGCD, proving LegoGCD exhibits less fluctuation than SimGCD in the FGVC-Aircraft dataset.