Learning Semi-supervised Gaussian Mixture Models for Generalized Category Discovery

Bingchen Zhao[†] Xin Wen[‡] Kai Han[‡*]
[†]University of Edinburgh [‡]The University of Hong Kong
bingchen.zhao@ed.ac.uk wenxin@eee.hku.hk kaihanx@hku.hk

Abstract

In this paper, we address the problem of generalized category discovery (GCD), i.e.*, given a set of images where part of them are labelled and the rest are not, the task is to automatically cluster the images in the unlabelled data, leveraging the information from the labelled data, while the unlabelled data contain images from the labelled classes and also new ones. GCD is similar to semi-supervised learning (SSL) but is more realistic and challenging, as SSL assumes all the unlabelled images are from the same classes as the labelled ones. We also do not assume the class number in the unlabelled data is known a-priori, making the GCD problem even harder. To tackle the problem of GCD without knowing the class number, we propose an EM-like framework that alternates between representation learning and class number estimation. We propose a semi-supervised variant of the Gaussian Mixture Model (GMM) with a stochastic splitting and merging mechanism to dynamically determine the prototypes by examining the cluster compactness and separability. With these prototypes, we leverage prototypical contrastive learning for representation learning on the partially labelled data subject to the constraints imposed by the labelled data. Our framework alternates between these two steps until convergence. The cluster assignment for an unlabelled instance can then be retrieved by identifying its nearest prototype. We comprehensively evaluate our framework on both generic image classification datasets and challenging fine-grained object recognition datasets, achieving state-of-the-art performance. Our code is available at* https://github.com/DTennant/GPC.

1. Introduction

The success of deep learning is driven by the availability of large-scale data with human annotations. Given enough annotated data, deep learning models are able to surpass human-level performance on many important computer vision tasks such as image classification [21]. But the cost of collecting a large annotated dataset is not always affordable, and it is also not possible to annotate all new classes emerging from the real world. Thus, designing models that can learn to deal with large-scale unlabelled data in the open world is of great value and importance. Semi-supervised learning (SSL) [38] is proposed as a solution to learn a model on both labelled data and unlabelled data, with many works achieving promising performance [1, 46, 44]. However, SSL assumes that labelled instances are provided for all object classes in the unlabelled data. The novel category discovery (NCD) task is introduced [17, 16] to automatically discover novel classes by transferring the knowledge learned from the labelled instances of known classes, assuming the unlabelled data only contain instances from new classes. Generalized category discovery (GCD) [48] further relaxes the assumption in NCD, and tackles a more

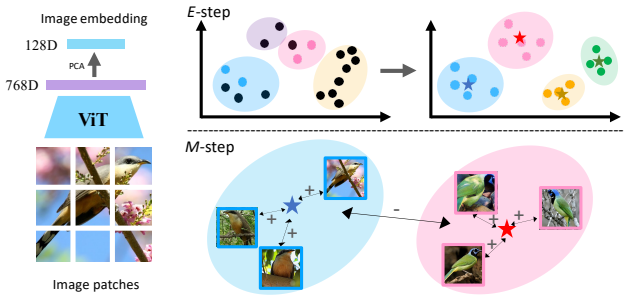


Figure 1: **Overview of our proposed EM-like framework.** The input images are fed into a ViT-B model to obtain a 768-dimensional feature vector, then the feature vector will be projected to a lower dimensional space using the projection calculated from PCA. We perform class number estimation and representation learning in this projected space. In the E-step, we use a semi-supervised GMM that can split separable clusters and merge cluttered clusters to estimate the class number and prototypes, which will be used in the M-step of representation learning with prototypical contrastive learning.

*Corresponding author.

generalized setting where the unlabelled data contains instances from both known and novel categories. Existing methods for NCD [17, 16, 58, 60, 61, 12, 26] and GCD [48, 11, 51, 45, 56, 59, 34] learn the representation and cluster assignment assuming the class number is known a priori [58, 26, 60, 12, 61] or precomputed [17, 48]. In practice, the number of categories in the unlabelled data is often unknown, while precomputing the class number without taking the representation learning into consideration is likely to lead to a sub-optimal solution.

In this paper, we argue that representation learning and the estimation of class numbers should be considered together and could reinforce each other, *i.e.*, a strong representation could help a more accurate estimation of the class numbers, and an accurate class number could help learn a better feature representation. To this end, we propose a unified EM-like framework that alternates between feature representation learning and class number estimation where the E-step is aimed at automatically estimating a proper class number and a set of class prototypes in the unlabelled data and the M-step is aimed at learning better representation with the class number and class prototypes estimated. In particular, we propose using a prototype contrastive representation learning [33] method for GCD, which requires a set of prototypes to serve as anchors for representation learning. Prototypical contrastive learning [33] is developed for unsupervised representation learning to generalize to different tasks, where the prototypes are obtained by over-clustering the dataset with one or multiple given prototype numbers, using non-parametric clustering algorithms like k-means. Instead, to handle the problem of GCD, we propose to estimate the prototype number and prototypes automatically and simultaneously. To do so, we introduce a semi-supervised variant of the Gaussian Mixture Model (GMM) with a stochastic splitting and merging mechanism to determine the most suitable clusters based on current representation. These clusters can then be used to form prototypes to facilitate contrastive representation learning. Our framework alternates between the E- and M-step until converging to achieve robust representation and reliable category estimation. After learning, the cluster assignment for an unlabelled instance, either from known or novel classes, can be retrieved by finding the nearest prototypes. Thus we name our framework as **GPC**: **G**aussian mixture model for generalized category discovery with **P**rotypical **C**ontrastive learning.

Our contributions in this paper are as follows: (1) We demonstrate that in generalized category discovery, the class number estimation and representation learning can reinforce each other in the learning process. Strong representations can give a better estimation of the class number, and vice versa. (2) We propose an EM-like framework that alternates between prototype estimation with a variant of GMM (E-step) and representation learning based on prototypical contrastive learning (M-step). (3) We introduce a semi-supervised variant of GMM with a stochastic splitting and merging mechanism to allow dynamic change of the prototypes by examining the cluster compactness and separability based on the Metropolis-Hastings ratio [19]. (4) We comprehensively evaluated our framework on both the generic image classification benchmark, including CIFAR10, CIFAR100, ImageNet-100, and the challenging fine-grained Semantic Shifts Benchmark suite, which includes CUB-200, Stanford-Cars, and FGVC-aircrafts, achieving the state-of-the-art results.

2. Related work

Novel category discovery (NCD) is first formalized in DTC [17], where the task is to discover new categories leveraging the knowledge of a set of labelled categories. Many methods have been proposed to tackle this challenge. To name a few, RankStat [15, 16] and DualRank [58] transfer knowledge from labelled to unlabelled classes using feature ranking statistics. OpenMix [61] shows that mixing labelled and unlabelled data can prevent overfitting. NCL [60] and WTA [26] use contrastive learning for NCD. UNO [12] introduces a unified cross-entropy loss for joint training on labelled and unlabelled data. [29] introduces a spacing loss for representation learning in NCD. NCDwF [28] studies the NCD problem under an incremental learning setting. Earlier methods like [22, 23] for generalized transfer learning can also be applied to NCD. Generalized category discovery (GCD) [48] extends NCD to a more open-world setting where unlabelled instances can come from both labelled and unlabelled categories. ORCA [2] also tackles GCD from a semi-supervised learning perspective. Several improved methods have been proposed for generalized category discovery (GCD), such as [11, 51, 39, 56, 45], which have shown promising results. Incremental GCD is considered in [57, 59]. Concurrently, CiPR [18] presents a hierarchical GCD approach employing selective neighbor clustering, leading to improved efficiency and performance without necessitating knowledge of the class number. [14] studies semantic category discovery by leveraging vision-language representations, thereby enabling GCD with class name prediction. However, most existing GCD methods assume that the novel class number is known *a priori*, which is often not the case in the real world. To address this limitation, [17, 48] precompute the number of novel classes with a semi-supervised k-means algorithm from pretrained representations. In this paper, we demonstrate that class number estimation and representation learning can be jointly considered to mutually benefit each other.

Contrastive learning [4, 5, 20, 7, 62] (CL) has been shown very effective for representation learning in a self-supervised manner, using the instance discrimination pretext [52] as the learning objective. The instance discrimination task learns

a representation by pulling positive samples from the augmentations of the same images closer and pushing negative samples from different images apart in the embedding space. Instead of contrasting over all instances in a mini-bath, prototypical contrastive learning (PCL) [33] proposes to contrast the features with a set of prototypes which can provide a higher level abstraction of dataset than instances and has been shown to be more data efficient without the need of large batch size. Though PCL is developed for unsupervised representation learning, if the prototypes are viewed as cluster centers, it can be leveraged in the partially supervised setting of GCD for representation learning to better fit the GCD task of partitioning data into different clusters. Thus, in this paper, we adopt PCL to fit the GCD setting for representation learning in which the downstream clustering task is directly considered.

Semi-supervised learning (SSL) has been a long standing research topic which many effective method proposed [40, 44, 1, 32, 46]. In SSL, the labelled and the unlabelled data are assumed to come from the same set of classes, and the task is to learn a classification model that can take advantage of both labelled and unlabelled data. Consistency-based methods are among the most effective methods for SSL, such as Mean-teacher [46], MixMatch [1], and FixMatch [44]. Self-supervised representation learning also shows to be helpful for SSL because it can provide a strong representation [55, 40]. Recent works extend semi-supervised learning by relaxing the assumption of exactly the same classes in the labeled and unlabelled data [43, 25, 54], but their focus is improving the performance of the labeled categories without discovering novel categories in the unlabeled set.

Unsupervised clustering has been studied for decades, and there are many existing classical approaches [35, 10, 6] as well as deep learning based approaches [53, 41, 13]. Recently, DeepDPM [42] is proposed to automatically determine the number of clusters for a given dataset by adopting a similar split/merge framework that changes the inferred number of clusters. However, due to the unsupervised nature of these methods, there is no prior or supervision over how a cluster should be formed, thus multiple equally valid clustering results following different clustering criteria can be produced. Thus, directly applying unsupervised clustering methods to the task of generalized category discovery is not feasible, as we would want the model to use one unique clustering criteria implicitly given by the labelled data.

3. Method

Given a collection of partially labelled data, $\mathcal{D} = \mathcal{D}^l \cup \mathcal{D}^u$, where $\mathcal{D}^l = \{(x_i, y_i^l)\} \in \mathcal{X} \times \mathcal{Y}_l$ is labelled, $\mathcal{D}^u = \{x_i, y_i^u\} \in \mathcal{X} \times \mathcal{Y}_u$ is unlabelled, and $\mathcal{Y}_l \subset \mathcal{Y}_u$, Generalized category discovery (GCD) aims at automatically assign labels for the unlabelled instances in $\mathcal{D}^u$, by transferring knowledge acquired from $\mathcal{D}^l$. Let the category number in $\mathcal{D}^l$ be $K^l = |\mathcal{Y}_l|$ and that in $\mathcal{D}^u$ be $K^u = |\mathcal{Y}_u|$. The number of new categories $\mathcal{D}^u$ is then $K^n = |\mathcal{Y}_u \setminus \mathcal{Y}_l| = K^u - K^l$. Though K^l can be accessed from the labelled data, we do not assume K^n or K^u to be known. This is a realistic setting to reflect the real open world, where we often have access to some labelled data, but in the unlabelled data, we also have instances from unseen new categories.

The key challenges for GCD are representation learning, category number estimation, and label assignment. Existing methods for NCD and GCD [17, 48] deal with these three challenges independently. However, we believe they are inherently linked with each other. Label assignment depends on representation and category number estimation. A good class number estimation can facilitate representation learning, thus better label assignment, and vice versa. Thus, in this paper, we aim to jointly handle these challenges in the learning process for a more reliable GCD.

To this end, we propose a unified EM-like framework that alternates between representation learning and class number estimation, while the label assignment turns out to be a by-product during class number estimation. In the E-step, we introduce a semi-supervised variant of the Gaussian Mixture Model (GMM) to estimate the class numbers by dynamically splitting separable clusters and merging cluttered clusters based on current representation, forming a set of class prototypes for both seen and unseen classes, and in the M-step, we train the model to produce discriminative representation by prototypical contrastive learning using the cluster centers from the GMM prototypes derived from the E-step during class number estimation. After training, the class assignment for each instance can be retrieved by simply identifying the nearest prototype.

3.1. Representation learning

The goal of representation learning is to learn a discriminative representation that can well separate different categories, not only the old ones, but also the new ones. Contrastive learning (CL) has been shown to be an effective choice for NCD [26] and GCD [48]. Self-supervised contrastive learning is defined as

$$\mathcal{L}_{CL} = -\log \frac{\exp(z_i \cdot z_i'/\tau)}{\sum_{j=1}^{n} \exp(z_i \cdot z_j'/\tau)} \tag{1}$$

where z_i and z_i' are the representations of two views obtained from the same image using random augmentations and τ is the temperature. Two views of the same instance are pulled closer, and different instances are pushed away during training. Self-supervised contrastive learning and its supervised variant, in which different instances from the same category are also pulled closer, are used in [48] for representation learning. However, as a stronger training signal is used for the labelled data, the representation is likely biased to the labelled data to some extent. Moreover, such a method does

not take the downstream clustering task into account during learning, thus a clustering algorithm is required to run independently after the representation learning.

In this paper, we adopt prototypical contrastive learning (PCL) [33] to the GCD setting to learn the representation $z_i = f(x_i) \in \mathbb{R}^d$. PCL uses a set of prototypes $\mathcal{C} = \{\mu_1, \ldots, \mu_K\}$ to represent the dataset for contrastive learning instead of the random augmentation generated views z_i'. PCL loss can be written as

$$\mathcal{L}_{PCL} = -\log \frac{\exp(z_i \cdot \mu_s/\tau)}{\sum_{j=1}^{K} \exp(z_i \cdot \mu_j/\tau)} \tag{2}$$

where μ_s is the corresponding prototype for z_i. It was originally designed as an alternative for self-supervised contrastive learning by over-clustering the training data to obtain the prototypes during training. We employ PCL here to learn reliable representation while taking the downstream clustering into account for GCD, where we have a set of partially labelled data. In our case, the prototypes can be interpreted as the class centers for each of the categories. To obtain the prototypes for the seen categories, we directly calculate the class mean by averaging all the feature vectors of the labelled instances. For the unseen categories, we obtain the prototypes with a semi-supervised variant of the Gaussian Mixture Model (GMM), as will be introduced in Sec. 3.2. This way, the cluster assignment for an unlabelled image can be readily achieved by finding the nearest prototype.

Additionally, we observe that only a few principal dimensions can already recover most of the variances in the representation space of z_i, which is known as *dimensional collapse* (DC) in [27, 24], and it is shown that DC can be caused by strong augmentations or implicit regularizations in the model, and preventing DC during training can lead to a better feature representation. To alleviate DC for representation learning in our case, we propose to first project the feature to a subspace obtained by principal component analysis (PCA) before the contrastive learning. Specifically, we apply PCA on a matrix Z formed by a mini-batch of features z_i, with a batch size of n, the feature dimension d, and the number of effective principal directions q. We have $Z \approx Udiag(S)V^\top$, where $U \in \mathbb{R}^{n\times q}$, $S \in \mathbb{R}^q$ and $V \in \mathbb{R}^{d\times q}$. We can then project features z_i to principal directions to obtain a more compact feature $v_i = V \cdot z_i$, and replace feature z_i with v_i in Eq. (2) for PCL. The prototypes are also computed in the projected space.

We jointly use self-supervised contrastive learning and PCL to train our model. The overall learning objective can be written as

$$\mathcal{L} = \mathcal{L}_{CL} + \lambda(t)\mathcal{L}_{PCL} \tag{3}$$

where $\lambda(t)$ is a linear warmup function defined as $\lambda(t) = \min(1, \frac{t}{T})$ where t is the current epoch and T warmup length

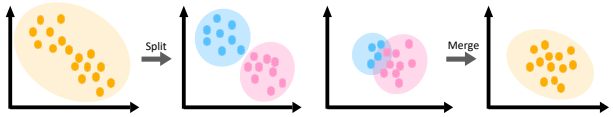


Figure 2: **Examples for splitting a separable cluster and merging two cluttered clusters.** Left: the cluster is split because the two sub-component in this cluster are easily separable. Right: two clusters are merged as they are cluttered and likely from the same class.

($T = 20$ in our experiments). The reason we use both CL and PCL is that, in the beginning, the representation is not well suited for clustering, and thus the obtained prototypes are not informative to facilitate the representation learning. Hence, we gradually increase the weight of PCL during training from 0 to 1 in the first T epochs.

3.2. Class number and prototypes estimation with semi-supervised Gaussian mixture model

In this section, we present a semi-supervised variant of the Gaussian mixture model (GMM) with each Gaussian component consisting of two sub-components to estimate the prototypes for representation learning in Sec. 3.1 and the unknown class number. GMM estimates the prototypes and assigns a label for each data point by finding its nearest prototype. The cluster label assignment and the prototypes are then used for prototypical contrastive learning. The GMM is defined as

$$p(z) = \sum_{i=1}^{K} \pi_i \mathcal{N}(z|\mu_i, \Sigma_i), \tag{4}$$

where $\mathcal{N}(z|\mu_i, \Sigma_i)$ is the Gaussian probability density function with mean $\mu_i \in \mathbb{R}^d$ and covariance $\Sigma_i \in \mathbb{R}^{d\times d}$, and π_i is the weight for i-th Gaussian component and we have $\sum_{i=1}^{N} \pi_i = 1$. Ideally, we would expect the component number K in the GMM to be equal to the class number K^u in $\mathcal{D}$. To estimate the unknown class number K^u, we leverage an automatic *splitting-and-merging* strategy into the modeling process to obtain an optimal K, which is expected to be as close to K^u as possible. We alternate between representation learning and K^u estimation until convergence to get discriminative representation learning and a reliable class number estimation. For initialization, K can be set to any number greater than K^l. In our experiments, we simply set the initial number of components to a default $K_{init} = K^l + \frac{K^l}{2}$. We run a semi-supervised k-means algorithm [48] with $k = K$ to obtain the μ and Σ for each component in the mixture model. Note that the semi-supervised k-means algorithm is constrained to the labelled data in a way that labelled instances from the same class are assigned to the same cluster, and labelled instances from different classes will not be

Algorithm 1: The overall algorithm of our method.

Input:
$\mathcal{D}$, $\mathcal{D}^l$, and $\mathcal{D}^u$ The datasets.
K_{init} Initial guess of K.

$K \leftarrow K_{init}$
for $e = 1$ **to** E **do**
  $z \leftarrow f(x), x \in \mathcal{D}$ `// extract features`
  $\mu, \Sigma \leftarrow \arg\max \sum_{i=1}^{K} \pi_i \mathcal{N}(z|\mu_i, \Sigma_i)$ `// estimate prototypes using GMM`
  for $i = 1$ **to** $len(\mathcal{D})$ **do**
    $\mathcal{B}^l \leftarrow \{x_i^l \sim \mathcal{D}^l\}_{i=1}^{N^l}$ `// sample a batch of` N^l `labelled images`
    $\mathcal{B}^u \leftarrow \{x_i^u \sim \mathcal{D}^u\}_{i=1}^{N^u}$ `// sample a batch of` N^u `unlabelled images`
    $f \leftarrow \arg\min \mathcal{L}(f, \mu, \mathcal{B}^l, \mathcal{B}^u)$ `// prototypical contrastive learning`
  end
  $H_s, H_m \leftarrow$ calc_prob(μ, Σ) `// probability for split and merge`
  $\mu, \Sigma \leftarrow$ perform_op(H_s, H_m) `// perform operations`
  $K \leftarrow len(\mu)$ `// update` K
end

Output: feature extractor $f(\cdot)$, cluster centers μ_i

assigned to the same cluster. To facilitate the splitting and merging process, for each Gaussian component defined by μ_i and Σ_i, we further depict it with a GMM with two sub-components $\mu_{i,1}, \mu_{i,2}$ and $\Sigma_{i,1}, \Sigma_{i,2}$ with $\pi_{i,1} + \pi_{i,2} = 1$. We run a k-means with $k = 2$ on the i-th component to obtain $\mu_{i,1}, \mu_{i,2}$ and $\Sigma_{i,1}, \Sigma_{i,2}$.

For a cluster whose two sub-components are roughly independent and equally sized (*e.g.*, left part of Fig. 2), *i.e.*, they are easily separable, we would like the model to split it into two such that the model can better fit the data distribution and the class assignment will be more accurate because it is less likely that such distinct clusters will belong to the same class. For two clusters that are cluttered with each other (*e.g.*, right part of Fig. 2), *i.e.*, difficult to distinguish, we would like to merge them into one, so that they will be considered as from the same class. Following this intuition, we use the Metropolis-Hastings framework [19] to compute a probability $p_s = \min(1, H_s)$ to stochastically split a cluster into two. The Hastings ratio is defined as

$$H_s = \frac{\Gamma(N_{i,1})h(\mathcal{Z}_{i,1};\theta)\Gamma(N_{i,2})h(\mathcal{Z}_{i,2};\theta)}{\Gamma(N_i)h(\mathcal{Z}_i;\theta)}, \tag{5}$$

where Γ is the factorial function, *i.e.*, $\Gamma(n) = n! = n \times (n-1) \times \cdots \times 1$, $\mathcal{Z}_i$ is the set of data points in cluster i, $\mathcal{Z}_{i,j}$ is the set of data points in the j-th sub-cluster of cluster i, $N_i = |\mathcal{Z}_i|$, $N_{i,j} = |\mathcal{Z}_{i,j}|$, $h(Z;\theta)$ is the marginal likelihood of the observed data $\mathcal{Z}$ by integrating out the μ and Σ parameters in the Gaussian, and θ is the prior distribution of μ and Σ. More details can be found in the supplementary. The intuition behind this H_s is that, if the number of data points in two sub-components is roughly balanced, which is measured by the $\Gamma(\cdot)$ terms, and the data points in the two sub-components are independent of each other, which is measure by the $h(\cdot;\theta)$ terms, there should be a greater chance of splitting the cluster. After performing a split operation, the μ_i and Σ_i of previous components i will be replaced with $\mu_{i,1}, \mu_{i,2}$ and $\Sigma_{i,1}, \Sigma_{i,2}$ of two sub-components. We will then run two k-means within the two newly formed components to obtain their corresponding sub-components. On the contrary, if two clusters are cluttered with each other, they should be merged. Similar to splitting, we determine the merging probability by $p_m = \min(1, H_m)$, where H_m is calculated similarly for two clusters i and j:

$$H_m = \frac{\Gamma(N_i + N_j)h(\mathcal{Z}_i \cup \mathcal{Z}_j;\theta)}{\Gamma(N_i)h(\mathcal{Z}_i;\theta)\Gamma(N_j)h(\mathcal{Z}_j;\theta)}. \tag{6}$$

Note that both H_s and H_m are within the range of $(0, +\infty)$, so we use $p_s = min(1, H_s)$ and $p_m = min(1, H_m)$ to convert it into a valid probability.

To take the labelled instances into consideration during the splitting-and-merging process, if a cluster consists of labelled instances, we set its $p_s = 0$; if for any two clusters containing instances from two labelled classes, we set their $p_m = 0$.

During the splitting-and-merging process, we first apply splitting according to the p_s and then apply merging according to p_m. The newly formed clusters by splitting will not be reused during the merging step. After finishing the splitting and merging, we can obtain the prototypes, and thus can estimate K, for our PCL-based representation learning. We alternate between representation learning and class number estimation for each training epoch until converge. The final K will be considered the estimated class number in $\mathcal{D}$. The cluster assignment for each unlabelled instance can be easily retrieved by identifying its nearest prototype, without the need of running a non-parametric clustering algorithm as [48]. The overall training process is summarized in Algorithm 1.

4. Experiments

4.1. Experimental setup

Benchmark and evaluation metric. We validate the effectiveness of our method on the generic image classification

Table 1: **Results on generic image classification datasets.**

No.	Methods	Known K	PCA	CIFAR10			CIFAR100			ImageNet-100		
				All	Old	New	All	Old	New	All	Old	New
(1)	k-means [35]	✓	✗	83.6	85.7	82.5	52.0	52.2	50.8	72.7	75.5	71.3
(2)	RankStats+ [16]	✓	✗	46.8	19.2	60.5	58.2	77.6	19.3	37.1	61.6	24.8
(3)	UNO+ [12]	✓	✗	68.6	98.3	53.8	69.5	80.6	47.2	70.3	95.0	57.9
(4)	ORCA [2]	✓	✗	81.8	86.2	79.6	69.0	77.4	52.0	73.5	92.6	63.9
(5)	Vaze *et al.* [48]	✓	✗	91.5	97.9	88.2	73.0	76.2	66.5	74.1	89.8	66.3
(6)	Ours (GPC)	✓	✗	92.0	98.3	88.7	77.4	84.8	62.4	76.5	94.0	68.5
(7)	Ours (GPC)	✓	✓	92.2	98.2	89.1	77.9	85.0	63.0	76.9	94.3	71.0
(8)	Vaze *et al.* [48]	✗	✗	88.6	96.2	84.9	73.2	83.5	57.9	72.7	91.8	63.8
(9)	Vaze *et al.* [48]	✗	✓	89.7	97.3	86.3	74.8	83.8	58.7	73.8	92.1	64.6
(10)	Ours (GPC)	✗	✗	88.2	97.0	85.8	74.9	84.3	59.6	74.7	92.9	65.1
(11)	Ours (GPC)	✗	✓	**90.6**	**97.6**	**87.0**	**75.4**	**84.6**	**60.1**	**75.3**	**93.4**	**66.7**

benchmark (including CIFAR-10/100 [31] and ImageNet-100 [47]) and also the recently proposed Semantic Shift Benchmark [49] (SSB) (including CUB-200 [50], Stanford Cars [30], and FGVC-Aircraft [36]). For each of the datasets, we follow [48] and sample a subset of all classes for which we have annotated labels during training. For experiments on SSB datasets, we directly use the class split from [49]. 50% of the images from these labelled classes will be used as the labelled instances in $\mathcal{D}^l$, and the remaining images are regarded as the unlabelled data $\mathcal{D}^u$ containing instances from labelled and unlabelled classes. See Tab. 2 for statistics of the datasets we evaluated. We evaluate model performance with clustering accuracy (ACC) following standard practice in the literature. At test-time, given ground truth labels y^* and model predicted cluster assignments $\hat{y}$, the ACC is calculated as $ACC = \frac{1}{M}\sum_{i=1}^{M} \mathbb{1}(y_i^* = g(\hat{y}_i))$ where g is the optimal permutation for matching predicted cluster assignment $\hat{y}$ to actual class label y_i^* and $M = |\mathcal{D}^u|$.

Table 2: **Data splits in the experiments.**

	labelled	unlabelled
CIFAR-10	5	5
CIFAR-100	80	20
ImageNet-100	50	50
CUB-200	100	100
Stanford-Cars	98	98
FGVC-aircraft	50	50

Implementation details. We train and test all the methods with a ViT-B/16 backbone [9] with pretrained weights from DINO [3]. We use the output of `[CLS]` token with a dimension of 768 as the feature representation for an input image. We only finetune the last block of the ViT-B backbone to prevent the model from overfitting to the labelled classes during training. We set the batch size for training the model to 128 with 64 labelled images and 64 unlabelled images and use a cosine annealing schedule for the learning rate starting from 0.1. The number of principal directions in the PCA is set to 128, which we found performs the best across all the datasets evaluated. We train all the methods for 200 epochs on each dataset for a fair comparison with previous works, and the best-performing model is selected using the accuracy on the validation set of the labelled classes. All experiments are done with an NVIDIA V100 GPU with 32GB memory.

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

In Tab. 1, we report the comparison with the state-of-the-art method of [48], strong baselines derived from NCD methods, and the k-means on the generic classification datasets. Notably, our method consistently achieves the best overall performance on all datasets, under the challenging setting where the class number is unknown. When the class number is known, our method also achieves the best performance on all datasets, except ImageNet-100, on which the best performance is achieved by ORCA [2]. In rows 1-7, we compare with other methods with the known class number in the unlabelled data, while in rows 8-11 we compare with [48] for the case of the unknown class number. We can see that our proposed framework outperforms other methods in most cases and especially when the number of classes is unknown. Comparing rows 10 and 11 to row 5, we can see that our proposed method without knowing the number of classes can even match the performance of previous strong baseline with the number of classes known to the model. Furthermore, from row 6 vs row 7 and row 10 vs row 11, we can see that the additional PCA layer can effectively improve the performance, also the performance

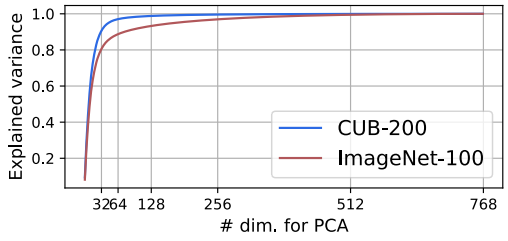
Table 3: **Results on Semantic Shift Benchmark datasets.**

No.	Methods	Known K	PCA	CUB			Stanford Cars			FGVC-aircraft		
				All	Old	New	All	Old	New	All	Old	New
(1)	k-means [35]	✓	✗	34.3	38.9	32.1	12.8	10.6	13.8	16.0	14.4	16.8
(2)	RankStats+ [16]	✓	✗	33.3	51.6	24.2	28.3	61.8	12.1	26.9	36.4	22.2
(3)	UNO+ [12]	✓	✗	35.1	49.0	28.1	35.5	70.5	18.6	40.3	56.4	32.2
(4)	ORCA [2]	✓	✗	35.3	45.6	30.2	23.5	50.1	10.7	22.0	31.8	17.1
(5)	Vaze *et al.* [48]	✓	✗	51.3	56.6	48.7	39.0	57.6	29.9	45.0	41.1	46.9
(6)	Ours (GPC)	✓	✗	54.2	54.9	50.3	41.2	58.8	31.6	46.1	42.4	47.2
(7)	Ours (GPC)	✓	✓	55.4	58.2	53.1	42.8	59.2	32.8	46.3	42.5	47.9
(8)	Vaze *et al.* [48]	✗	✗	47.1	55.1	44.8	35.0	56.0	24.8	40.1	40.8	42.8
(9)	Vaze *et al.* [48]	✗	✓	49.2	**56.2**	46.3	36.3	56.6	25.9	43.2	**40.9**	44.6
(10)	Ours (GPC)	✗	✗	50.2	52.8	45.6	36.7	56.3	26.3	39.7	39.6	42.7
(11)	Ours (GPC)	✗	✓	**52.0**	55.5	**47.5**	**38.2**	**58.9**	**27.4**	**43.3**	40.7	**44.8**

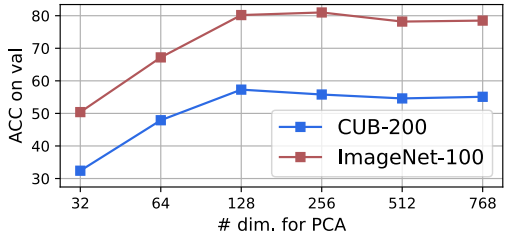
improvement from PCA is larger on the 'New' classes than on the 'Old' classes, which validates that the PCA can keep the representation space from collapsing and improve the performance on classes without using any labels. Due to the fact that labelled instances provide a stronger training signal, we can see from rows 6 - 11 that performance on 'Old' classes is generally steady. Comparing row 7 to row 5 and row 11 to row 8, we can see our full method outperforms the previous state-of-the-art method Vaze *et al.* [48] by large margins on both known and unknown class number cases. Tab. 3 shows performance comparison on the more challenging fine-grained Semantic Shift Benchmark [49]. A similar trend of Tab. 1 holds true for the results on SSB. Our approach achieves competitive performance in all cases and again reaches a better performance when the number of classes is unknown. In Tab. 4, we present the results on the Herbarium-19 dataset which is a long-tailed dataset, adding additional challenges for the GCD task. Again, our method performs the best on 'All' and 'New' classes. These results demonstrate the effectiveness of our method.

Table 4: **Results on the Herbarium-19 Dataset.**

Methods	All	Old	New
k-means [35]	13.0	12.2	13.4
RS+ [16]	27.9	**55.8**	12.8
UNO+ [12]	28.3	53.7	14.7
ORCA [2]	20.9	30.9	15.5
GCD [48]	35.4	51.0	27.0
Ours (GPC)	**36.5**	51.7	**27.9**



(a) Explained variance of the original feature w.r.t. the number of principal directions.



(b) The clustering ACC on validation set w.r.t. the number of principal directions

Figure 3: **The effects of the number of principal directions in PCA on the feature representations.**

4.3. Novel class number estimation

One of the important yet overlooked components in the NCD and GCD literature is the estimation of unknown class numbers. Our proposed framework leverages a modified GMM to estimate the class number, in which we need to define an initial guess of the class number. We validate the effects of different choices of the initial guess K_{init} w.r.t. the estimated class number in Tab. 5. Note that the number

Table 5: **Results of varying the initial guessed K^n_{init}.** 'GT K^n' is the ground truth number of novel classes. K^n is the estimated number of novel classes. Our proposed framework is generally robust in estimating the number of novel classes, and we found that using the initial guess of $K_{init} = K^l + \frac{K^l}{2}$ can be a simple and reliable choice.

Dataset	K^l	GT K^n	Vaze *et al.* [48]	$K^n_{init} =$	3	5	10	20	30	50	100
CIFAR-10	5	5	4	$K^n =$	5	5	5	6	6	8	14
CIFAR-100	80	20	20	$K^n =$	16	20	20	21	22	27	36
ImageNet-100	50	50	59	$K^n =$	58	48	57	55	54	50	60
CUB	100	100	131	$K^n =$	79	87	86	88	92	112	101
SCars	98	98	132	$K^n =$	84	90	86	87	89	115	104

in Tab. 5 is $K^n_{init} = K_{init} - K^l$. We can see that our proposed framework is generally robust to a wide range of initial guesses. We found that $K_{init} = K^l + \frac{K^l}{2}$ is a simple and reliable choice. Hence we use this for all datasets.

4.4. Training complexity

Given that our framework necessitates the fitting of GMMs, an extended training duration is required. We compare the performance of our method to the extended method of Vaze *et al.* [48], which is pushed to $1.5\times$ its original number of training epochs. The results in Tab. 6 demonstrate that our method outperforms [48] across both CUB and ImageNet-100 datasets while maintaining a similar training duration. Furthermore, in contrast to the baseline method, our approach eliminates the need for additional post-training procedures such as running SS-k-means on the entire dataset for label assignment.

Table 6: **Comparison between GPC and Vaze *et al.* [48] ($1.5\times$) on CUB and ImageNet-100 Datasets.**

Method	CUB			ImageNet-100		
	All	Old	New	All	Old	New
Vaze *et al.* [48]	51.3	56.6	48.7	74.1	89.8	66.3
Vaze *et al.*($1.5\times$)	52.0	56.8	49.0	74.7	90.3	66.7
Ours (GPC)	**55.4**	**58.2**	**53.1**	**76.9**	**94.3**	**71.0**

4.5. Ablation study

Number of dimensions in PCA The PCA in our framework requires setting a number for the number of principal directions to extract from data. In Fig. 3, we show the results of using a different number of principal directions in PCA on CUB-200 and ImageNet-100 datasets. We can see from Fig. 3a that for both datasets, 128 principal directions can already explain most of the variances in the data, thus we choose the PCA dimension to be 128 for all our experiments. We further experiment with other different choices of the PCA dimension and shows the result in Fig. 3b, which again confirms that 128 principal directions are already expensive enough, and obtain the best performance over other choices, that are either too few or too many, effectively avoiding DC.

Different methods for prototype estimation Our semi-supervised GMM plays an important role in prototype estimation for representation learning based on prototypical contrastive learning. Here, we replace our semi-supervised GMM with other alternatives that do not produce prototypes automatically. Particularly, we compare our method with DBSCAN [10], Agglomerative clustering [37], and semi-supervised k-means [17, 48]. The prototypes are then obtained by averaging the data points that are assigned to the same cluster. For a fair comparison, the same regulations to prevent the wrong clustering results for labelled instances are applied to all methods, *i.e.*, during the clustering process, two labelled instances with the same label will fall into the same cluster, and two instances with different labels will be assigned to different clusters. The results are reported in Tab. 7a. Our method achieves the best performance on all three datasets, indicating that better prototypes are obtained by our approach to facilitate representation learning. Note that DBSCAN requires two important user-defined parameters, radius, and minimum core points, the ideal values of which lack a principled way to obtain in practice, while our method is parameter-free and can seamlessly be combined with the representation learning to jointly enhance each other, obtaining better performance.

Combining our GMM with other GCD methods We further combine our semi-supervised GMM with automatic splitting and merging with other methods, allowing joint representation learning and category discovery without a predefined category number. As the state-of-the-art GCD method [48] does not contain any parametric classifier during representation, so it can be directly combined with our GMM. For the RankStat and the DualRank methods that have a parametric classifier for category discovery, we treat the weights of the classifier as the cluster centers and run our GMM to automatically determine the category number during representation learning. The results are presented

Table 7: **Combining components of GPC with other methods.** "IM-100" denotes ImageNet-100.

(a) Different prototype estimation methods.

Clustering Algo.	CUB	IM-100	SCars
Ester *et al.* [10]	45.6	66.1	34.8
Murtagh *et al.* [37]	52.1	74.6	39.8
Vaze *et al.* [48]	49.2	73.2	37.4
Ours (GPC)	54.1	76.6	41.9

(b) Combining our GMM with other methods.

Representation	CUB	IM-100	SCars
Han *et al.* [16]	34.6	38.4	29.3
Zhao *et al.* [58]	37.8	39.7	33.2
Vaze *et al.* [48]	50.6	73.4	37.8
Ours (GPC)	54.1	76.6	41.9

in Tab. 7b. Comparing with row 9 in Tab. 1 and Tab. 3, we can see using our GMM can also improve [48] on CUB and Stanford Cars, while our proposed framework consistently achieves better performance on all datasets, again validating that our design choices.

Class number estimation with different representations Here, we validate our class number estimation method on top of the representations learned by other GCD approaches and report the results in Tab. 8. It can be seen, applying our method on other GCD representations can achieve reasonably well results. Notably, by applying our class number estimation method on top of the representation by the existing state-of-the-art method, we can obtain better class number estimation results, though the overall best results are obtained with the representation learned in our framework.

Table 8: **Class number estimation with different learned representations.** "C-100" stands for the CIFAR-100 dataset.

Representation	C-100	CUB	SCars	IM-100
Ground Truth K^n	20	100	98	50
Vaze *et al.*[48]	20	131	132	59
Ours w/ [16] feat.	19	111	94	55
Ours w/ [58] feat.	22	116	89	49
Ours w/ [48] feat.	21	121	109	57
Ours (GPC)	20	112	103	53

Partial overlap between $\mathcal{Y}_l$ and $\mathcal{Y}_u$ We evaluate the performance of our method when we relax $\mathcal{Y}_l \subset \mathcal{Y}_u$ to $\mathcal{Y}_l \cap \mathcal{Y}_u \neq \emptyset$, *i.e.*, the two sets may only partially overlap. We vary the number of overlapped classes and report the results in Tab. 9. Our approach consistently outperforms the method proposed by Vaze *et al.* [48] in all configurations. These compelling results not only showcase the robustness of our method but also highlight its effectiveness in scenarios involving partial overlap between known and unknown classes.

Table 9: **Results on CUB of only partial overlap between $\mathcal{Y}_l$ and $\mathcal{Y}_u$.**

$\lvert\mathcal{Y}_l \cap \mathcal{Y}_u\rvert$	25			50			75		
	All	Old	New	All	Old	New	All	Old	New
Vaze *et al.* [48]	49.5	50.1	48.2	51.2	50.7	52.2	52.7	50.9	54.5
Ours (GPC)	51.2	52.6	49.5	52.3	51.6	54.8	53.6	51.4	55.9

Varying ratio of Old/New categories We measure the estimated new class numbers when varying the ratio of Old/New categories while having $K_{init} = K^l + \frac{K^l}{2}$ on CUB in Tab. 10. As can be seen, for all cases, our method outperforms [48]. Meanwhile, we can also see that when the initial guess is too far from the ground truth, the estimation will be less accurate.

Table 10: **Estimated class numbers on CUB with a varying ratio of Old/New classes.**

Old/New	20/180	40/160	60/140	80/120
GCD [48]	87	102	114	104
Ours (GPC)	93	126	135	116

5. Conclusion

In this paper, we present an EM-like framework for the challenging GCD problem without knowing the number of new classes, with the E-step automatically determining the class number and prototypes and the M-step being robust representation learning. We introduce a semi-supervised variant of GMM with a stochastic splitting and merging mechanism to obtain the prototypes and leverage these evolving prototypes for representation learning by prototypical contrastive learning. We demonstrated that class number estimation and representation learning can facilitate each other for more robust category discovery. Our framework obtains state-of-the-art performance on multiple public benchmarks.

Acknowledgements This work is supported by Hong Kong Research Grant Council - Early Career Scheme (Grant No. 27208022) and HKU Seed Fund for Basic Research.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 1, 3

[2] Kaidi Cao, Maria Brbić, and Jure Leskovec. Open-world semi-supervised learning. In *ICLR*, 2022. 2, 6, 7, 15

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 6, 15, 16

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[5] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2

[6] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI*, 2002. 3

[7] Quan Cui, Bingchen Zhao, Zhao-Min Chen, Borui Zhao, Renjie Song, Jiajun Liang, Boyan Zhou, and Osamu Yoshie. Discriminability-transferability trade-off: An information-theoretic perspective. In *ECCV*, 2022. 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 18

[9] 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 *ICLR*, 2021. 6

[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, 1996. 3, 8, 9

[11] Yixin Fei, Zhongkai Zhao, Siwei Yang, and Bingchen Zhao. Xcon: Learning with experts for fine-grained category discovery. In *BMVC*, 2022. 2

[12] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *ICCV*, 2021. 2, 6, 7

[13] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *ICCV*, 2017. 3

[14] Kai Han, Yandong Li, Sagar Vaze, Jie Li, and Xuhui Jia. What's in a name? beyond class indices for image recognition. *arXiv preprint arXiv:2304.02364*, 2023. 2

[15] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *ICLR*, 2020. 2

[16] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Autonovel: Automatically discovering and learning novel visual categories. *IEEE TPAMI*, 2021. 1, 2, 6, 7, 9

[17] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *ICCV*, 2019. 1, 2, 3, 8

[18] Shaozhe Hao, Kai Han, and Kwan-Yee K. Wong. Cipr: An efficient framework with cross-instance positive relations for generalized category discovery. *arXiv preprint arXiv:2304.06928*, 2023. 2

[19] Wilfred Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 1970. 2, 5

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[22] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. In *ICLR*, 2018. 2

[23] Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. Multi-class classification without multi-class labels. In *ICLR*, 2019. 2

[24] Tianyu Hua, Wenxiao Wang, Zihui Xue, Yue Wang, Sucheng Ren, and Hang Zhao. On feature decorrelation in self-supervised learning. In *ICCV*, 2021. 4

[25] Junkai Huang, Chaowei Fang, Weikai Chen, Zhenhua Chai, Xiaolin Wei, Pengxu Wei, Liang Lin, and Guanbin Li. Trash to treasure: harvesting ood data with cross-modal matching for open-set semi-supervised learning. In *ICCV*, 2021. 3

[26] Xuhui Jia, Kai Han, Yukun Zhu, and Bradley Green. Joint representation learning and novel category discovery on single- and multi-modal data. In *ICCV*, 2021. 2, 3

[27] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *ICLR*, 2021. 4

[28] K J Joseph, Sujoy Paul, Gaurav Aggarwal, Soma Biswas, Piyush Rai, Kai Han, and Vineeth N Balasubramanian. Novel class discovery without forgetting. In *ECCV*, 2022. 2

[29] K J Joseph, Sujoy Paul, Gaurav Aggarwal, Soma Biswas, Piyush Rai, Kai Han, and Vineeth N Balasubramanian. Spacing loss for discovering novel categories. In *CVPR Workshop on Continual Learning in Computer Vision*, 2022. 2

[30] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, 2013. 6, 18

[31] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Technical Report*, 2009. 6, 18

[32] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 3

[33] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2020. 2, 3, 4

[34] Mingxuan Liu, Subhankar Roy, Zhun Zhong, Nicu Sebe, and Elisa Ricci. Large-scale pre-trained models are surprisingly strong in incremental novel class discovery. *arXiv preprint arXiv:2303.15975*, 2023. 2

[35] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967. 3, 6, 7

[36] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 6, 18

[37] Fionn Murtagh and Pierre Legendre. Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of classification*, 2014. 8, 9

[38] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *NeurIPS*, 2018. 1

[39] Nan Pu, Zhun Zhong, and Nicu Sebe. Dynamic conceptional contrastive learning for generalized category discovery. In *CVPR*, 2023. 2

[40] Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Semi-supervised learning with scarce annotations. In *CVPR Deep-Vision workshop*, 2020. 3

[41] Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Lsd-c: Linearly separable deep clusters. In *ICCV Workshop on Visual Inductive Priors for Data-Efficient Deep Learning*, 2021. 3

[42] Meitar Ronen, Shahaf E. Finder, and Oren Freifeld. Deepdpm: Deep clustering with an unknown number of clusters. In *CVPR*, 2022. 3

[43] Kuniaki Saito, Donghyun Kim, and Kate Saenko. Openmatch: Open-set consistency regularization for semi-supervised learning with outliers. In *NeurIPS*, 2021. 3

[44] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020. 1, 3

[45] Yiyou Sun and Yixuan Li. Opencon: Open-world contrastive learning. *TMLR*, 2023. 2

[46] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 1, 3

[47] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020. 6

[48] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *CVPR*, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 14, 16, 17

[49] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need? In *ICLR*, 2022. 6, 7, 15

[50] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. Caltech-UCSD Birds 200. *Computation & Neural Systems Technical Report*, 2010. 6, 18

[51] Xin Wen, Bingchen Zhao, and Xiaojuan Qi. Parametric classification for generalized category discovery: A baseline study. In *ICCV*, 2023. 2

[52] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 2

[53] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, 2016. 3

[54] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *ECCV*, 2020. 3

[55] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *ICCV*, 2019. 3

[56] Sheng Zhang, Salman Khan, Zhiqiang Shen, Muzammal Naseer, Guangyi Chen, and Fahad Khan. Promptcal: Contrastive affinity learning via auxiliary prompts for generalized novel category discovery. In *CVPR*, 2023. 2

[57] Xinwei Zhang, Jianwen Jiang, Yutong Feng, Zhi-Fan Wu, Xibin Zhao, Hai Wan, Mingqian Tang, Rong Jin, and Yue Gao. Grow and merge: A unified framework for continuous categories discovery. *arXiv preprint arXiv:2210.04174*, 2022. 2

[58] Bingchen Zhao and Kai Han. Novel visual category discovery with dual ranking statistics and mutual knowledge distillation. In *NeurIPS*, 2021. 2, 9

[59] Bingchen Zhao and Oisin Mac Aodha. Incremental generalized category discovery. In *ICCV*, 2023. 2

[60] Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *CVPR*, 2021. 2

[61] 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 *CVPR*, 2021. 2

[62] Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving contrastive learning by visualizing feature transformation. In *ICCV*, 2021. 2