Dynamic Conceptional Contrastive Learning for Generalized Category Discovery

Nan Pu Zhun Zhong* Nicu Sebe
The Department of Information Engineering and Computer Science
University of Trento, Trento, Italy
{nan.pu, zhun.zhong, niculae.sebe}@unitn.it

Abstract

Generalized category discovery (GCD) is a recently proposed open-world problem, which aims to automatically cluster partially labeled data. The main challenge is that the unlabeled data contain instances that are not only from known categories of the labeled data but also from novel categories. This leads traditional novel category discovery (NCD) methods to be incapacitated for GCD, due to their assumption of unlabeled data are only from novel categories. One effective way for GCD is applying self-supervised learning to learn discriminate representation for unlabeled data. However, this manner largely ignores underlying relationships between instances of the same concepts (e.g., class, super-class, and sub-class), which results in inferior representation learning. In this paper, we propose a Dynamic Conceptional Contrastive Learning (DCCL) framework, which can effectively improve clustering accuracy by alternately estimating underlying visual conceptions and learning conceptional representation. In addition, we design a dynamic conception generation and update mechanism, which is able to ensure consistent conception learning and thus further facilitate the optimization of DCCL. Extensive experiments show that DCCL achieves new state-of-the-art performances on six generic and fine-grained visual recognition datasets, especially on fine-grained ones. For example, our method significantly surpasses the best competitor by 16.2% on the new classes for the CUB-200 dataset. Code is available at https://github.com/TPCD/DCCL

1. Introduction

Learning recognition models (*e.g.*, image classification) from labeled data has been widely studied in the field of machine learning and deep learning [13, 18, 31]. In spite of their tremendous success, supervised learning techniques rely heavily on huge annotated data, which is not suitable for open-world applications. Thus, the researchers recently have paid much effort on learning with label-imperfection data, such as semi-supervised learning [23, 33], self-supervised learning [12, 42], weakly-supervised learning [41,45], few-shot learning [32,38], open-set recognition [30] and learning with noisy labels [40], etc.

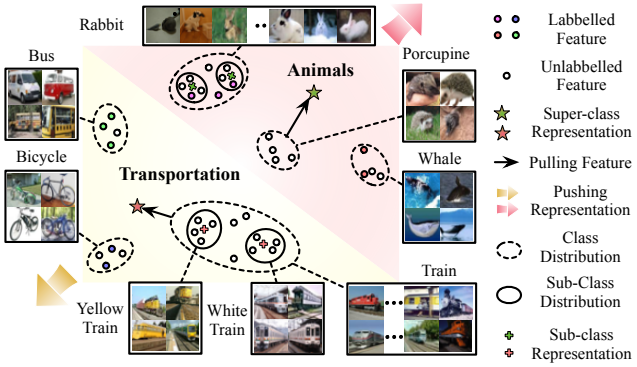


Figure 1. Diagram of the proposed Dynamic Conceptional Contrastive Learning (DCCL). Samples from the conceptions should be close to each other. For example, samples from the same classes (bus) at the class level, samples belonging to the transportation (bus and bicycle) at the super-class level, and samples from trains with different colors at the sub-class level. Our DCCL potentially learns the underlying conceptions in unlabeled data and produces more discriminative representations.

Recently, inspired by the fact that Humans can easily and automatically learn new knowledge with the guidance of previously learned knowledge, novel category discovery (NCD) [9, 11, 28, 44, 47] is introduced to automatically cluster unlabeled data of unseen categories with the help of knowledge from seen categories. However, the implementation of NCD is under a strong assumption that all the unlabeled instances belong to unseen categories, which is not practical in real-world applications. To address this limitation, Vaze *et al*. [35] extend NCD to the generalized category discovery (GCD) [35], where unlabeled images are

*Corresponding Author.

from both novel and labeled categories.

GCD is a challenging open-world problem in that we need to 1) jointly distinguish the known and unknown classes and 2) discover the novel clusters without any annotations. To solve this problem, Vaze *et al*. [35] leverage the contrastive learning technique to learn discriminative representation for unlabeled data and use k-means [21] to obtain final clustering results. In this method, the labeled data are fully exploited by supervised contrastive learning. However, self-supervised learning is applied to the unlabeled data, which enforces samples to be close to their augmentation counterparts while far away from others. As a consequence, the underlying relationships between samples of the same conceptions are largely overlooked and thus will lead to degraded representation learning. Intuitively, samples that belong to the same conceptions should be similar to each other in the feature space. The conceptions can be regarded as: classes, super-classes, sub-classes, etc. For example, as shown in Fig. 1, samples of the same class should be similar to each other, *e.g.,* samples of the bus, samples of the bicycle. In addition, in the super-classes view, classes of the transportation, *e.g.*, Bus and Bicycle, should belong to the same concept. Hence, the samples of transportation should be closer than that of other concepts (*e.g.*, animals). Similarly, samples belong to the same sub-classes (*e.g.,* red train) should be closer to that of other sub-classes (*e.g.,* white train). Hence, embracing such conceptions and their relationships can greatly benefit the representation learning for unlabeled data, especially for unseen classes.

Motivated by this, we propose a Dynamic Conceptional Contrastive Learning (DCCL) framework for GCD to effectively leverage the underlying relationships between unlabeled data for representation learning. Specifically, our DCCL includes two steps: Dynamic Conception Generation (DCG) and Dual-level Contrastive Learning (DCL). In DCG, we dynamically generate conceptions based on the hyper-parameter-free clustering method equipped with the proposed semi-supervised conceptional consolidation. In DCL, we propose to optimize the model with conception-level and instance-level contrastive learning objectives, where we maintain a dynamic memory to ensure comparing with the up-to-date conceptions. The DCG and DCL are alternately performed until the model converges.

We summarize the contributions of this work as follows:

- We propose a novel dynamic conceptional contrastive learning (DCCL) framework to effectively leverage the underlying relationships between unlabeled samples for learning discriminative representation for GCD.
- We introduce a novel dynamic conception generation and update mechanism to ensure consistent conception learning, which encourages the model to produce more discriminative representation.
- Our DCCL approach consistently achieves superior performance over state-of-the-art GCD algorithms on both generic and fine-grained tasks.

2. Related Work

2.1. Novel Category Discovery

Novel category discovery [11] (NCD) tasks aim at discovering new categories by leveraging the knowledge of a set of labeled categories. RankStat [10] indicates that self-supervised pre-training is helpful for NCD. NCL [46] adopts contrastive learning to improve representation learning. UNO [9] proposes a unified objective for jointly learning on unlabeled and labeled data. Most recently, NCD has been extended to a generalized category discovery (GCD) [35], in which the unlabeled data include both labeled and unlabeled categories. Later, ORCA [3] defines an open-world semi-supervised learning task, which is similar to GCD. Although these definitions are relatively practical, most methods remain to assume that the class number of clustered data is known. Nevertheless, such prior knowledge is often not acquired in advance for real-world applications. To handle the drawback, DTC [11] and GCD [35] employed an independent algorithm to search the optimal class number, however, they did not associate clustering estimation with representation learning. *Unlike these offline algorithms, we propose to jointly consider downstream clustering and representation learning. The experimental results show they are mutually beneficial for each other.*

2.2. Contrastive Learning based on Memory Buffer

Contrastive learning [5, 6, 22, 37, 43] (CL) has been shown to be significantly effective for representation learning in a self-supervised manner. MoCo [12] demonstrates that sampling positive-negative pairs from an instance-level buffer can benefit CL and reduce the impact of the size of the training batch. Then, instead of contrasting over all instances in a mini-batch, prototypical contrastive learning [20] (PCL) that contrasts the instance features with a set of prototypes, has been shown to provide comprehensive supervision. However, PCL still needs an instance-level memory buffer to yield the prototype set, which is not computation- and memory-efficient. Recently, SCL [14] propose a cluster-level momentum encoder but considers only three fixed numbers of classes during training, which is still limited compared to our dynamic method. *Different from PCL and SCL that consider the fixed numbers of classes during the whole training process, our DCCL dynamically estimates the number of classes for different training stages in a efficient way, which encourages models to learn more discriminative representation.*

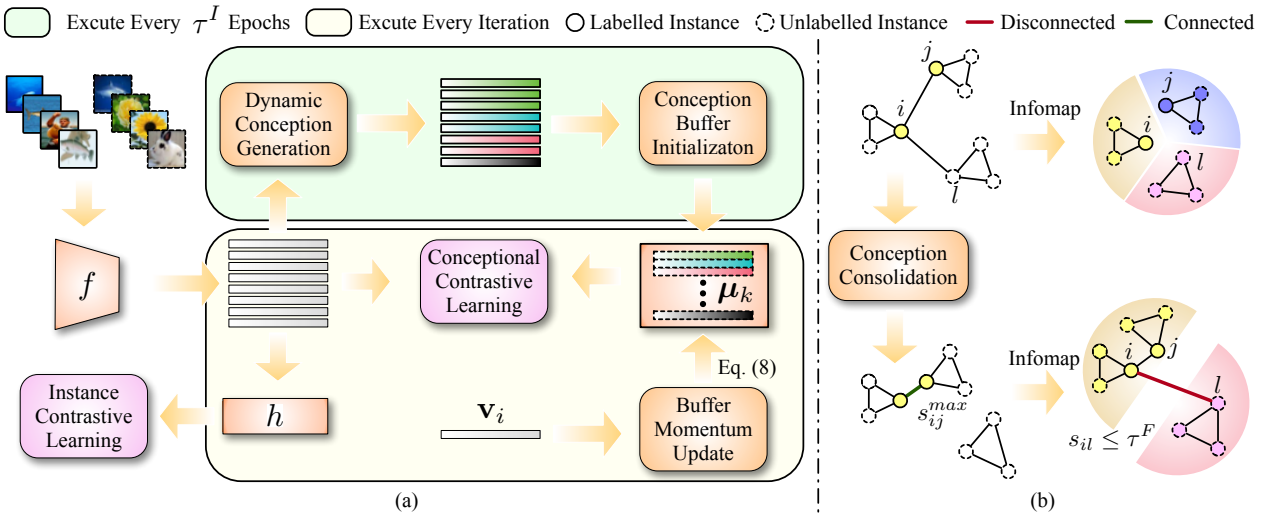


Figure 2. (a) Overview of our DCCL framework. We first extract features and cluster the features to generate conceptional labels, then initialize conception representations by our DCG, and last learn representations by joint instance-level and conception-level objectives. During the training process, the DCG and dual-level representation learning are performed alternately, in which the conception buffer is updated every iteration to keep the consistency of the changing instance features and conceptional representations. (b) Illustration of the proposed conception consolidation. Without consolidating the relationships of conceptions by label information, Infomap tends to over-cluster data and thus provides the supervision that has a high risk to over-correct affinities between neighbor instances.

2.3. Semi-Supervised Learning and Clustering

Semi-supervised learning (SSL) has been a long-standing research topic [39]. Different from GCD, SSL often assumes that the labeled and unlabeled data come from the same set of classes, in which consistency-based methods are the most effective methods for SSL, such as Mean-teacher [34], MixMatch [2], and FixMatch [33]. Consistency-based methods are the most effective methods for SSL, such as Mean-teacher [34], MixMatch [2], and FixMatch [33]. Moreover, semi-supervised classification is a relatively well-defined task, while the supervised information in semi-supervised clustering can take different forms [19], such as two instances are known to be must-linked in a relationship matrix or some cluster assignments are known beforehand. For instance, Basu et al. [1] proposed to initialize the clusters based on the data points for which cluster assignments are known. However, these methods can not adaptively assign the number of clusters for generating conceptions. *To mitigate these limitations, we improve the classical InfoMap [27] to fully leverage both labeled and unlabeled data and dynamically produce varying conceptional labels for different contrastive learning epochs.*

3. Dynamic Conceptional Contrastive Learning

3.1. Problem Formulation

Generalized category discovery (GCD) aims at automatically categorizing unlabeled images in a dataset, where the partial data are labeled and the remaining are unlabeled. The unlabeled images come from either labeled (known) classes or unlabeled (unknown) classes. This is a more realistic open-world setting than the common closed-set classification that assumes the labeled and unlabeled data belong to the same classes. Let the dataset be $\mathcal{D} = \mathcal{D}^L \cup \mathcal{D}^U$, where $\mathcal{D}^L = \{(\mathbf{x}_i^L, \mathbf{y}_i^L)\}_{i=1}^{M^L} \in \mathcal{X} \times \mathcal{Y}^L$, L denotes the labeled subset and $\mathcal{D}^U = \{(\mathbf{x}_i^U, \mathbf{y}_i^U)\}_{i=1}^{M^U} \in \mathcal{X} \times \mathcal{Y}^U$ denotes the unlabeled subset with unknown $\mathbf{y}_i^U \in \mathcal{Y}^U$. Only a subset of classes contains labeled instances, i.e., $\mathcal{Y}^L \subset \mathcal{Y}^U$. The number of labeled classes N^L can be directly calculated from the labeled data, while the number of unlabeled classes N^U is not known during model training. Let f and h be a feature extractor and a MLP projection head. The extracted representation is defined as $\mathbf{v}_i = f(\mathbf{x}_i)$.

3.2. Overview

To tackle the problem of GCD, we propose a novel framework DCCL (see Fig. 2), to jointly learn representations using dual-level contrastive learning (DCL) and explore all possible relationships between labeled and unlabeled instances in a conceptional view. First, we extend the classical unsupervised clustering algorithm, Infomap [27], to a semi-supervised manner, which aims at dynamically generating reasonable conceptional representations and associating the labeled and unlabeled instances during representation learning. By alternately executing the dynamic conception generation (DCG) and DCL, DCL benefits from informative supervised information to generate higher-quality representations. Meanwhile, DCG gradually

produces more comprehensive guidance based on a deeper understanding of conceptual relationships. DCG and DCL mutually benefit each other, thus resulting in a better representation for downstream clustering.

3.3. Dynamic Conception Generation

Although the test-time semi-supervised k-means [35] (SSK) succeeds in achieving considerable performance gain, it fails to jointly consider the representation learning and supervision information from semi-supervised clustering. Moreover, it is infeasible to directly perform SSK in each training epoch for assigning pseudo labels, because the real number of clusters is unknown during training. To overcome these drawbacks, we propose a dynamic conception generation (DCG) based on the hyper-parameter-free Infomap [27] algorithm. Specifically, we first propose a semi-supervised conceptional consolidation method to construct a similarity network, then execute the Infomap clustering algorithm on the constructed network to get conceptional label assignments, and finally calculate conception representation and initialize conceptional memory buffer.

Conception Consolidation. In a given network, Infomap aims at partitioning semantic-similar sub-networks by the pattern of connections. To leverage the supervision from labeled data, we propose to enforce the similarity constraints into the networking, according to the labeled data that belong to the same category. Formally, we construct an adjacent matrix $\mathcal{A}$ to represent the possible connection relationships among all instances. The weight of the edge of the i-th and j-th instances is given by:

$$\mathcal{A}_{ij} = \begin{cases} s_i^{max}, & \text{if } \mathbf{y}_i, \mathbf{y}_j \in \mathcal{Y}^L \text{ and } \mathbf{y}_i = \mathbf{y}_j \\ s_{ij}, & \text{if } \mathbf{y}_i \text{ or } \mathbf{y}_j \in \mathcal{Y}^U \text{ and } s_{ij} > \tau^F \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$s_i^{max} = \arg\max_j \{s_{ij} \mid j \in \mathcal{D}\}, \quad (2)$$

$$s_{ij} = [(\mathbf{v}_i / \|\mathbf{v}_i\|) \cdot (\mathbf{v}_j / \|\mathbf{v}_j\|) + 1]/2 \in [0, 1], \quad (3)$$

where $\cdot$ denotes dot product and $\|\cdot\|$ is l_2 normalization. The τ^F is a threshold to select high-confidence links. Through the conception consolidation illustrated in Fig. 2(b), we can establish a reliable relationship network with rich structural information for the subsequent clustering.

Remark. We set the similarities of positive pairs with the maximal value of neighborhood similarities instead of 1. This is because we experimentally find that when imposing 1 for constraining positive pairs, Infomap tends to group all the labeled positive instances as an individual cluster.

Entropy Minimization Clustering. In Infomap algorithm [26], the clustering problem is equivalent to minimizing the entropy that represents the minimum description length of the coding network. By solving the minimization objective, we acquire a conceptional label set $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{M^L+M^U} \in \mathcal{Y}^G$ for both labeled and unlabeled instances. Then, we combine the extracted feature vectors $\mathcal{V}$ and corresponding conceptional labels to construct a generated feature dataset $\mathcal{D}^G = \{(\mathbf{v}_i, \mathbf{c}_i)\}_{i=1}^{M^L+M^U} \in \mathcal{V} \times \mathcal{Y}^G$. $|\mathcal{Y}^G|$ denotes the number of estimated conceptions.

Algorithm 1: Algorithm Pipeline of our DCCL

Input: Feature Extractor f, Projection Head h, Labeled data $\mathcal{D}^L$ and Unlabeled data $\mathcal{D}^U$.
Output: f and h.
for $n = 1$ ***in*** $[1, max\_epoch]$ **do**
  if n ***mod*** $\tau^I == 0$ **then**
    Extract features and construct adjacency matrix $\mathcal{A}$ by Eq. (1), Eq. (2) and Eq. (3);
    Perform InfoMap [27] clustering to assign conceptional labels $\mathcal{C}$;
    Initialize conceptional buffer by Eq. (4);
  end
  for $i = 1$ ***in*** $[1, max\_iteration]$ **do**
    Sample mini-batches from $\mathcal{D}^L \cup \mathcal{D}^U$;
    Calculate overall optimization objective by Eq. (10);
    Update f and h by SGD [25];
    Update conceptional buffer by Eq. (8);
  end
end

Conceptional Memory Initialization. In this paper, our DCCL maintains a conception-level memory buffer that provides dynamic conceptional representations for dual-level contrastive learning, which is elaborated in Sec. 3.4. Here, we introduce the initialization of the conceptional memory buffer (CMB). We use the mean feature vector of the instances that share the same conceptional label to form a unique conceptional representation. Formally, the initial conceptional representation set is defined as:

$$\mathcal{U} = \{\boldsymbol{\mu}_k\}_{k=1}^{K}, \quad \boldsymbol{\mu}_k = \frac{1}{|\mathcal{D}_k^G|} \sum_{\mathbf{v}_i \in \mathcal{D}_k^G} \mathbf{v}_i, \quad K = |\mathcal{Y}^G|, \quad (4)$$

where $\mathcal{D}_k^G$ denotes the k-th conception subset, *i.e.*, if $\mathbf{v}_i \in \mathcal{D}_k^G, \mathbf{c}_i = k$. During whole training process, the initialization of CMB is executed every τ^I epochs on center-cropped images. Thus, the number of conceptions K is dynamically changing along with model training.

3.4. Dual-Level Contrastive Learning

In this section, we first explain the proposed conception-level contrastive learning, then elaborate on the update of the conceptional memory buffer, and finally introduce the employed instance-level contrastive learning approach.

Conception-Level Contrastive Learning. Based on the generated conceptional representations in Sec. 3.3, we

propose to perform conception-level contrastive learning. Specifically, we first sample N^C conception labels and a fixed number N^I of instances for each conception label, resulting in a mini-batch $\mathcal{B}^C$ with $N^C \times N^I$ instances. Next, each instance representation is compared to all the conceptional representations. We pull the instance representation from its corresponding conceptional representation closer and push the instance representation away from other conceptional representations. Formally, we define the conceptional contrastive loss function as the following:

$$\mathcal{L}_i^C = -\log \frac{\exp\left(\mathbf{v}_i \cdot \boldsymbol{\mu}_{\mathbf{c}_i}/\tau^C\right)}{\sum_{k=1,k\neq \mathbf{c}_i}^{K} \exp(\mathbf{v}_i \cdot \boldsymbol{\mu}_k/\tau^C)}, \tag{5}$$

where τ^C is a temperature hyper-parameter to control the strength of the conception-level contrastive learning. In addition, in order to explicitly encourage learned representations with a large inter-conception margin, we propose a dispersion loss to further push the different conception representations away from each other. The loss function for the m-th and the n-th conceptions in $\mathcal{B}^C$ is:

$$\mathcal{L}(m,n) = \left[\tau^M - || \frac{1}{|\mathcal{B}_m^C|} \sum_{\mathbf{v}_i \in \mathcal{B}_m^C} \mathbf{v}_i || \cdot || \frac{1}{|\mathcal{B}_n^C|} \sum_{\mathbf{v}_j \in \mathcal{B}_n^C} \mathbf{v}_j || \right]_+, \tag{6}$$

where τ^M is a threshold to filter the conception pairs with high uncertainty. We assume that two conception representations that are close tend to be highly entangled conceptions. Separating these conceptions has a high risk of over-correct. We explore the impact of τ^M in Sec. 4.4. The dispersion loss function over a mini-batch is defined as:

$$\mathcal{L}^D = \frac{1}{N^C} \sum_{m=1}^{N^C} \frac{1}{N^C} \sum_{n=1}^{N^C} \mathcal{L}(m,n). \tag{7}$$

Conceptional Memory Update. Different from [12, 20] that need to save all training instances, our CMB stores only the conception representations, which significantly reduces storage cost. Furthermore, the instance-wise update is easy to lead to an inconsistent update during each training iteration. To mitigate this drawback, we first adopt the re-sampling method as detailed in the previous section. Next, we propose to update the conception representation by each corresponding instance feature following a momentum update mechanism. The process is formulated as:

$$\boldsymbol{\mu}_{\mathbf{c}_i} \leftarrow \eta \boldsymbol{\mu}_{\mathbf{c}_i} + (1-\eta)\mathbf{v}_i, \tag{8}$$

where η is the momentum updating factor.

Remark. DCCL updates the memory buffer and computes the losses both at the conceptional level, which consistently updates the conceptional representation to maintain the conceptional consistency during the whole training process.

Instance-Level Contrastive Learning. Inspired by [35], we combine supervised contrastive loss and self-supervised contrastive loss as an instance contrastive loss (ICL), to fine-tune the model. Formally, we assume x_i and $\hat{x}_i$ are two views (random augmentations) of the same image in a randomly-sampled mini-batch $\mathcal{B}^I$. Let h be a MLP projection head. The extracted representation $\mathbf{v}_i$ is further projected by h to high-dimensional embedding space for instance-level contrastive learning. The loss function is:

$$\mathcal{L}_i^I = (\lambda - 1) \log \frac{\exp\left(h\left(\mathbf{v}_i\right) \cdot h\left(\hat{\mathbf{v}}_i\right)/\tau\right)}{\sum_{j\in\mathcal{B}^I, j\neq i} \exp\left(h\left(\mathbf{v}_i\right) \cdot h\left(\mathbf{v}_j\right)/\tau^S\right)}$$
$$\frac{-\lambda}{|\mathcal{P}(i)|} \sum_{p\in\mathcal{P}(i)} \log \frac{\exp\left(h\left(\mathbf{v}_i^L\right) \cdot h\left(\mathbf{v}_p^L\right)/\tau\right)}{\sum_{j\in\mathcal{B}^L, j\neq i} \exp\left(h\left(\mathbf{v}_i^L\right) \cdot h\left(\mathbf{v}_j^L\right)/\tau^L\right)}, \tag{9}$$

where $\mathcal{B}^L$ donates the labeled subset within the mini-batch $\mathcal{B}^I$ and $\mathcal{B}^I = \mathcal{B}^L \cup \mathcal{B}^U$. $\mathcal{P}(i)$ is the positive index set for the anchor image $i \in \mathcal{B}^L$. λ is a trade-off factor to balance the contributions of self-supervised and supervised learning. For a fair comparison, we follow [35] and set λ to 0.35.

3.5. Joint Optimization

During the whole training process, we alternately perform dynamic conception generalization and dual-level contrastive representation learning, until the maximal training epoch. The pseudo-code of DCCL is elaborated in Algorithm 1. The overall objective over on mini-batch $\mathcal{B}$ is given by the weighted sum of each loss function:

$$\mathcal{L}_{total} = \frac{1}{|\mathcal{B}|} \sum_{i\in\mathcal{B}} \mathcal{L}_i^I + \alpha \frac{1}{|\mathcal{B}|} \sum_{i\in\mathcal{B}} \mathcal{L}_i^C + \beta \mathcal{L}^D, \tag{10}$$

where α and β are the weights to adjust the strengths of two loss functions. In all experiments, we use l_2 normalized feature vector $||\mathbf{v}||_2$ for clustering evaluation.

4. Experiments

4.1. Experimental Setup

Data and Evaluation Metric. We evaluate DCCL on three generic image classification datasets, namely CIFAR-10 [17], CIFAR-100 [17] and ImageNet-100 [35]. ImageNet-100 denotes randomly sub-sampling 100 classes from the ImageNet [7] dataset. The dataset statistics and train-test splits are described in Tab. 1. We further evaluate DCCL on three more challenging fine-grained image classification datasets: CUB-200 [36], Stanford Cars [16], and Oxford-IIIT Pet [24]. The original training set of each fine-grained dataset is separated into labeled and unlabeled parts. We follow [35] sample a subset of half the classes as "Old" categories. 50% of instances of each labeled class are drawn to form the labeled set, and all the remaining data constitute the unlabeled set. For evaluation, we measure the clustering

Table 1. Statistics of the datasets and the splits for GCD. The first three are generic datasets while the last three are fine-grained datasets.

Dataset		CIFAR10 [17]	CIFAR100 [17]	ImageNet-100 [7]	CUB-200 [36]	SCars [16]	Pet [24]
Labelled	# Classes	5	80	50	100	98	19
	# Images	12,500	20,000	31,860	1,498	2,000	942
Unlabelled	# Classes	10	100	100	200	196	37
	# Images	37,500	30,000	95,255	4496	6,144	2,738

Table 2. Results on generic image recognition datasets.

Method	CIFAR10			CIFAR100			ImageNet-100		
	All	Old	New	All	Old	New	All	Old	New
k-means	83.6	85.7	82.5	52.0	52.2	50.8	72.7	75.5	71.3
RankStats+	46.8	19.2	60.5	58.2	77.6	19.3	37.1	61.6	24.8
UNO+	68.6	**98.3**	53.8	69.5	80.6	47.2	70.3	95.0	57.9
GCD	91.5	97.9	88.2	73.0	76.2	66.5	74.1	89.8	66.3
DCCL	**96.3**	96.5	**96.9**	**75.3**	**76.8**	**70.2**	**80.5**	**90.5**	**76.2**

Table 3. Results on fine-grained datasets.

method	CUB-200			Stanford-Cars			Oxford-Pet		
	All	Old	New	All	Old	New	All	Old	New
k-means	34.3	38.9	32.1	12.8	10.6	13.8	77.1	70.1	80.7
RankStats+	33.3	51.6	24.2	28.3	61.8	12.1	-	-	-
UNO+	35.1	49.0	28.1	35.5	**70.5**	18.6	-	-	-
GCD	51.3	56.6	48.7	39.0	57.6	29.9	80.2	85.1	77.6
DCCL	**63.5**	**60.8**	**64.9**	**43.1**	55.7	**36.2**	**88.1**	**88.2**	**88.0**

accuracy by comparing the predicted label assignment with the ground truth, following the protocol in [35].

Implementation Details. We adopt the ViT-B-16 pre-trained by DINO [4] as our backbone network. The output [CLS] token is used as the feature representation, which is also used for conception-level contrastive learning. Following [35], we project the representations by a projection head and use the projected embeddings for instance-level contrastive learning. We set the dimension of projected embeddings to 65,536 following [4]. At training time, we feed two views with random augmentations to the model. We only fine-tune the last block of the vision transformer with an initial learning rate of 0.01 and the head is trained with an initial learning rate of 0.1. All methods are trained for 200 epochs with a cosine annealing schedule. The size of the mini-batch is set to 128 with N^C=8 and N^I=16. For a fair comparison, we follow [35] and set the temperatures of two supervised contrastive losses τ^S, τ^L and τ^C to 0.07, 0.05 and 0.05, respectively. The τ^F is empirically set to 0.6 for the fine-grained datasets and 0.7 for the generic datasets. Other hyper-parameters are discussed in Sec. 4.4. In testing, we first use the class number estimation algorithm [35] to predict the number of classes of the testing dataset, and then use semi-supervised k-means for clustering evaluation. In dynamic conception generation, we adopt faiss [15] to accelerate the construction of relationship networks. Our experiments are conducted on RTX 3090 GPUs.

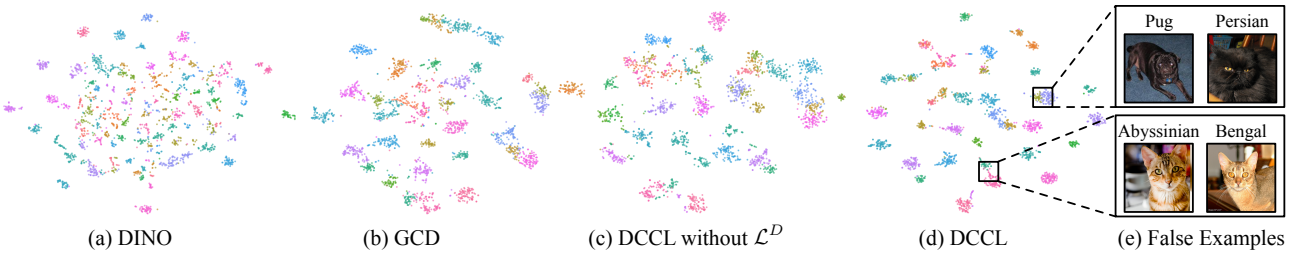
4.2. Comparison with State-of-the-Art

To evaluate the performances of our DCCL, we conduct three group experiments by comparing our DCCL with three strong GCD baselines, including RankStats [10] and UNO [9] and the state-of-the-art GCD method [35].

Comparison on Generic Datasets. As shown in Tab. 2, our DCCL is compared with other competitors on the generic image recognition datasets. Overall, the results in Tab. 2 show our DCCL consistently outperforms all others by a significant margin. Specifically, DCCL outperforms the GCD method [35] by 4.8% on CIFAR-10, 2.3% on CIFAR-100, and 6.4% on ImageNet-100 for 'All' classes, and by 8.7% on CIFAR-10, 3.7% on CIFAR-100, and 9.9% on ImageNet-100 for 'Unseen' classes. These results experimentally demonstrate the generated dynamic conceptions provide effective supervision to learn better representations for unlabeled data. Moreover, UNO+ shows a strong accuracy on "Old" classes, but its accuracy when testing on "New" classes is relatively lower. This is because UNO+ trains the linear classifier on "Old" classes, thus resulting in an inevitable bias. On the contrary, our DCCL gets a relatively good balance on both the "Old" and "Unseen" classes, without bias to the labeled data.

Comparison on Fine-Grained Datasets. In general, the differences between different classes in fine-grained datasets are subtle, which leads the fine-grained visual understanding to be more challenging for GCD. For verifying the effects of DCCL on fine-grained tasks, we compare our method with others on fine-grained image recognition datasets. The results in Tab. 3 show that DCCL consistently outperforms all other methods for "All" and "New" classes. Specifically, on CUB-200 and Oxford-Pet, DCCL achieves 12.2% and 7.9% improvement over the state-of-the-art for "All" classes. Especially for "New" classes, DCCL outperforms GCD by 16.2% on the CUB-200 dataset. These results demonstrate that our DCCL is efficient in capturing the conceptional information shared across different fine-grained classes, thereby generating precise and helpful supervision for representation learning.

Visualization of Feature Distributions. To qualitatively explore the clustered features on Pets dataset [24], we visualize the t-SNE embeddings projected from the features extracted by pre-trained ViT [4], GCD [35], the DCCL without the proposed dispersion loss and our full DCCL method. As shown in Fig. 3, our features are more discriminative than the features from the pre-trained ViT and GCD. By comparing Fig. 3(c) and Fig. 3(d), the proposed dispersion loss effectively pushes cluster centers away from



(a) DINO (b) GCD (c) DCCL without $\mathcal{L}^D$ (d) DCCL (e) False Examples

Figure 3. Visualization of features distributions of the unlabeled set of the Pet [24] dataset. (a)-(d) are the results generated from DINO [4], GCD [35], our DCCL without $\mathcal{L}^D$ and full DCCL, in turn. (e) is a visualization of false samples that are easy to be incorrectly clustered.

Table 4. Effectiveness of each component of our DCCL. "MU" and "CC" denote the proposed momentum update by Eq. (8) and the conception consolidation proposed in Sec. 3.3, respectively.

Index	Component					CUB-200 [36]			Pet [24]		
	$\mathcal{L}^I$	$\mathcal{L}^C$	$\mathcal{L}^D$	MU	CC	All	Old	New	All	Old	New
a)	✓					51.3	56.6	48.7	80.2	85.1	77.6
b)		✓				54.9	52.3	55.4	81.6	80.7	81.0
c)		✓		✓		57.7	54.0	58.1	83.5	81.1	80.3
d)		✓	✓	✓		59.5	53.3	60.8	84.3	83.1	84.5
e)		✓	✓	✓	✓	60.1	59.4	60.7	85.8	86.8	84.6
f)	✓	✓	✓	✓	✓	**63.5**	**60.8**	**64.9**	**88.1**	**88.2**	**88.0**

each other. A large inter-cluster margin not only improves cluster boundaries for "Old" and "New" categories, but also compacts intra-cluster distribution.

Summary. The experimental results show that our DCCL achieves significant improvements on both generic and fine-grained datasets. Especially for discovering "New" categories in challenging fine-grained tasks, our dynamic conceptional contrastive learning succeeds in mining shared conceptions, which is especially beneficial for the generalized fine-grained new category discovery.

4.3. Effectiveness Evaluation

To verify the effectiveness of each component in our DCCL, we conduct five experiments on both CUB-200 [36] and Pet [24] datasets, as shown in Tab. 4. Note that the configuration of the experiment a) is the same with GCD [35], which is the baseline method in our experiments.

Effectiveness of Conceptional contrastive Learning. Based on the results of the experiment a) and b), we find that using only the conceptional contrastive learning can achieves competitive performance, compared to baseline method with only instance-level contrastive learning.

Effectiveness of Conception-Level Momentum Updating. Comparing the experiment b) and c), we can find that consistent update of conceptional representations by the proposed momentum update can bring considerable improvements on both new and old classes. This implies that due to periodically generating conceptional representations, the conceptional labels are kept fixed within one training period, which leads to a severe sub-optimal problem, during conceptional contrastive learning. Thus, the proposed momentum update mitigates the problem to some extent.

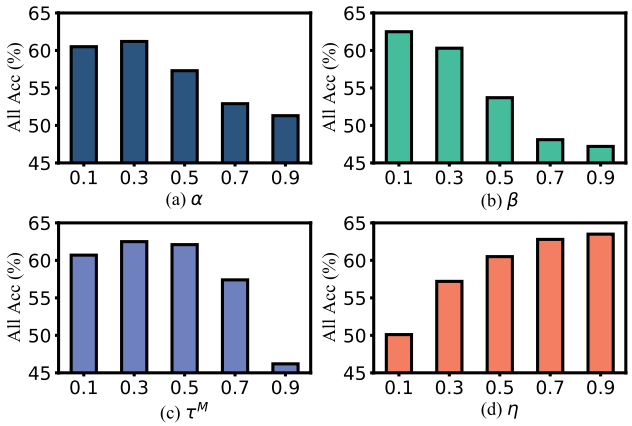


Figure 4. Impact of hyper-parameters. The clustering accuracy on "All" categories is reported.

Effectiveness of Dispersion Loss. The experiment d) in Tab. 4 shows that by adding the proposed dispersion loss, the model's performances on new classes acquire further improvements by 2.7% on "New" classes for the CUB-200 dataset. The improvements can be observed in Fig. 3(d).

Effectiveness of Conception Consolidation. Without the proposed conception consolidation that considers labeled information to impose semi-supervised constraintsd, our DCCL suffers from a performance balance between the new and the old classes, as shown in Tab. 4 b), c) and d). In the experiment e) and f), our full method shows superior performance on all evaluation metrics, which experimentally demonstrates that our conception consolidation plays an essential role in rectifying the estimated latent conception relations between seen and unseen classes.

4.4. Hyper-Parameter Analyses

In this section, we discuss the impact of the hyper-parameters in our DCCL, including loss weights (α and β), the threshold parameter of dispersion loss (τ^M), momentum updating factor (η), and the frequency of DCG (τ^I).

Impact of loss weights and threshold parameters. For the evaluation of loss weights, we use the hold-off validation data to determine their values. Specifically, we first select the optimal α to achieve the best accuracy on the "All" score, then we find the optimal β based on the selected α. The impact of different values is shown in Fig. 4 (a)-(c).

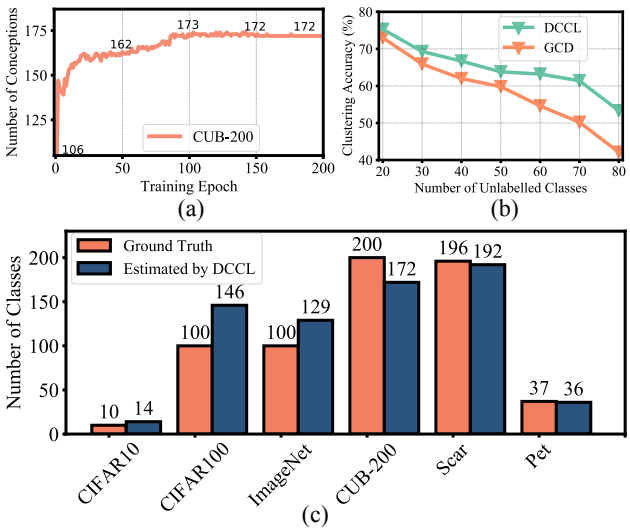


Figure 5. (a) visualizes the dynamics of DCG, which implies that our DCG adaptively estimates conceptional representation for different training stages. (b) illustrates the trend of performances on the CIFAR100 dataset, with varying the number of unlabeled classes. (c) is the comparison between the real number of classes in datasets and the estimated number of conceptions.

Similarly, we choose the best τ^M with the selected α and β. Finally, our final model is obtained by using $\alpha = 0.3$, $\beta = 0.1$ and $\tau^M = 0.3$.

Impact of Update Rates. The impact of updating factor of CMB is illustrated in in Fig. 4 (d). The more smooth running average can obtain better performance. Considering the balance between computational consumption and performance, we thus set $\tau^I = 5$ and $\eta = 0.9$ in all experiments.

4.5. Further Investigation of DCG

To explore our dynamic conception generation (DCG), we conduct two group experiments and the experimental results are shown in Fig. 5 and Tab. 5. From the Fig. 5 (a) we find that in the initial training stage, our DCG tends to generate fewer conceptions than at the convergence stage. This is because in the beginning, the model could not understand fine-grained classes well. Thus, DCG generates coarse-grained supervision like super-class, which has a low risk to over-correct. Later, with the growth of feature discriminability, DCG builds elaborate conceptional relationships to further refine the learned representations.

From the Fig. 5 (c), we find that the number of conceptions estimated by DCCL is close to the ground truth of the number of classes in the corresponding dataset. However, it is worth noting that the conception representations in this paper are not equivalent to the cluster centers. The DCG aims to adaptively generate proper conceptional representations that are beneficial for contrastive learning, instead of predicting the real number of classes within a dataset. Furthermore, we try to directly replace our DGC with semi-supervised k-means [35] (SSK) with the prior of known k. From the results in Tab. 5, SSK fails to generate effective latent conceptions, even leading to worse clustering.

Table 5. Comparison of different clustering methods for DCG.

Clustering Method	CUB-200 [36]		
	All	Old	New
k-means [21]	54.1	53.4	53.3
SSK [35]	55.9	55.1	54.2
FINCH [29]	55.8	56.1	55.4
DBSCAN [8]	60.5	54.7	59.8
InfoMap [27]	61.4	55.2	62.7
DCG (Ours)	**63.5**	**60.8**	**64.9**

Discussion. Based on the comparison in Fig. 5 (b), we notice that when training on generic datasets, our DCG tends to generate more conceptions than the actual number of dataset classes, while generating fewer conceptions on fine-grained datasets. A possible explanation is that due to large inter-class differences in generic datasets, learning more sub-classes conceptions enables models to gain discriminability. Similarly, since fine-grained classes share more common attributes or conceptions, such super-class information can significantly benefit fine-grained GCD.

4.6. Evaluation with Different Split Protocols

To explore the effects of DCCL under strict annotation limitation, we propose to test models on varying splits of CIFAR100 [17]. We visualize the accuracy of "All" classes in Fig. 5 (b), which indicates that our DCCL has stronger robustness when only a few labeled classes are available. Meanwhile, with the growth of the number of unlabeled classes, GCD expresses a severe performance degradation.

5. Conclusion

In this paper, we propose to cope with the generalized category discovery (GCD) from the perspective of mining underlying relationships between known and unknown categories. To implement this idea, we propose a dynamic conceptional contrastive learning framework to alternately explore latent conceptional relationships and perform conceptional contrastive learning. This mechanism enables models to learn more discriminable representations. Furthermore, to mitigate the inconsistency of updating conception representations during the training process, we propose a conception-level momentum update to facilitate the model toward better optimization. Extensive experimental results show that our DCCL achieves a new state-of-the-art performance on GCD tasks.

Acknowledgement This work has been supported by the EU H2020 project AI4Media (No. 951911) and by the PRIN project CREATIVE (Prot. 2020ZSL9F9).

References

[1] Sugato Basu, Arindam Banerjee, and Raymond J. Mooney. Semi-supervised clustering by seeding. In Claude Sammut and Achim G. Hoffmann, editors, *ICML*, 2002. 3

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

[3] Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning. In *ICLR*, 2022. 2

[4] 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, 7

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. Simclr: A simple framework for contrastive learning of visual representations. In *International Conference on Learning Representations*, 2020. 2

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

[7] 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. 5, 6

[8] 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. 8

[9] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *ICCV*, 2021. 1, 2, 6

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

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

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1, 2, 5

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

[14] Jiabo Huang and Shaogang Gong. Deep clustering by semantic contrastive learning. In *BMVC*, 2022. 2

[15] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 2019. 6

[16] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCVW*, 2013. 5, 6

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5, 6, 8

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 2017. 1

[19] Tilman Lange, Martin HC Law, Anil K Jain, and Joachim M Buhmann. Learning with constrained and unlabelled data. In *CVPR*, 2005. 3

[20] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021. 2, 5

[21] J MacQueen. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, 1967. 2, 8

[22] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Contrastive learning for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

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

[24] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012. 5, 6, 7

[25] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 1999. 4

[26] Martin Rosvall, Daniel Axelsson, and Carl T Bergstrom. The map equation. *The European Physical Journal Special Topics*, 2009. 4

[27] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the national academy of sciences*, 2008. 3, 4, 8

[28] Subhankar Roy, Mingxuan Liu, Zhun Zhong, Nicu Sebe, and Elisa Ricci. Class-incremental novel class discovery. In *ECCV*, 2022. 1

[29] Saquib Sarfraz, Vivek Sharma, and Rainer Stiefelhagen. Efficient parameter-free clustering using first neighbor relations. In *CVPR*, 2019. 8

[30] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE TPAMI*, 2012. 1

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1

[32] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *NeurIPS*, 2017. 1

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

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

[35] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *CVPR*, 2022. 1, 2, 4, 5, 6, 7, 8

[36] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5, 6, 7, 8

[37] Xiaohan Wang, Linchao Zhu, and Yi Yang. T2vlad: global-local sequence alignment for text-video retrieval. In *CVPR*, 2021. 2

[38] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 2020. 1

[39] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *Arxiv*, 2021. 3

[40] Li Yi, Sheng Liu, Qi She, A Ian McLeod, and Boyu Wang. On learning contrastive representations for learning with noisy labels. In *CVPR*, 2022. 1

[41] Hong-Xing Yu and Wei-Shi Zheng. Weakly supervised discriminative feature learning with state information for person identification. In *CVPR*, 2020. 1

[42] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *ICCV*, 2019. 1

[43] Shuai Zhao, Xiaohan Wang, Linchao Zhu, and Yi Yang. Slimmable networks for contrastive self-supervised learning. *arXiv*, 2022. 2

[44] Yuyang Zhao, Zhun Zhong, Nicu Sebe, and Gim Hee Lee. Novel class discovery in semantic segmentation. In *CVPR*, 2022. 1

[45] Mingkai Zheng, Fei Wang, Shan You, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Weakly supervised contrastive learning. In *ICCV*, 2021. 1

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

[47] 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. 1