Information Fusion 112 (2024) 102547

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

# Prediction consistency regularization for Generalized Category Discovery

Yu Duan [a], Junzhi He [a], Runxin Zhang [b], Rong Wang [b], Xuelong Li [b], Feiping Nie [a,*]

[a] School of Computer Science, School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), and the Key Laboratory of Intelligent Interaction and Applications (Ministry of Industry and Information Technology), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, PR China

[b] School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), and the Key Laboratory of Intelligent Interaction and Applications (Ministry of Industry and Information Technology), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, PR China

ARTICLE INFO

*Keywords:*
Open-world semi-supervised learning
Generalized category discovery
Consistency regularization

ABSTRACT

Generalized Category Discovery (GCD) is a recently proposed open-world problem that aims to automatically discover and cluster based on partially labeled data. The mainstream GCD methods typically involve two steps: representation learning and classification assignment. Some methods focus on representation and design effective contrastive learning strategies and subsequently utilize clustering methods to obtain the final results. In contrast, some methods attempt to jointly optimize the linear classifier and the model, directly obtaining the predictions. However, the linear classifier is strongly influenced by supervised information, which limits its ability to discover novel categories. In this work, to address the aforementioned issues, we propose the Prediction Consistency Regularization (PCR), which combines the advantages of the aforementioned methods and achieves prediction consistency at both the representation-level and label-level. We employ the Expectation–Maximization (EM) framework to iteratively optimize the model with theoretical guarantees. On one hand, PCR overcomes the limitation of standalone clustering methods that fail to capture fine-grained information within features. On the other hand, it avoids an excessive reliance on supervised information, which can result in the linear classifier getting trapped in local optima. Finally, we comprehensively evaluate our proposed PCR on five benchmark datasets through extensive experiments, and the results demonstrate its superiority over the previous state-of-the-art methods. Our code is available at https://github.com/DuannYu/PCR.

## 1. Introduction

In the past decade, semi-supervised learning (SSL) has demonstrated superior performance in various tasks. It estimates the sample distribution by leveraging a small set of labeled samples and a large number of unlabeled ones. Most existing SSL methods assume that labeled data belong to known categories, wherein each category has a small number of samples. However, in real scenarios, many practical tasks such as intent detection [1,2] and image recognition [3] are*open-world*. Therefore, well-trained SSL models cannot achieve satisfactory performance on unseen categories.

To handle these issues, Novel Category Discovery (NCD) has attracted the researchers' attentions. In this setting, NCD aims to categorize samples from an unlabeled dataset, referred to as novel samples, into distinct classes by utilizing a set of labeled samples from known categories. Once this setting has been formalized, many researchers follow and propose a large number of improved methods, such as self-supervised pre-training [4], multi-view self-labeling [5], mixup augmentation [6], and meta learning [7]. However, in the real world, unlabeled data consists of both known and novel categories simultaneously. Therefore, researchers proposed a novel setting suitable for these scenarios, termed Generalized Category Discovery (GCD). In GCD [8], the task is to accurately classify an unlabeled dataset, which contains both known and novel categories.

Most existing GCD methods primarily comprise two components: representation learning and classification assignment. The mainstream GCD methods often pay more attention to the former [9–11]. They utilize self-supervised learning, supervised learning [12], and even Large Language Models (LLMs) [13–15] to obtain representations for clustering, by using semi-supervised k-means (SSK) to obtain the final classification assignment. However, these two-step separate strategies may not fully leverage the advantages of well-trained representations. In contrast, another perspective is constructing a linear classifier followed by representation learning, and jointly optimizing both of them to obtain class assignment, such as leveraging classification objectives

* Corresponding author.
*E-mail addresses:* duanyuee@gmail.com (Y. Duan), hejunzhi@mail.nwpu.edu.cn (J. He), zhangrunxin66@gmail.com (R. Zhang), wangrong07@tsinghua.org.cn (R. Wang), li@nwpu.edu.cn (X. Li), feipingnie@gmail.com (F. Nie).

https://doi.org/10.1016/j.inffus.2024.102547
Received 4 March 2024; Received in revised form 4 June 2024; Accepted 21 June 2024
Available online 25 June 2024

including cross entropy, self-distillation loss [16], and maximizing mutual information [17]. Every coin has two sides, these jointly learning strategies may cause the model to over-fit to the supervised information during the early training stage and become trapped in local optima, limiting its ability to discover novel categories.

To overcome these limitations and leverage the advantages of both types of methods, in this paper, we propose a novel category discovery method called Prediction Consistency Regularization (PCR). In a word, we expect that the linear classifier based on representation learning and SSK give the classification prediction consistency. Specifically, PCR comprises two main components: Classification Distribution Consistency (CDC) and Representation Learning Consistency (RLC). In CDC, our objective is to achieve classification prediction consistency between these two types of classifiers. In RLC, we combine the pseudo-labels generated by the two classifiers with the embeddings from representation learning, calculate the corresponding cluster centers, and ensure that the centers have the same distributions. Finally, we optimize PCR using an expectation–maximization (EM) framework. In the E-step, we estimate the pseudo-labels of the samples using SSK, and in the M-step, we minimize the proposed objective functions. The main contributions of this paper are summarized as follows:

- We present a novel method called Prediction Consistency Regularization (PCR), which effectively combines the advantages of representation learning and a linear classifier to achieve state-of-the-art performance in GCD.
- We introduce two components, Classification Distribution Consistency (CDC) and Representation Learning Consistency (RLC), which ensure consistent predictions both in the label space and representation space, respectively.
- By optimizing PCR in an EM framework, our method consistently achieves superior performance over state-of-the-art GCD methods on both generic and fine-grained tasks.

## 2. Related works

### 2.1. Semi-supervised learning

The mainstream of Semi-supervised Learning (SSL) is based on consistency regularization to achieve state-of-the-art performance. In this setting, it is expected that the classifier provides consistent predictions across different data augmentation views. For instance, ReMixMatch [18] and FixMatch [19] utilize confident pseudo-labels obtained from weakly augmented views to guide the corresponding strongly augmented views. To further enhance performance, FreeMatch [20] and Dash [21] aim to select confident pseudo-labels using adaptive thresholds rather than fixed ones. However, the data augmentations used in these works are all at pixel level, including CutOut [22], AutoAugment [23], and RandAugment [24], etc. They often lead to limited diversity for augmented samples. To handle above issues, ISDA [25] designed a semantic-level data augmentation method motivated by the linear characteristic of deep features. Furthermore, PLSP [26] proposed a semantic consistency regularization that bring spatial label learning to semi-supervised learning.

To this end, the aforementioned methods are under a closed-set setting, where all labeled and unlabeled data are assumed to belong to known categories. However, in numerous real-world scenarios, this simple and crude assumption often fails to hold, particularly when the unlabeled data is collected from unconstrained environments.

### 2.2. Novel category discovery

In order to extend SSL to a more realistic scenario, Novel Category Discovery (NCD) [27] relaxes the closed-set assumption and aims to discover and classify instances belonging to unknown or novel classes. NCD is achieved under a weakly supervised setting where a labeled set of known classes is provided during training. The initial works of NCD predominantly involve two steps [27–29]. The first step involves representation learning, while the second step focuses on transfer learning for discovering novel category. Recent works [30–33] concentrate on representation learning for both labeled and unlabeled samples, employing separate classification heads. For instance, RankStat [31] suggests that self-supervised pre-training offers advantages in obtaining pseudo-labels. UNO [30] introduces a unified objective function for training by leveraging both unlabeled and labeled samples.

### 2.3. Generalized category discovery

Generalized Category Discovery (GCD) [8], also known as *open-world* semi-supervised learning, represents an extension of NCD. Specifically, GCD allows for the existence of unlabeled data in both known and novel categories. Similar to NCD, GCD also mainly consist of two parts, representation learning and classification assignment. Vaze et al. [8] first formalize the problem of GCD. They employ self-supervised and supervised contrastive learning to obtain cluster-favorable representations, followed by semi-supervised k-means (SSK) for final classification predictions. After the initialization of this setting, numerous researchers have proposed methods to enhance the performance of GCD. Previous works often concentrate on representation learning. For instance, DCCL [13] employs InfoMap [34] to obtain conceptual prototypes (cluster centers) and makes samples towards the nearest prototypes. AMEND [35] incorporates expanded neighborhood information in contrastive learning to generate robust features, resulting in superior performance on fine-grained datasets. In contrast, SimGCD [16] simplifies the framework presented by GCD [8]. It introduces linear classifier with the help of prototype vectors, jointly optimizing the projector and classifier to obtain pseudo-labels.

More recently, with the successful development of Large Language Model (LLM), prompt tuning has emerged as a powerful technique in the field of Natural Language Processing (NLP). It has also been extended to images through visual prompt learning [36]. For example, CLIP-GCD [14] utilizes CLIP's vision-language representations to retrieve the top-k relevant text segments and incorporates their embeddings for semi-supervised clustering of joint image and text data. PromptCAL [15] uses these multi-module representations to provide a weaker semantic supervision information.

Moreover, GCD, as an open-set semi-supervised learning task, whether the number of classes is a priori is also one of the key steps in designing the strategies. If the number of classes is unknown, methods often need to use extra steps to estimate it. For example, the authors in Ref. [37] use the elbow method to adjust the number of classes. In Ref. [38], the authors utilize the Silhouette Coefficient to dynamically split clusters and determine the final cluster numbers. Moreover, in Ref. [9], the authors preset different categories numbers and run semi-supervised k-means to observe performance to find the optimal one. On the other hand, when the number of clusters is known, researchers tend to focus more on representation learning and classifier training. For example, such as SimGCD [16] and PIM [17], which can directly output classification predictions without any additional clustering steps.

## 3. Proposed method

### 3.1. Problem settings and method overview

We first introduce the setting of GCD, which aims to discover novel categories by leveraging known classes knowledge. Given a dataset $\mathcal{D}$ consists of two parts, a labeled set $\mathcal{D}_{\mathcal{L}} = \{(\mathbf{x}_i, \mathbf{y}_i)_{i=1}^{N} \in \mathcal{X} \times \mathcal{Y}_{\mathcal{L}}\}$ and unlabeled set $\mathcal{D}_{\mathcal{U}} = \{\mathbf{x}_i\}_{i=1}^{M}$. The images in unlabeled set belong to $\mathcal{Y}_{\mathcal{U}}$ and $\mathcal{Y}_{\mathcal{L}} \subset \mathcal{Y}_{\mathcal{U}}$. During the train stage, the model could not access the labels in $\mathcal{D}_{\mathcal{U}}$. In this work, we assume the number of known categories $|C_k|$ and novel categories $|C_n|$ are known, and the total number of categories is $|C| = |C_k| + |C_n|$.

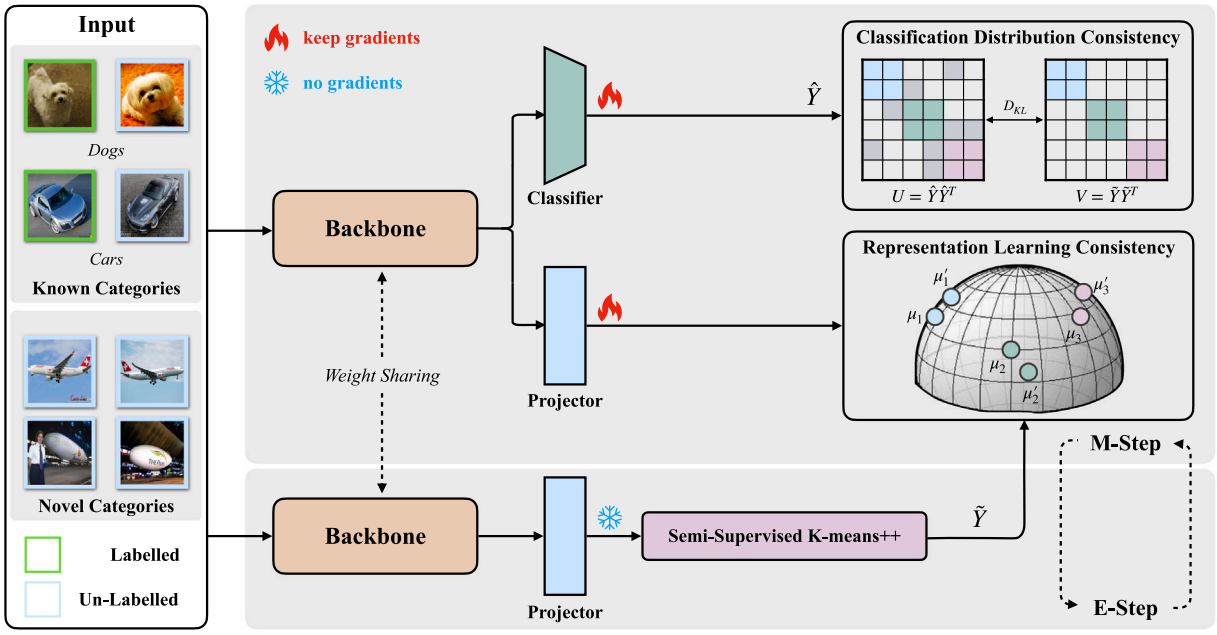


**Fig. 1.** The overview of the proposed PCR in an EM framework.

To leverage the advantages of representation learning and classification assignment, this paper introduces a method for discovering novel categories called Prediction Consistency Regularization (PCR). As shown in Fig. 1, PCR integrates information from both a linear classifier and SSK, enhancing the representation learning process. Formally, for a given image $\mathbf{x}_i$, we obtain $l_2$-normalized embedding $\mathbf{z}_i = \phi(f(\mathbf{x}_i))$ by employing a feature extraction backbone $f$ and MLP projector $\phi$. Additionally, we define two classification predictions: $\hat{\mathbf{y}}_i = \psi(f(\mathbf{x}_i))$ from the linear classifier $\psi$ and $\tilde{\mathbf{y}}_i$ from SSK. During the training stage, to ensure that $\hat{\mathbf{y}}_i$ and $\tilde{\mathbf{y}}_i$ have the same distributions, we propose PCR. PCR incorporates two novel techniques: Representation Learning Consistency (RLC) and Classification Distribution Consistency (CDC), as detailed in Section 3.2. We subsequently explain the process of obtaining cluster-favorable embeddings and label assignment in Section 3.3. Lastly, we present the complete objective functions of PCR and its EM optimization with theoretical guarantees in Section 3.4.

### 3.2. Prediction consistency regularization

To address this performance degrades and draw inspiration from recent GCD methods, we propose Prediction Consistency Regularization (PCR) to leverage the advantages of both types of classifiers. PCR primarily comprises two components: Classification Distribution Consistency (CDC) and Representation Learning Consistency (RLC). The former explicitly ensures that predictions from different classifiers have the same distributions, while the latter ensures that samples with the same pseudo-labels are close to each other, which implicitly leads to classifiers giving the same classification assignments. Subsequently, we will provide a detailed description of them.

#### 3.2.1. Classification Distribution Consistency (CDC)

Previous close-set SSL methods based on prediction consistency have shown the significant improvements in many tasks [20,21,26]. However, as mentioned above, directly introducing these ideas to GCD is not a good idea. On one hand, linear classifiers tend to assign samples to known classes (we will discuss later in experimental parts). On the other hand, SSK solely rely on good feature representations such that samples from the same class will be close to a prototype while stay far from others. The different characteristics of these two classifiers motivate us to design a loss to encourage the consistency between

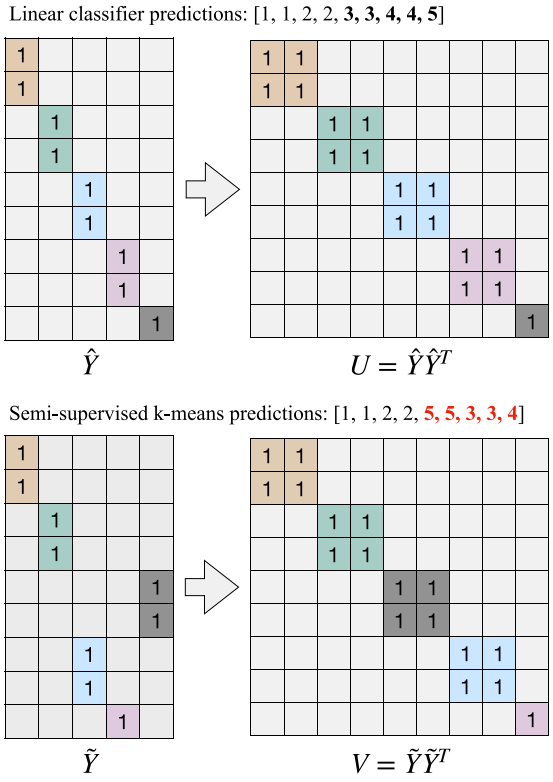


**Fig. 2.** The illustration of pseudo-labels mismatch.

their predictions on all samples to regularize the feature representations. However, under the *open-world* setting, classification assignment inevitably mismatch between pseudo-labels and ground truth. Fig. 2 show a simple example of pseudo-labels mismatch. Here, we propose to use pairwise similarity consistency between each prediction to avoid above issues. Specifically, in a mini-batch $\mathcal{B}$, denote $\hat{\mathbf{Y}}, \tilde{\mathbf{Y}} \in \mathcal{R}^{|\mathcal{B}|\times|C|}$ are all samples prediction assignments from linear classifier and SSK, respectively. We first compute their pairwise similarities of prediction

$\mathbf{U} = \hat{\mathbf{Y}}\hat{\mathbf{Y}}^T$ and $\mathbf{V} = \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T$, respectively. And then we minimize the Kullback–Leibler (KL) divergence between both of them to achieve the prediction consistency, which can be written as

$$\mathcal{L}_{cdc} = \mathcal{D}_{KL}(\mathbf{U} \parallel \mathbf{V}), \tag{1}$$

where $\mathbf{U} = \hat{\mathbf{Y}}\hat{\mathbf{Y}}^T$ and $\mathbf{V} = \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T$.

#### 3.2.2. Representation Learning Consistency (RLC)

As we all know, the prediction and confidence of pseudo-labels play a crucial role in SSL. Existing clustering methods often achieve cluster-favorable representations by utilizing reliable pseudo-labels. For example, TCL [39] and NN [40] employ pseudo-labels to enhance cluster performance. Proxy Anchor Loss [41] treats pseudo-labels as supervised information, encouraging samples to be close to their corresponding anchors. However, merely pulling samples belonging to the same class together may lead to trivial solutions. To handle these issues, [42] proposes an orthogonality regularization to learn prototype vectors that represent distinct characteristics.

Motivated by the aforementioned observations, we propose a representation learning consistency (RLC). Specifically, we combine the pseudo-labels generated two types of classifiers and treat them as the targets. Next, we compute cluster centers based on the embeddings and the two types of pseudo-labels to ensures a consistent distribution of the cluster centers in the latent space. Formally, let the predictions of the linear classifier and SSK be denoted as $\hat{\mathbf{y}}_i$ and $\tilde{\mathbf{y}}_i$ respectively. Within a mini-batch, we obtain two types of cluster centers by combining the embeddings $\mathbf{z}$ with their pseudo-labels. Here, we refer to the centers generated by the linear classifier as parametric centers, and they can be expressed as

$$\hat{\boldsymbol{\mu}}_i = \frac{\sum_{j\in\mathcal{B}_l} \mathbf{z}_j \cdot \delta(\mathbf{y}_j = i) + \sum_{j\in\mathcal{B}_u} \mathbf{z}_j \cdot \delta(\hat{\mathbf{y}}_j = i)}{\parallel \sum_{j\in\mathcal{B}_l} \mathbf{z}_j \cdot \delta(\mathbf{y}_j = i) + \sum_{j\in\mathcal{B}_u} \mathbf{z}_j \cdot \delta(\hat{\mathbf{y}}_j = i) \parallel}. \tag{2}$$

Similarly, we define SSK-centers generated by the SSK as follow,

$$\tilde{\boldsymbol{\mu}}_i = \frac{\sum_{j\in\mathcal{B}_l} \mathbf{z}_j \cdot \delta(\mathbf{y}_j = i) + \sum_{j\in\mathcal{B}_u} \mathbf{z}_j \cdot \delta(\tilde{\mathbf{y}}_j = i)}{\parallel \sum_{j\in\mathcal{B}_l} \mathbf{z}_j \cdot \delta(\mathbf{y}_j = i) + \sum_{j\in\mathcal{B}_u} \mathbf{z}_j \cdot \delta(\tilde{\mathbf{y}}_j = i) \parallel}. \tag{3}$$

It should be noted that the different predictions are matched by Kuhn-Munkres algorithm [43]. After getting two types of centers, we propose the following RLC loss to ensure the prediction consistency and impose them to be uncorrelated,

$$\mathcal{L}_{rlc} = -\frac{1}{|\mathcal{C}_{\mathcal{U}}|} \sum_{i\in|\mathcal{C}_{\mathcal{U}}|} \log \frac{\exp\left(\hat{\boldsymbol{\mu}}_i^T \tilde{\boldsymbol{\mu}}_i/\tau\right)}{\sum_{i\neq j} \exp\left(\hat{\boldsymbol{\mu}}_i^T \tilde{\boldsymbol{\mu}}_j/\tau\right)}. \tag{4}$$

As shown in Eq. (4), RLC has following advantages:

- The numerator of Eq. (4) implicitly ensures consistency between the two classifiers in terms of their distributions. By leveraging SSK results obtained from global data, it guides the linear classifier to produce more confident results.
- RLC allows features from different categories to be as far apart as possible, making the clusters have more clear boundaries and facilitating the discovery of novel classes.
- RLC obtains a large number of pseudo-labels of unlabeled samples that successfully introduce SSK into the model training, thereby improving the model's performance.

Finally, the overall objective function of PCR is written as follow,

$$\mathcal{L}_{pcr} = \alpha\mathcal{L}_{cdc} + \beta\mathcal{L}_{rlc}, \tag{5}$$

where $\alpha$ and $\beta$ are trade-off parameters.

### 3.3. Representation learning and classification prediction

Previous sections discuss how to achieve the classification predictions distribute consistently. As we all know, extracting features and suitable classifiers play vital roles in obtaining good performance. In this section, we briefly introduce how to obtain meaningful representations and make classifiers give more convincing predictions.

#### 3.3.1. Representations learning

Inspired by [13,16], we combine self-supervised and supervised contrastive loss for representation learning. Formally, given two embeddings $\mathbf{z}_i$ and $\mathbf{z}'_i$ extracting from two random augmentation views $\mathbf{x}_i$ and $\mathbf{x}'_i$, we jointly optimize a self-supervised contrastive loss on all samples in a mini-batch $\mathcal{B}$, such as

$$\mathcal{L}^u_{fea} = -\frac{1}{|\mathcal{B}|} \sum_{i\in\mathcal{B}} \log \frac{\exp\left(\mathbf{z}_i^T \mathbf{z}'_i/\tau_u\right)}{\sum_{i\in\mathcal{B}, i\neq j} \exp\left(\mathbf{z}_i^T \mathbf{z}'_j/\tau_u\right)}, \tag{6}$$

and supervised contrastive loss on labeled sets, i.e.,

$$\mathcal{L}^s_{fea} = -\frac{1}{|\mathcal{B}_l|} \sum_{i\in\mathcal{B}^l} \frac{1}{|\mathcal{P}_i|} \sum_{q\in\mathcal{P}_i} \log \frac{\exp\left(\mathbf{z}_i^T \mathbf{z}'_q/\tau_s\right)}{\sum_{i\in\mathcal{B}, i\neq j} \exp\left(\mathbf{z}_i^T \mathbf{z}'_j/\tau_s\right)}, \tag{7}$$

where $\tau_u$ and $\tau_s$ are temperature values, $\mathcal{P}_i$ denotes a positive set that shares the sample labels with $\mathbf{x}_i$, and $\mathcal{B}_l$ is a subset of $\mathcal{B}$ consisting of all labeled data in mini-batch. Finally, the representation learning objective function is written as follow,

$$\mathcal{L}_{fea} = \lambda\mathcal{L}^s_{fea} + (1-\lambda)\mathcal{L}^u_{fea}, \tag{8}$$

where $\lambda$ is also a trade-off parameter.

#### 3.3.2. Classification prediction

To this end, we need to introduce how obtain the classification predictions $\hat{\mathbf{y}}_i$ and $\tilde{\mathbf{y}}_i$. For parametric predictions $\hat{\mathbf{y}}_i$, we use cross entropy loss on labeled samples, i.e.,

$$\mathcal{L}^s_{cls} = l_{\mathrm{CE}}(\mathbf{y}_i, \hat{\mathbf{y}}_i), \tag{9}$$

where $l_{\mathrm{CE}}(\mathbf{a}, \mathbf{b}) = -\langle \mathbf{a}, \log \mathbf{b} \rangle$, $\mathbf{y}_i$ is one-hot label of $\mathbf{x}_i$. Similarly, we use cross entropy loss between predictions and pseudo-labels on unlabeled samples

$$\mathcal{L}^u_{cls} = l_{\mathrm{CE}}(\hat{\mathbf{y}}_i, \hat{\mathbf{y}}'_i) - \epsilon H(\bar{\mathbf{y}}), \tag{10}$$

where $\hat{\mathbf{y}}'_i$ is another view output sharpened by temperature factor $\sigma$, and $H(\bar{\mathbf{y}}_i)$ denotes the mean-entropy maximization regularization [44], and $\epsilon$ is hyper-parameter preventing classifier from non-trivial solutions. To this end, the overall loss for training linear classifier is

$$\mathcal{L}_{cls} = \lambda\mathcal{L}^s_{cls} + (1-\lambda)\mathcal{L}^u_{cls}, \tag{11}$$

where $\lambda$ is also a trade-off parameter.

Additionally, we perform the SSK at the beginning of each epoch to obtain the $\tilde{\mathbf{y}}_i$, which is fixed in the latter training to optimize the proposed PCR.

### 3.4. EM framework optimization

In this section, we firstly introduce how to optimize the whole framework of PCR, and then provide the theoretical analysis.

#### 3.4.1. Optimization procedure

During the whole training process, we alternately perform SSK and model training, until convergence. The optimization of PCR is done in an EM framework, where **E-step** and **M-step** are detailed as follows.

**E-Step**: This step mainly aims to obtain the cluster assignments of SSK. At the beginning of each epoch, we apply SSK to assign all samples to obtain their classification predictions. Specifically, we modify the traditional k-means into a constrained one by ensuring that instances in $\mathcal{D}_{\mathcal{L}}$ are assigned to the correct cluster based on their respective ground-truth labels. The initial $|\mathcal{Y}_{\mathcal{L}}|$ centers for $\mathcal{D}_{\mathcal{L}}$ are obtained based on the ground-truth labels, and the remaining $|\mathcal{Y}_{\mathcal{U}} \backslash \mathcal{Y}_{\mathcal{L}}|$ novel cluster centers are obtained from $\mathcal{D}_{\mathcal{U}}$ using k-means++ [45], constrained on the centers of $\mathcal{D}_{\mathcal{L}}$. During each center update, samples from the same class in $\mathcal{D}_{\mathcal{L}}$ are consistently assigned to the same cluster, while each sample in $\mathcal{D}_{\mathcal{U}}$ can be assigned to any cluster based on its distance to each center. Once the SSK converges, each sample in $\mathcal{D}_{\mathcal{U}}$ can be

assigned a cluster label. Moreover, we will update the representations memory buffer at the end of each mini-batch.

**M-Step**: As shown in Fig. 1, PCR is consist of three parts, including linear classifier, projector and SSK, and the first two parts need to be updated by gradient descend. According from Eq. (5) to Eq. (11), the overall loss of our proposed methods is

$$\mathcal{L} = \mathcal{L}_{fea} + \mathcal{L}_{cls} + \mathcal{L}_{pcr}. \tag{12}$$

Finally, we leverage above objective function to train model.

#### 3.4.2. Theoretical analysis

Given the input image $\mathbf{X}$, model parameters $\theta$ and the classification predictions $\tilde{\mathbf{Y}}$ obtained from SSK, to estimate the model parameters $\theta$, it is common to introduce the log likelihood function $\theta = \ln \mathcal{P}(\mathbf{X}|\theta)$, which denotes the likelihood of parameters $\theta$ given the data $\mathbf{X}$.

The EM framework is an iterative procedure for maximizing $\mathcal{L}(\theta)$. Let $\theta_t$ be the current estimate for $\theta$ after the $t$th iteration. Our objective is to compute and estimate maximizes $\mathcal{L}(\theta)$.

$$\begin{aligned}
&\mathcal{L}(\theta) - \mathcal{L}(\theta_t) \\
&= \ln \mathcal{P}(\mathbf{X}|\theta) - \ln \mathcal{P}(\mathbf{X}|\theta_t) \\
&= \ln \left( \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta) \right) - \ln \mathcal{P}(\mathbf{X}|\theta_t) \\
&= \ln \left( \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta) \cdot \frac{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t)} \right) - \ln \mathcal{P}(\mathbf{X}|\theta_t) \\
&= \ln \left( \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \frac{\mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t)} \right) - \ln \mathcal{P}(\mathbf{X}|\theta_t) \\
&\geq \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \left( \frac{\mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta_t) \mathcal{P}(\tilde{\mathbf{Y}}|\theta)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t)} \right) - \ln \mathcal{P}(\mathbf{X}|\theta_t) \\
&= \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \left( \frac{\mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \mathcal{P}(\mathbf{X}|\theta_t)} \right) \\
&= \mathcal{H}(\theta|\theta_t),
\end{aligned} \tag{13}$$

where $\mathcal{H}(\theta|\theta_t)$ is defined by the negated sum it is replacing. Then, we have the following inequality

$$\mathcal{L}(\theta) \geq \mathcal{L}(\theta_t) + \mathcal{H}(\theta|\theta_t). \tag{14}$$

Our objective is to maximize the function $\mathcal{L}(\theta)$ by choosing an appropriate value for $\theta$. Let $l(\theta|\theta_t) = \mathcal{L}(\theta_t) + \mathcal{H}(\theta|\theta_t)$, which is bounded above by the likelihood function $\mathcal{L}(\theta_t)$. Therefore, increasing $\mathcal{H}(\theta|\theta_t)$ will also increase $\mathcal{L}(\theta)$. To achieve the greatest increase in $\mathcal{L}(\theta)$, the EM algorithm selects an updated value $\theta_{t+1}$ that maximizes $l(\theta|\theta_t)$.

$$\theta_{t+1} = \arg\max_{\theta} \{\mathcal{L}(\theta_t) + \mathcal{H}(\theta|\theta_t)\}. \tag{15}$$

Ignoring terms which are constant w.r.t. $\theta$, the equation can be further deduced:

$$\begin{aligned}
\theta_{t+1} &= \arg\max_{\theta} \left\{ \mathcal{H}(\theta|\theta_t) \right\} \\
&= \arg\max_{\theta} \left\{ \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \left( \frac{\mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \mathcal{P}(\mathbf{X}|\theta_t)} \right) \right\} \\
&= \arg\max_{\theta} \left\{ \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \left( \frac{\mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta)}{\mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \mathcal{P}(\mathbf{X}|\theta_t)} \right) \right\} \\
&= \arg\max_{\theta} \left\{ \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \mathcal{P}(\mathbf{X}|\tilde{\mathbf{Y}}, \theta) \mathcal{P}(\tilde{\mathbf{Y}}|\theta) \right\} \\
&= \arg\max_{\theta} \left\{ \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \frac{\mathcal{P}(\mathbf{X}, \tilde{\mathbf{Y}}, \theta)}{\mathcal{P}(\tilde{\mathbf{Y}}, \theta)} \frac{\mathcal{P}(\tilde{\mathbf{Y}}, \theta)}{\mathcal{P}(\theta)} \right\} \\
&= \arg\max_{\theta} \left\{ \sum_{\tilde{\mathbf{Y}}} \mathcal{P}(\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t) \ln \mathcal{P}(\mathcal{X}, \tilde{\mathbf{Y}}|\theta) \right\} \\
&= \arg\max_{\theta} \left\{ \mathbf{E}_{\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t} \left[ \ln \mathcal{P}(\mathcal{X}, \tilde{\mathbf{Y}}|\theta) \right] \right\}.
\end{aligned} \tag{16}$$

The alternate training algorithm thus consists of iterating: (1) **E-step**: Determine the conditional expectation $\mathbf{E}_{\tilde{\mathbf{Y}}|\mathbf{X}, \theta_t}[\ln \mathcal{P}(\mathbf{X}, \tilde{\mathbf{Y}}|\theta)]$ according SSK and (2) **M-step**: Maximize this expression with respect to $\theta$. It is evident that end-to-end training for maximizing $\mathcal{L}(\theta)$ is not equivalent to iterative training. The advantage of two-stage learning is that it provides a framework for better estimation for both model and classification prediction.

**Table 1**
Details and statistics of the datasets.

| Datasets | Balance | Labeled | | Unlabeled | |
|---|---|---|---|---|---|
| | | Image | Class | Image | Class |
| FGVC-Aircraft | ✓ | 1.7K | 50 | 5.0K | 100 |
| CUB | ✓ | 1.5K | 100 | 4.5K | 200 |
| Stanford Cars | ✓ | 2.0K | 98 | 6.1K | 196 |
| CIFAR10 | ✓ | 12.5K | 5 | 37.5K | 10 |
| CIFAR100 | ✓ | 20.0K | 80 | 30.0K | 100 |

## 4. Experiments

### 4.1. Experimental settings

#### 4.1.1. Dataset

To thoroughly validate the effectiveness of the proposed method, we conducted experiments on five commonly used datasets. These datasets consist of three fine-grained image classification datasets including FGVC-Aircraft [46], CUB [47], and Stanford Cars [48], as well as two general image recognition datasets, including CIFAR-10 and CIFAR-100 [49]. Following previous works [8,13], we split the categories into labeled (known) classes $\mathcal{Y}_{\mathcal{L}}$ and unlabeled (novel) classes $\mathcal{Y}_{\mathcal{U}}$, where $\mathcal{Y}_{\mathcal{L}} \subset \mathcal{Y}_{\mathcal{U}}$ is a subset of all classes. Subsequently, the top 50% of the images from the labeled classes are selected as labeled data $\mathcal{D}_{\mathcal{L}}$, while the remaining images in the dataset are used as unlabeled data $\mathcal{D}_{\mathcal{U}}$. For example, in FGVC-Aircraft, which has 100 classes, we consider classes 0–49 as labeled classes and classes 50–99 as unlabeled ones.

Table 1 summarizes the detailed statistics and separation of datasets.

#### 4.1.2. Evaluation protocol

Similar to [8,16,17], we estimate the performance using clustering accuracy (ACC), which can be written as follow:

$$\text{ACC} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I} \left( \Omega(\hat{\mathbf{y}}) = \mathbf{y}^* \right), \tag{17}$$

where $N = |\mathcal{D}_{\mathcal{U}}|$, and $\Omega$ represents the optimal matching of predicted labels $\hat{\mathbf{y}}$ with ground truth labels $\mathbf{y}^*$.

#### 4.1.3. Implementation details

Following the prior GCD works [8,13], our backbone network utilizes a ViT-B/16 model pre-trained with DINO. The feature representation of the images consists of the output from the [CLS] token with a dimensionality of 768. Only the last block of the backbone is fine-tuned. We use nearest neighbors to construct mini-batches, with each batch having a size of 125, composed of 25 image samples and their four nearest neighbors. The training epoch is set to 200, the initial learning rate is set to 0.1, and cosine annealing is applied to decay the learning rate on each dataset. We introduce the PCR at the beginning of 60-th epoch. For fair comparison, the balance factor $\lambda$ is set to 0.35, and the temperature parameters $\tau_u$ and $\tau_c$ are set to 0.07 and 1.0, respectively. All experiments are conducted using an NVIDIA GeForce RTX 3090 GPU.

### 4.2. Comparison with state-of-the-art

In this section, we compare our proposed PCR method with ten state-of-the-art GCD methods, including GCD [9], GPC [10], CAC [11], RS+[31], UNO+[30], ORCA [51], PIM [17], SimGCD [16] and GCA [12]. K-means and CC [50] are two baseline methods. We categorize

**Table 2**
Comparison results (%) with state-of-the-art methods.

| Methods | CIFAR-10 | | | CIFAR-100 | | | FGVC-Aircraft | | | CUB | | | Stanford Cars | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | All | Known | Novel | All | Known | Novel | All | Known | Novel | All | Known | Novel | All | Known | Novel |
| k-means | 83.6 | 85.7 | 82.5 | 52.0 | 52.2 | 50.8 | 16.0 | 14.4 | 16.8 | 34.3 | 38.9 | 32.1 | 12.8 | 10.6 | 13.8 |
| GCD[a] [9] | 91.5 | 97.9 | 88.2 | 73.0 | 76.2 | 66.5 | 45.0 | 41.1 | 46.9 | 51.3 | 56.6 | 48.7 | 39.0 | 57.6 | 29.9 |
| GPC[a] [10] | 90.6 | 97.6 | 87.0 | 75.4 | **84.6** | 60.1 | 46.3 | 42.5 | 47.9 | 55.4 | 58.2 | 53.1 | 42.8 | 59.2 | 32.8 |
| CAC[a] [11] | 92.3 | 91.4 | 94.4 | 78.5 | 81.4 | 75.6 | – | – | – | 58.0 | 65.0 | 43.9 | 47.6 | 70.6 | 33.8 |
| CC [50] | 61.6 | 76.4 | 54.2 | 54.7 | 64.3 | 35.5 | 38.7 | 54.7 | 30.7 | 46.9 | 48.6 | 46.1 | 33.5 | 67.3 | 17.7 |
| RS+ [31] | 46.8 | 19.2 | 60.5 | 58.2 | 77.6 | 19.3 | 26.9 | 36.4 | 22.2 | 33.3 | 51.6 | 24.2 | 28.3 | 61.8 | 12.1 |
| UNO+ [30] | 68.6 | **98.3** | 53.8 | 69.5 | 80.6 | 47.2 | 40.3 | 56.4 | 32.2 | 35.1 | 49.0 | 28.1 | 35.5 | 70.5 | 18.6 |
| ORCA [51] | 81.8 | 86.2 | 79.6 | 69.0 | 77.4 | 52.0 | 22.0 | 31.8 | 17.1 | 35.3 | 45.6 | 30.2 | 23.5 | 50.1 | 10.7 |
| PIM [17] | 94.7 | 97.4 | 93.3 | 78.3 | 84.2 | 66.5 | – | – | – | 62.7 | **75.7** | 56.2 | 43.1 | 66.9 | 31.6 |
| SimGCD [16] | **97.1** | 95.1 | **98.1** | 80.1 | 81.2 | 77.8 | 54.0 | 62.9 | 49.6 | 60.9 | 67.5 | 57.7 | 50.9 | 72.2 | 40.6 |
| GCA [12] | 92.8 | 94.4 | 91.9 | 76.6 | 79.5 | 70.7 | 47.1 | 57.1 | 42.2 | 62.3 | 72.0 | 57.5 | 45.4 | 65.5 | 35.6 |
| PCR (Ours) | 97.0 | 97.4 | 96.9 | **82.0** | 81.8 | **82.2** | **55.3** | **63.2** | **51.4** | **62.8** | 69.1 | **59.6** | **54.2** | **75.1** | **44.2** |

[a] Denotes the methods need to extra cluster assignment steps.

**Table 3**
Ablation study on the different components of proposed method.

| Index | Components | | | FGVC-Aircraft | | | CUB | | | Stanford Cars | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\mathcal{L}_{fea}+\mathcal{L}_{cls}$ | $\mathcal{L}_{cdc}$ | $\mathcal{L}_{rlc}$ | All | Known | Novel | All | Known | Novel | All | Known | Novel |
| (1) | ✓ | ✗ | ✗ | 54.0 | 62.9 | 49.6 | 60.9 | 67.5 | 57.7 | 50.9 | 72.2 | 40.6 |
| (2) | ✓ | ✓ | ✗ | 54.2 | 61.9 | 50.3 | 62.1 | 66.3 | 60.1 | 52.6 | 74.6 | 42.0 |
| (3) | ✓ | ✗ | ✓ | 54.1 | 62.2 | 50.1 | 61.7 | 67.7 | 58.7 | 53.4 | 75.4 | 42.8 |
| (4) | ✓ | ✓ | ✓ | **55.3** | **63.2** | **51.4** | **62.8** | **69.1** | **59.6** | **54.2** | **75.1** | **44.2** |

the above methods into two groups based on whether they use extra clustering assignment steps or not.

Table 2 presents a summary of the experimental results on five benchmark datasets, with the best results are **bolded**.

Table 2 demonstrates that PCR outperforms most GCD methods, particularly on three fine-grained datasets, highlighting the effectiveness of our method in fine-grained category discovery. Specifically, for FGVC Aircraft, CUB, and Stanford Cars, our method achieves improvements of 1.3%, 1.9%, and 3.3% over the state-of-the-art method in terms of All classes, respectively. In terms of the Novel classes, our method outperforms SimGCD by 1.8%, 1.9%, and 3.6% on FGVC Aircraft, CUB, and Stanford Cars, respectively.

However, we have observed that among fine-grained datasets poses a challenge in discovering novel classes, leading to commonly low ACC for the novel categories. Additionally, joint learning methods demonstrate superior performance in known classes, while SSK is good at discovering novel classes. It is evident that linear classifiers are sensitive to supervised information, and easy to overfit distributions of samples belonging to known classes. Further details on these phenomena will be discussed later.

### 4.3. Ablation analysis

We conduct extensive ablation experiments in Table 3. These experiments investigate the effectiveness of various components of the objective loss function, including $\mathcal{L}_{cdc}$ and $\mathcal{L}_{rlc}$, on three fine-grained datasets. Our baseline is defined as $\mathcal{L}_{fea}+\mathcal{L}_{cls}$.

The overall results from experiments (1) to (4) strongly support the effectiveness of our proposed components, demonstrating significant improvements. Specifically, when comparing experiments (2) and (3), we observe that $\mathcal{L}_{cdc}$ is effective in predicting known classes, while $\mathcal{L}_{rlc}$ yields improved clustering results for novel categories. This can be attributed to the fact that $\mathcal{L}_{cdc}$ solely relies on clustering results to ensure consistency in classifier outputs, thereby displaying less impact on feature distribution. Consequently, the model's training is primarily guided by the supervised information, leading to enhanced performance in existing classes. In contrast, $\mathcal{L}_{rlc}$ implicitly integrates both types of classification prediction information and strives to maximize the separation between samples from different classes. This method facilitates the acquisition of more discriminative features, enabling SSK to make improved global classification predictions and consequently improving the capability of parameterized classifiers in discovering novel classes.

### 4.4. Impact of trade-off hyper-parameters

In this test, we investigate the impact of varying values of $\alpha$ and $\beta$ within the ranges $S_\alpha = \{0.5, 1.0, 2.0, 5.0\}$ and $S_\beta = \{0.1, 0.5, 1.0\}$, respectively. As shown in Fig. 3, the performance can degrade when using excessively large values of $\alpha$. One possible reason for this is that larger values of $\alpha$ introduce noise from pseudo-labels, as indicated by Eq. (12), particularly in the case of Stanford Cars (bottom row of Fig. 3). These pseudo-labels heavily depend on the classifier's output and generate incorrect predictions, thereby diminishing the quality of instance-class level representations. On the other hand, a larger value of $\beta$ enables the model to extract more refined representations, signifying the effectiveness of $\mathcal{L}_{rlc}$ in creating greater separation among samples from different classes.

### 4.5. In depth analysis

At the previous parts, we always use linear classifier to obtain the final classification assignments. In this subsection, we will talk about the difference between linear classifier and SSK in depth.

#### 4.5.1. Linear classifier v.s. SSK

In this subsection, we compare the performance between SSK on E-step and linear classifier's output. Fig. 4 presents a summary of the performance of the two classifier types on four datasets. As shown in Fig. 4, linear classifier achieves better performance than SSK. Especially in the case of novel categories, the linear classifier outperforms the SSK by 5.2%, 13.6%, 13.5%, and 23.7% on the four datasets, respectively. This suggests that the linear classifier has the ability to explore more fine-grained semantic relationships by leveraging additional supervised information and self-distillation loss.

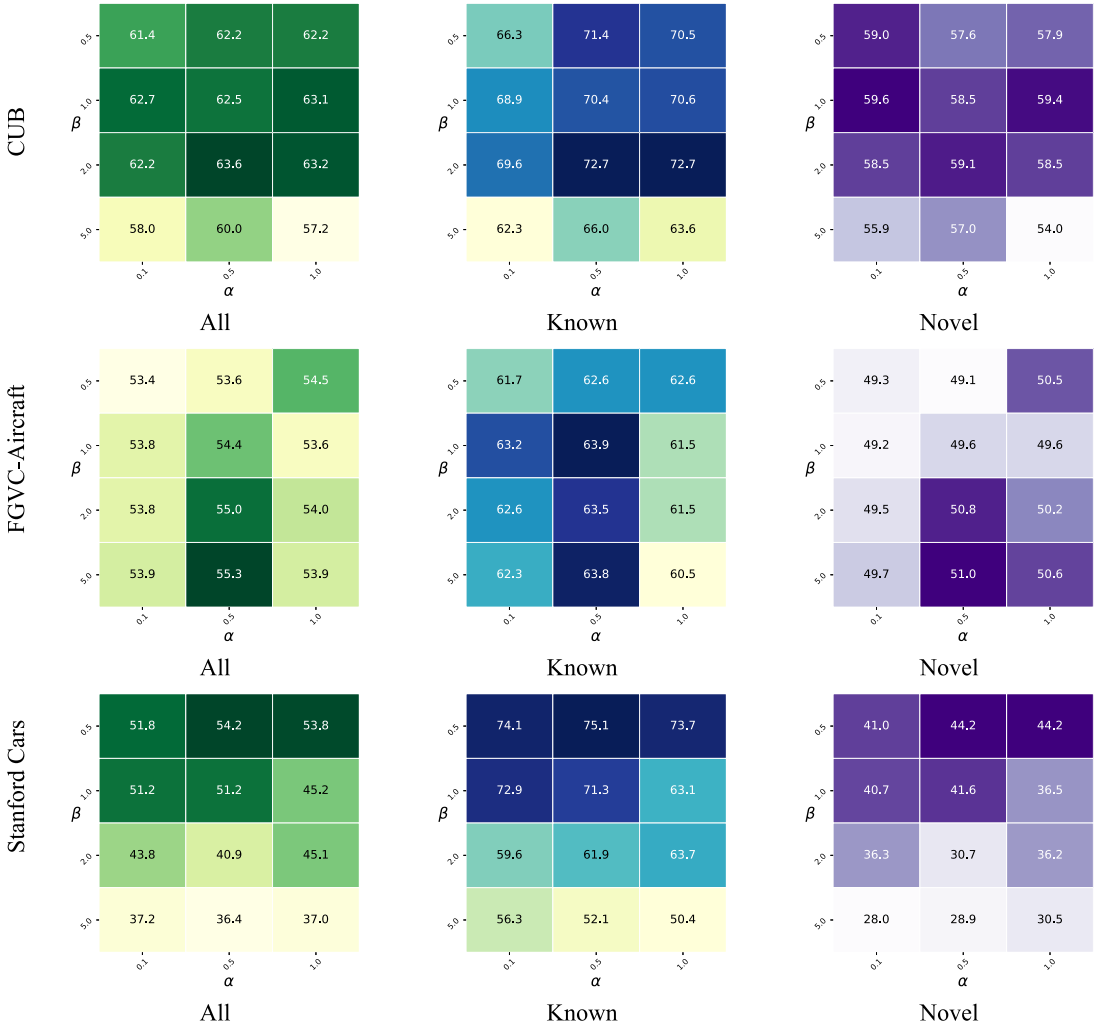


**Fig. 3.** The effect of the trade-off hyperparameters on All, Known and Novel categories.

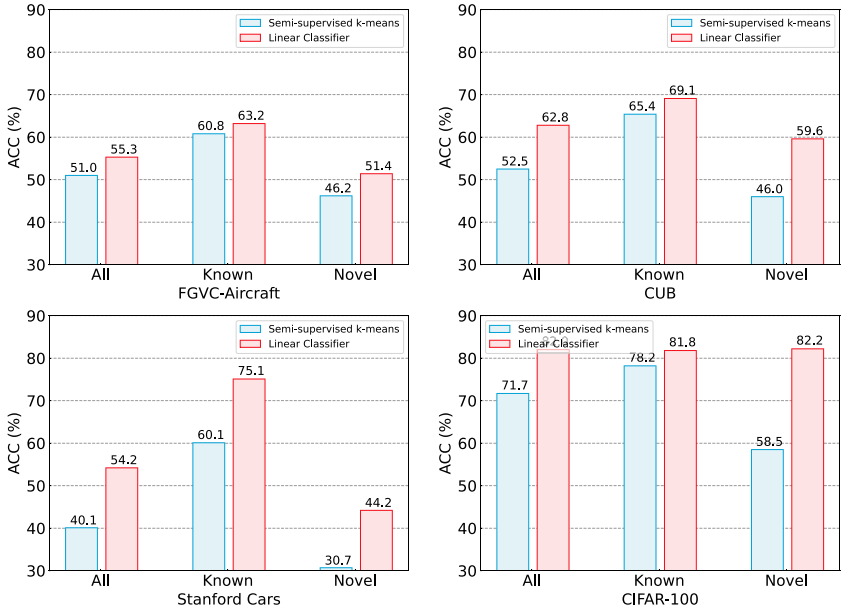


**Fig. 4.** The performance difference between SSK and linear classifier on four datasets.

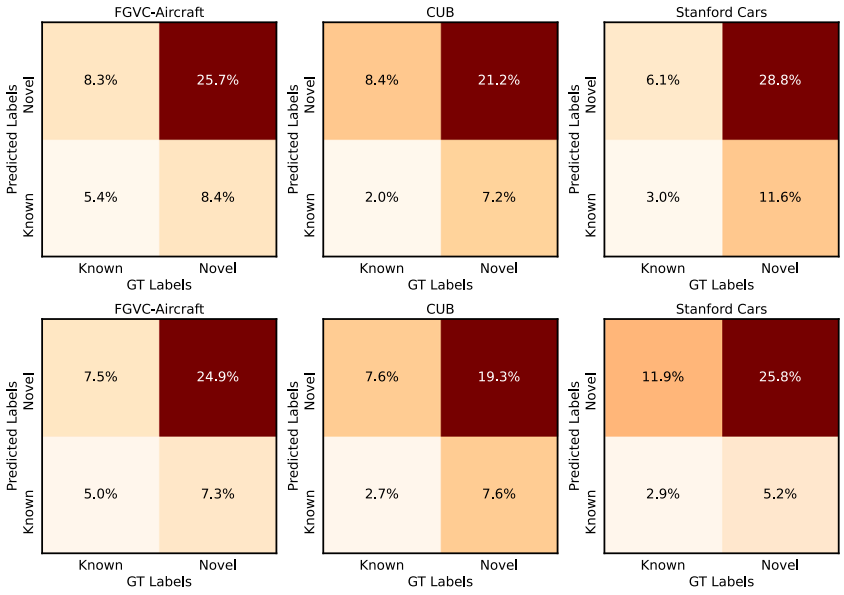


**Fig. 5.** Bias of prediction errors on three fine-grained datasets with different classifiers. The top row is baseline and the bottom row is PCR.

#### 4.5.2. The prediction error bias

Here, we will discuss how prediction consistency improves the performance in depth. Broadly speaking, prediction errors can be categorized into four types: Known-to-Known (K2K), Known-to-Novel (K2N), Novel-to-Known (N2K) and Novel-to-Novel (N2N). For instance, K2K refers to the mis-classification of samples from known classes into other known classes, while N2K represents the mis-classification of samples from novel classes into incorrect known classes. Fig. 5 illustrates the distribution of error types for $\mathcal{L}_{fea} + \mathcal{L}_{cls}$ (baseline) and PCR across the FGVC-Aircraft, CUB, and Stanford Cars datasets. It is evident that both the baseline and PCR demonstrate a higher concentration of errors when predicting novel classes. Moreover, PCR exhibits lower prediction errors compared to the baseline, with reductions of 0.8%, 1.9%, and 3.0% for N2N on the FGVC-Aircraft, CUB, and Stanford Cars datasets, respectively. The error rates for the other three error types are relatively comparable. It directly indicates the outstanding performance of PCR in predicting novel classes and further validates their ability to better combine the global information from SSK in representation learning.

#### 4.5.3. Effectiveness of prediction consistency

In this section, we will dive into the significance of maintaining prediction consistency. As we know, KL divergence is merely one method for quantifying the disparity between two distributions. In this case, we minimize the F-norm between $\mathbf{U}$ and $\mathbf{V}$ as a substitute for the KL divergence in Eq. (1). We set the coefficient of the F-norm to 1.0, and the results are consolidated in Table 4. It is evident that the model can achieve competitive results under two distinct constraint conditions. The utilization of F-norm yields superior results in predicting known classes, whereas KL divergence performs better in the discovery of novel classes. This validates the effectiveness of maintaining distribution consistency among different classifiers. One potential explanation is that by preserving the distribution consistency of predictions, the model can more effectively explore the underlying semantic information within the data and extract features conducive to clustering.

### 4.6. The whole performance during training stage

Fig. 6 illustrates the performance of our model throughout the training process on four benchmark datasets. It is evident that the model's performance stabilizes after 100 epochs. Notably, for the FGVC-Aircraft and Stanford Cars, a notable performance drop occurs around epochs 70–80, and finally obtain the optimal results. One possible reason is the introduction of PCR at the 60th epoch of training, which breaks the original feature distribution and enables the model to avoid local optima. This directly demonstrates the effectiveness of our proposed PCR. Conversely, we observe that the ACC for the known classes is consistently the highest across all datasets. In contrast, for the CIFAR-100, the ACC for the three distinct classes is almost the same at the end of training. We believe that this can be attributed to the large scale and substantial differences between the classes, which encourages the model to extract robust representations. Furthermore, from an experimental setup standpoint (refer to Table 1), out of the total 100 classes in CIFAR-100, 80 classes are known. This large amount of supervised information empowers the classifiers to make more accurate distinctions.

**Table 4**
The performance difference between different distribution measurements.

| | FGVC-Aircraft | | | CUB | | | Stanford Cars | | |
|---|---|---|---|---|---|---|---|---|---|
| | All | Known | Novel | All | Known | Novel | All | Known | Novel |
| F-norm | 54.8 | 63.4 | 50.5 | 62.5 | 70.5 | 58.6 | 51.0 | 70.3 | 41.7 |
| KL-divergence | 55.3 | 63.2 | 51.4 | 62.8 | 69.1 | 59.6 | 54.2 | 75.1 | 44.2 |

## 5. Conclusions

In this paper, we propose a novel GCD framework called PCR, which maintains the prediction consistency between linear classifier and SSK from explicit and implicit manners. PCR overcomes the limitation of SSK, which fails to capture fine-grained information within features, while also mitigating the risk of the linear classifier becoming trapped in local optima due to excessive supervised information. Finally, we utilize an EM framework to iteratively optimize the model, providing theoretical guarantees. Extensive experiments demonstrate that PCR consistently outperforms the baseline by substantial margins, establishing it as a state-of-the-art method. In the future, we will extend GCD to various tasks, including text prediction and image segmentation. Furthermore, we intend to integrate popular techniques like prompt learning and reinforcement learning into open-set learning.

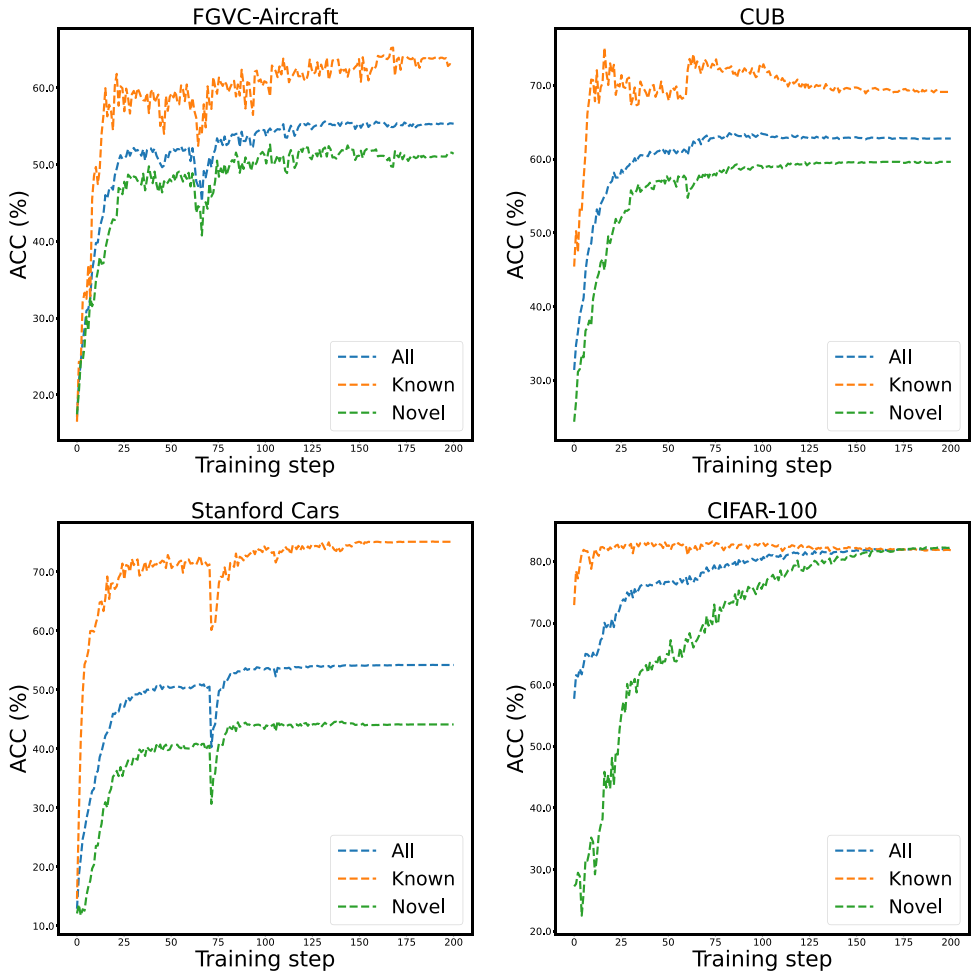


**Fig. 6.** The complete performance of PCR on four datasets.

## CRediT authorship contribution statement

**Yu Duan:** Writing – original draft, Methodology, Conceptualization. **Junzhi He:** Writing – review & editing, Software, Formal analysis, Data curation. **Runxin Zhang:** Visualization, Validation, Investigation, Formal analysis. **Rong Wang:** Supervision, Funding acquisition. **Xuelong Li:** Supervision. **Feiping Nie:** Writing – review & editing, Validation, Supervision, Formal analysis.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

The data is commonly used as benchmarks. The readers can be downloaded at their official website if they need.

## Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0101902, in part by the Natural Science Basic Research Program of Shaanxi (Program No. 2021JM-071), in part by the National Natural Science Foundation of China under Grant 62176212, Grant 61936014 and Grant 61772427, and in part by the Fundamental Research Funds for the Central Universities under Grant G2019KY0501.

## References

[1] W. An, F. Tian, P. Chen, Q. Zheng, W. Ding, New user intent discovery with robust pseudo label training and source domain joint training, IEEE Intell. Syst. 38 (4) (2023) 21–31, http://dx.doi.org/10.1109/MIS.2023.3283909.

[2] X. Song, Y. Mou, K. He, Y. Qiu, J. Zhao, P. Wang, W. Xu, Continual generalized intent discovery: Marching towards dynamic and open-world intent recognition, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023, Association for Computational Linguistics, 2023, pp. 4370–4382, URL https://aclanthology.org/2023.findings-emnlp.289.

[3] Z. Zhong, E. Fini, S. Roy, Z. Luo, E. Ricci, N. Sebe, Neighborhood contrastive learning for novel class discovery, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, Virtual, June 19-25, 2021, Computer Vision Foundation / IEEE, 2021, pp. 10867–10875, http://dx.doi.org/10.1109/CVPR46437.2021.01072, URL https://openaccess.thecvf.com/content/CVPR2021/html/Zhong_Neighborhood_Contrastive_Learning_for_Novel_Class_Discovery_CVPR_2021_paper.html.

[4] K. Han, S. Rebuffi, S. Ehrhardt, A. Vedaldi, A. Zisserman, AutoNovel: Automatically discovering and learning novel visual categories, IEEE Trans. Pattern Anal. Mach. Intell. 44 (10) (2022) 6767–6781, http://dx.doi.org/10.1109/TPAMI.2021.3091944.

[5] E. Fini, E. Sangineto, S. Lathuilière, Z. Zhong, M. Nabi, E. Ricci, A unified objective for novel class discovery, in: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, IEEE, 2021, pp. 9264–9272, http://dx.doi.org/10.1109/ICCV48922.2021.00915.

[6] Z. Zhong, L. Zhu, Z. Luo, S. Li, Y. Yang, N. Sebe, OpenMix: Reviving known knowledge for discovering novel visual categories in an open world, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, Virtual, June 19-25, 2021, Computer Vision Foundation / IEEE, 2021, pp. 9462–9470, http://dx.doi.org/10.1109/CVPR46437.2021.00934, URL https://openaccess.thecvf.com/content/CVPR2021/html/Zhong_OpenMix_Reviving_Known_Knowledge_for_Discovering_Novel_Visual_Categories_in_CVPR_2021_paper.html.

[7] H. Chi, F. Liu, W. Yang, L. Lan, T. Liu, B. Han, G. Niu, M. Zhou, M. Sugiyama, Meta discovery: Learning to discover novel classes given very limited data, in: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, 2022, URL https://openreview.net/forum?id=MEpKGLsY8f.

[8] S. Vaze, K. Han, A. Vedaldi, A. Zisserman, Generalized category discovery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7492–7501.

[9] S. Vaze, K. Han, A. Vedaldi, A. Zisserman, Generalized category discovery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7492–7501.

[10] B. Zhao, X. Wen, K. Han, Learning semi-supervised Gaussian mixture models for generalized category discovery, 2023, arXiv preprint arXiv:2305.06144.

[11] X. Yang, X. Pan, I. King, Z. Xu, Generalized category discovery with clustering assignment consistency, in: International Conference on Neural Information Processing, Springer, 2023, pp. 535–547.

[12] J. Otholt, C. Meinel, H. Yang, Guided cluster aggregation: A hierarchical approach to generalized category discovery, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 2618–2627.

[13] N. Pu, Z. Zhong, N. Sebe, Dynamic conceptional contrastive learning for generalized category discovery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7579–7588.

[14] R. Ouldnoughi, C.-W. Kuo, Z. Kira, CLIP-GCD: Simple language guided generalized category discovery, 2023, arXiv preprint arXiv:2305.10420.

[15] S. Zhang, S. Khan, Z. Shen, M. Naseer, G. Chen, F.S. 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, 2023, pp. 3479–3488.

[16] X. Wen, B. Zhao, X. Qi, Parametric classification for generalized category discovery: A baseline study, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 16590–16600.

[17] F. Chiaroni, J. Dolz, Z.I. Masud, A. Mitiche, I. Ben Ayed, Parametric information maximization for generalized category discovery, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 1729–1739.

[18] D. Berthelot, N. Carlini, E.D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, C. Raffel, ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring, in: ICLR, OpenReview.net, 2020.

[19] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. Raffel, E.D. Cubuk, A. Kurakin, C. Li, FixMatch: Simplifying semi-supervised learning with consistency and confidence, in: NeurIPS, 2020.

[20] Y. Wang, H. Chen, Q. Heng, W. Hou, Y. Fan, Z. Wu, J. Wang, M. Savvides, T. Shinozaki, B. Raj, B. Schiele, X. Xie, FreeMatch: Self-adaptive thresholding for semi-supervised learning, in: ICLR, OpenReview.net, 2023.

[21] Y. Xu, L. Shang, J. Ye, Q. Qian, Y. Li, B. Sun, H. Li, R. Jin, Dash: Semi-supervised learning with dynamic thresholding, in: ICML, in: Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 11525–11536.

[22] T. DeVries, G.W. Taylor, Improved regularization of convolutional neural networks with cutout, 2017, arXiv preprint arXiv:1708.04552.

[23] E.D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, Q.V. Le, AutoAugment: Learning augmentation strategies from data, in: CVPR, Computer Vision Foundation / IEEE, 2019, pp. 113–123.

[24] E.D. Cubuk, B. Zoph, J. Shlens, Q.V. Le, Randaugment: Practical automated data augmentation with a reduced search space, in: CVPR Workshops, Computer Vision Foundation / IEEE, 2020, pp. 3008–3017.

[25] Y. Wang, G. Huang, S. Song, X. Pan, Y. Xia, C. Wu, Regularizing deep networks with semantic data augmentation, IEEE Trans. Pattern Anal. Mach. Intell. 44 (7) (2022) 3733–3748.

[26] X. Li, Y. Jiang, C. Li, Y. Wang, J. Ouyang, Learning with partial labels from semi-supervised perspective, in: AAAI, AAAI Press, 2023, pp. 8666–8674.

[27] K. Han, A. Vedaldi, A. Zisserman, Learning to discover novel visual categories via deep transfer clustering, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 8401–8409.

[28] Y.-C. Hsu, Z. Lv, Z. Kira, Learning to cluster in order to transfer across domains and tasks, 2017, arXiv preprint arXiv:1711.10125.

[29] Y.-C. Hsu, Z. Lv, J. Schlosser, P. Odom, Z. Kira, Multi-class classification without multi-class labels, 2019, arXiv preprint arXiv:1901.00544.

[30] E. Fini, E. Sangineto, S. Lathuiliere, Z. Zhong, M. Nabi, E. Ricci, A unified objective for novel class discovery, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9284–9292.

[31] K. Han, S.-A. Rebuffi, S. Ehrhardt, A. Vedaldi, A. Zisserman, Autonovel: Automatically discovering and learning novel visual categories, IEEE Trans. Pattern Anal. Mach. Intell. 44 (10) (2021) 6767–6781.

[32] B. Zhao, K. Han, Novel visual category discovery with dual ranking statistics and mutual knowledge distillation, Adv. Neural Inf. Process. Syst. 34 (2021) 22982–22994.

[33] Z. Zhong, E. Fini, S. Roy, Z. Luo, E. Ricci, N. Sebe, Neighborhood contrastive learning for novel class discovery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10867–10875.

[34] M. Rosvall, C.T. Bergstrom, Maps of random walks on complex networks reveal community structure, Proc. Natl. Acad. Sci. 105 (4) (2008) 1118–1123.

[35] A. Banerjee, L.S. Kallooriyakath, S. Biswas, AMEND: Adaptive margin and expanded neighborhood for efficient generalized category discovery, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 2101–2110.

[36] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, S.-N. Lim, Visual prompt tuning, in: European Conference on Computer Vision, Springer, 2022, pp. 709–727.

[37] W. Pan, J. Yan, H. Chen, J. Yang, Z. Xu, X. Li, J. Yao, Human-machine interactive tissue prototype learning for label-efficient histopathology image segmentation, in: IPMI, in: Lecture Notes in Computer Science, vol. 13939, Springer, 2023, pp. 679–691.

[38] J. Yan, H. Chen, X. Li, J. Yao, Deep contrastive learning based tissue clustering for annotation-free histopathology image analysis, Comput. Med. Imaging Graph. 97 (2022) 102053.

[39] Y. Li, M. Yang, D. Peng, T. Li, J. Huang, X. Peng, Twin contrastive learning for online clustering, Int. J. Comput. Vis. 130 (9) (2022) 2205–2221.

[40] Z. Dang, C. Deng, X. Yang, K. Wei, H. Huang, Nearest neighbor matching for deep clustering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13693–13702.

[41] S. Kim, D. Kim, M. Cho, S. Kwak, Proxy anchor loss for deep metric learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3238–3247.

[42] J. Peng, C. Desrosiers, M. Pedersoli, Diversified multi-prototype representation for semi-supervised segmentation, 2021, arXiv preprint arXiv:2111.08651.

[43] H.W. Kuhn, The Hungarian method for the assignment problem, Nav. Res. Logist. Q. 2 (1-2) (1955) 83–97.

[44] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. Rabbat, N. Ballas, Masked siamese networks for label-efficient learning, in: European Conference on Computer Vision, Springer, 2022, pp. 456–473.

[45] D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, in: SODA, SIAM, 2007, pp. 1027–1035.

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

[47] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, P. Perona, Caltech-UCSD Birds 200, California Institute of Technology, 2010.

[48] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3D object representations for fine-grained categorization, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 554–561.

[49] A. Krizhevsky, G. Hinton, et al., Learning Multiple Layers of Features from Tiny Images, Toronto, ON, Canada, 2009.

[50] Y. Li, P. Hu, Z. Liu, D. Peng, J.T. Zhou, X. Peng, Contrastive clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 10, 2021, pp. 8547–8555.

[51] K. Cao, M. Brbic, J. Leskovec, Open-world semi-supervised learning, in: International Conference on Learning Representations, 2021.