MetaGCD: Learning to Continually Learn in Generalized Category Discovery

Yanan Wu[1,2*], Zhixiang Chi[3*], Yang Wang[4], Songhe Feng[1,2 †]
[1]Key Laboratory of Big Data & Artificial Intelligence in Transportation,
Ministry of Education, Beijing Jiaotong University, Beijing, 100044, China
[2]School of Computer and Information Technology, Beijing Jiaotong University, Beijing, 100044, China
[3]Department of Electrical and Computer Engineering, University of Toronto, Toronto, M5G1V7, Canada
[4]Department of Computer Science and Software Engineering,
Concordia University, Montreal, H3G2J1, Canada
{ynwu0510,shfeng}@bjtu.edu.cn, zhixiang.chi@mail.utoronto.ca, yang.wang@concordia.ca

Abstract

In this paper, we consider a real-world scenario where a model that is trained on pre-defined classes continually encounters unlabeled data that contains both known and novel classes. The goal is to continually discover novel classes while maintaining the performance in known classes. We name the setting Continual Generalized Category Discovery (C-GCD). Existing methods for novel class discovery cannot directly handle the C-GCD setting due to some unrealistic assumptions, such as the unlabeled data only containing novel classes. Furthermore, they fail to discover novel classes in a continual fashion. In this work, we lift all these assumptions and propose an approach, called MetaGCD, to learn how to incrementally discover with less forgetting. Our proposed method uses a meta-learning framework and leverages the offline labeled data to simulate the testing incremental learning process. A meta-objective is defined to revolve around two conflicting learning objectives to achieve novel class discovery without forgetting. Furthermore, a soft neighborhood-based contrastive network is proposed to discriminate uncorrelated images while attracting correlated images. We build strong baselines and conduct extensive experiments on three widely used benchmarks to demonstrate the superiority of our method.

1. Introduction

Object categories in real-world environments are dynamically evolving and expanding over time. However, conventional deep learning-based visual recognition methods normally focus on closed-world scenarios with pre-defined categories [19, 37]. Such systems are brittle when deployed to an ever-changing realistic open-world setting, where object instances may come from new categories. In contrast, recognizing the known categories and utilizing them to discern the unknowns are intrinsic to human perception.

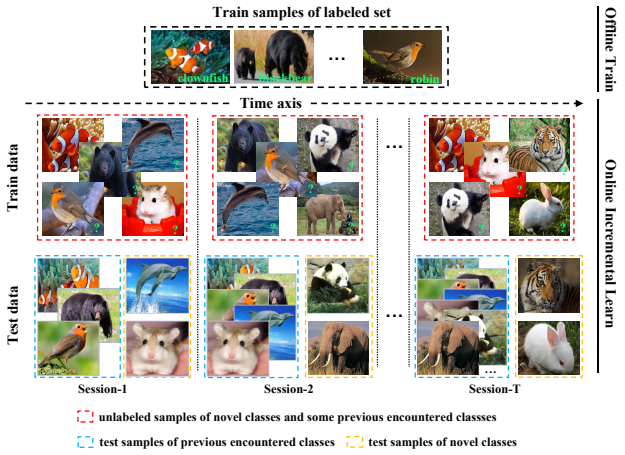

Figure 1: **Illustration of our C-GCD setting**. During the offline training, we learn an initial model based on training samples of the labeled set. During each subsequent online incremental learning, we are given some unlabeled images belonging to both known and novel classes. Our goal is to update the model in each incremental session so that the model can maintain the performance on old classes while discovering novel classes.

Recently, discovering the novel classes among unlabeled data has been an active area of research [15, 11, 40, 35, 45]. However, most prior works make several assumptions that are unrealistic in practice. For example, the works in [15, 11, 40] assume the co-existence of both labeled data (with known classes) and unlabeled data (contains potential unknown classes to be discovered) at the training phase and

*The authors contributed equally to this work.
†Corresponding Author

the models are learned from scratch. This leads to repetitive large-scale training every time when new classes are expected to be discovered. The works in [15, 11, 35, 45] assume the newly encountered unlabeled data only belongs to the novel classes. This is unrealistic in practice. To meet such conditions, a rigorous filtering method is needed to precisely filter out known class data to avoid degenerate solutions. Due to these limitations, none of these works can be used to build recognition systems that can deal with evolving object categories sequentially over a long time horizon.

In this paper, we consider a more flexible setting for real-world applications. Let us consider the application of home robots. The robots are equipped with an offline trained object recognition model on pre-defined categories during manufacturing. After deployment, the robots are expected to operate in diverse environments. While operating, they continually receive data that belongs to known and possibly unknown classes. Ideally, we would like the robots to continually discover and learn novel classes from such data. We dub such a setting as ***Continuous Generalized Class Discovery (C-GCD)***. As shown in Fig. 1, C-GCD has two phases: 1) an offline training phase that allows the model to be trained on large-scale labeled data with pre-defined classes; 2) when the model is deployed, it continually encounters unlabeled data that comes from both known and novel classes on a *longer horizon*. At each incremental session, the data from the previous sessions is *inaccessible*. The model needs to precisely classify the known classes and discover novel ones to expand its knowledge base. Obviously, the main challenge of C-GCD is to discover the novel classes among *unlabeled* images that contain both *known* and *unknown* categories while maintaining the performance on *old* classes. However, learning novel knowledge normally leads to notorious catastrophic forgetting [7], which further exacerbates the model performance.

There are some initial attempts on C-GCD [21, 45]. However, they only consider C-GCD at the deployment stage mentioned above. The offline training stage is not fully exploited in these works. Concretely, the labeled data during offline training is only used for pre-training model *representations*. Therefore, the model at the offline stage is unaware of its subsequent learning duty (discover novel classes and retain the performance of known classes) [8] and is also prone to overfit to the labeled set [40]. Such learning objective misalignment leads to cumbersome heuristic strategies to facilitate the new learning task while keeping the previous knowledge. For example, to learn the novel classes, [21] requires a self-labeling method, which may cause error propagation. A routing strategy is also required to determine the known and novel classifier heads. [45] relies on a thresholding method to filter novel instances. However, the overall robustness of the method can be sensitive to the threshold. To alleviate the forgetting issues, [21, 45] propose to distill the knowledge from the pre-trained base models. Data replay is also utilized to either directly select representative labeled examplars [45] or generate pseudo-latent representations from them [21]. Consequently, the base models and the replay buffers have to be stored locally which may cause storage problems, especially in a resource-constraint environment.

In this work, we propose a fully learning-based solution, named MetaGCD, to minimize the hand-engineered heuristics in prior works. Concretely, at the offline training phase, instead of pre-training a model *representation*, we directly train an *initialization* that is learned to discover novel categories with less forgetting when deployed. It is realized by meta-learning-based bi-level optimization [12] to couple the offline training and downstream learning objectives. During the offline training, we simulate the testing scenario and construct pseudo incremental novel class discovery sessions using the labeled data. At each incremental session, we discover novel classes by updating the model using an unsupervised contrastive loss. The meta-objective is then defined by validating the updated model on *all* classes encountered on a labeled pseudo test set. Therefore, the meta-objective of the offline training is aligned with the evaluation protocol at deployment. It enforces the model to learn to balance two conflicting objectives, namely discovering new objects and not-forgetting old objects. The meta-objective also reinforces the unsupervised updated model to be supervised by the true labels to ensure valid novel class discovery.

MetaGCD uses unsupervised contrastive learning to explore the relationship among instances for novel class discovery. Therefore, it is less prone to label overfitting [40]. However, we observe that the negative pairs in contrastive learning normally dominate the loss function. So we further propose soft neighborhood contrastive learning to mine more positiveness. Concretely, for each image instance, we select the nearest candidate neighbors within the batch to treat them as soft positive samples to contribute to the discriminative feature learning. Overall, our contributions are summarised as follows:

- We consider a realistic setting C-GCD for applications in real-world scenarios. It allows the model trained on pre-defined classes to continually explore novel classes through incoming unlabeled data while simultaneously keeping the performance of known classes.
- We propose a meta-learning approach where the learning objective is well aligned with the evaluation protocol during testing. It directly optimizes the model to achieve novel object discovery without forgetting.
- A soft neighborhood contrastive learning method is also proposed to mine more soft positive pairs to elevate the discovery capability.
- We establish strong baselines and show that our method achieves superior performance with less hand-

engineered design through extensive experiments.

2. Related Work

Discovering novel classes. *Novel Class Discovery* (NCD) aims to discover the novel classes from unlabeled data by utilizing the prior knowledge from the labeled data [15, 11, 17]. However, NCD assumes that the unlabeled data only belongs to the novel classes, which is unrealistic. Alternatively, a *generalized* version of NCD (GCD) [40] relaxes such constrain. Although GCD allows the unlabeled data to contain both known and novel classes, they are both required to be present in the training phase. It leads to repetitive large-scale training when different groups of unlabeled data are continually presented to the recognition system. Recently, a class incremental variant of NCD (class-iNCD) is proposed to learn the tasks of labeled known and unlabeled novel classes sequentially [35]. When learning the novel classes, the data of old classes are inaccessible. In the end, the model is evaluated on all encountered classes. Nevertheless, only a few incremental sessions containing unlabeled novel classes are allowed in class-iNCD. This limitation hinders its applicability under the realistic setting with continually evolved object categories. Our proposed C-GCD alleviates the above limitations in real-world scenarios. Our approach can learn from labeled pre-defined classes during offline training, and then continuously encounter unlabeled data with both known and novel classes after deployment. Our model will learn to discover novel classes without forgetting old classes. C-GCD is also related to the classic class-incremental setting [26, 41, 31, 29]. But C-GCD is more challenging as the newly evolved classes are unlabeled and an automatic class discovery mechanism is required [21].

Meta-learning. Existing meta-learning methods can be categorized into: 1) model-based [36, 2, 4]; 2) Optimization-based [34, 12, 5]; and 3) metric-based [38]. Typical meta-learning methods utilize bi-level optimization to train a model that is applicable for downstream adaptations. Our work is built upon MAML [12], which trains a model initialization through episodes of tasks for fast adaptation via gradient updates. Such learning paradigm has been widely applied in different vision tasks, such as test-time adaptation [39, 28, 32, 30], continual learning [43, 20, 42] and domain shift [44, 46, 13]. In our case, the adaptation is achieved in an unsupervised manner, and the bi-level optimization is utilized to combine two conflicting learning objectives: discovering the novel classes without forgetting the old classes.

Contrastive learning. Contrastive learning has been popular in self-supervised visual representation learning [1, 6, 18, 33, 27]. It explores the relationships among data instances by constructing positive and negative pairs. Therefore, the overfitting on the label space is reduced to improve the generalization of downstream tasks. Zhong *et al.* [47] apply contrastive learning to discover novel classes by exploring the data neighborhood and choosing pseudo-positive pairs. However, those pseudo-positive pairs contribute equally regardless of their closeness compared to the reference sample. In this work, we introduce the soft positiveness concept to allow adaptive contribution.

3. The Proposed Method

Problem definition. The goal of C-GCD is to have the offline trained model continually discover and learn novel object classes from unlabeled data containing both known and novel classes. We define a sequence of T learning sessions $\{\mathcal{S}^0, \mathcal{S}^1, \cdots, \mathcal{S}^T\}$. Let $x^t \in \mathcal{X}^t$ and $y^t \in \mathcal{Y}^t$ denote the input and label space at session t. We represent each session as: $\mathcal{S}^0 = \{(\mathbf{x}_i^0, \mathbf{y}_i^0)\}_{i=1}^{Z_0}$ and $\mathcal{S}^t = \{(\mathbf{x}_i^t)\}_{i=1}^{Z_t}$. Note, only the first session (*i.e.*, $t = 0$) contains large-scale labeled samples. As for $t > 0$, $\mathcal{S}^t$ only contains *unlabeled* data. At the t^{th} session, only $\mathcal{S}^t$ is accessible, and the incoming data belongs to both learned *known* class from previous sessions and *novel* classes. Therefore, we can denote $\mathcal{Y}^t = \mathcal{Y}^{t-1} \cup \mathcal{Y}_n^t$, where $\mathcal{Y}_n^t$ represents the *novel* classes to be discovered at session t. After learning on $\mathcal{S}^t$, the model is evaluated on all test images accumulated until session t to test the performance on $\mathcal{Y}^{t-1}$ (ideally the model should not forget old classes) and the discovery capability on $\mathcal{Y}_n^t$. Compared with previous works [15, 11, 35, 45], C-GCD is much more challenging due to several factors. First, the unlabeled data contains both *known* and *unknown* classes, *i.e.*, $\mathcal{Y}^t = \mathcal{Y}^{t-1} \cup \mathcal{Y}_n^t$. Second, labeled data is absent at $t > 0$, *i.e.*, $\mathcal{S}^0 \cup \mathcal{S}^t = \varnothing$ where $t > 0$. Finally, since C-GCD operates on a long horizon, *i.e.*, $t \gg 1$, the catastrophic forgetting issue is more severe.

Method overview. Fig. 2 shows an overview of MetaGCD. Following [40], we use a model without parametric classification heads since it is more suitable for dealing with novel classes. Novel class discovery is performed by directly clustering the feature spaces and class labels are assigned through the classic k-means algorithm. Concretely, we learn a model initialization using the labeled data during offline training. During each continual learning session, we update the model using a soft neighborhood contrastive learning (see Fig. 2 (a)) on unlabeled data. To fully exploit the labeled data in offline learning, we further develop a bi-level optimization based on meta-learning to simulate the online learning scenario, so that the model is ready to adapt to new incoming unlabeled data and discover novel objects after the offline training (see Fig. 2 (b)). In the following, we describe these two parts of our method in detail.

3.1. Contrastive learning based clustering network

Considering the characteristics of labeled and unlabeled data, we employ different contrastive learning strategies. To

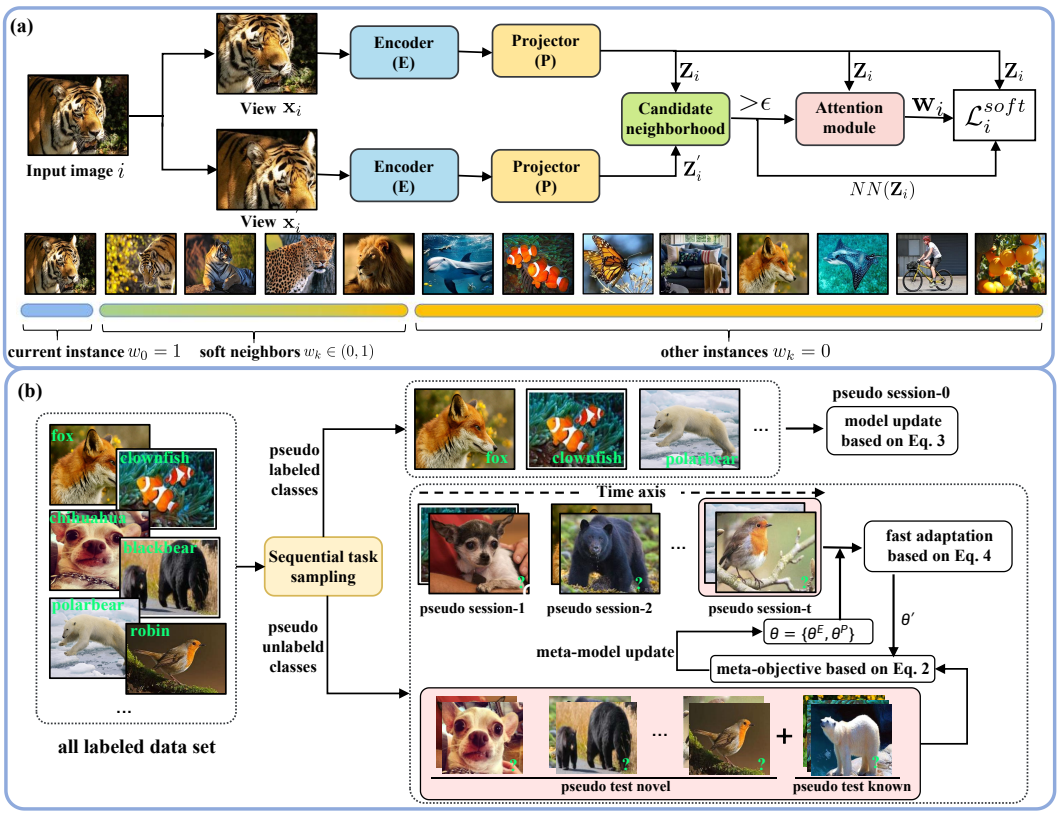

Figure 2: **Overview of the proposed MetaGCD**. (a) Our soft neighborhood contrastive learning network aims to discriminate uncorrelated instances while absorbing correlated instances to learn discriminative representations. (b) Our meta-learning optimization strategy utilizes the offline labeled data to simulate the testing incremental learning process by sampling sequential learning tasks. By learning from these sampled sequential tasks, our model learns a good initialization, so that it can effectively adapt to discover new novel classes without forgetting old classes.

train on the labeled data, we utilize a combination of unsupervised and supervised contrastive losses. When discovering the latent classes in continually encountered unlabeled data, we propose to mine soft positive neighbors for each data instance to elevate the discriminative feature learning.

3.1.1 Representation learning on labeled data

To learn a robust and semantically meaningful representation on labeled data, we utilize both self-supervised [14] and supervised [22] contrastive losses. Let $\mathbf{x}_i$ and $\mathbf{x}_i'$ be two randomly augmented versions of i^{th} instance sample, the unsupervised contrastive loss is expressed as:

$$\mathcal{L}_i^{ucl} = -\log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_i'/\tau)}{\sum_n \mathbb{I}_{[n \neq i]} \exp(\mathbf{z}_i \cdot \mathbf{z}_n/\tau)} \quad (1)$$

where $\mathbf{z}_i = \phi(f(\mathbf{x}_i))$, $\mathbb{I}_{[n \neq i]}$ is an indicator function, and τ is a temperature value. f is the feature extractor, and ϕ is a multi-layer perceptron (MLP) projection head.

The supervised contrastive counterpart is expressed as:

$$\mathcal{L}_i^{scl} = -\frac{1}{|\mathcal{N}(i)|} \sum_{q \in \mathcal{N}(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_q/\tau)}{\sum_n \mathbb{I}_{[n \neq i]} \exp(\mathbf{z}_i \cdot \mathbf{z}_n/\tau)} \quad (2)$$

where $\mathcal{N}(i)$ denotes the indices of instances having the same label as $\mathbf{x}_i$ within the batch. Finally, these two losses are weighted by λ to train on the labeled data:

$$\mathcal{L}_{labeled} = (1-\lambda) \sum_{i \in B} \mathcal{L}_i^{ucl} + \lambda \sum_{i \in B} \mathcal{L}_i^{scl} \quad (3)$$

3.1.2 Soft neighborhood contrastive learning on unlabeled data

When learning on unlabeled data, only the unsupervised contrastive loss Eq. 1 can be used. However, the samples within the same class could be mistakenly treated as negatives due to the missing labels. In addition, the number of negative pairs significantly surpasses positive pairs. Such imbalanced loss contribution could be sub-optimal. Aligning the positive and negative pairs with the true classes

emerges as a desired solution. [47] has attempted to address the limitations by mining more positive pairs in the neighbored of each data sample. However, the pseudo-positive pairs are treated equally, regardless of how close they are to that data sample. To address this issue, we propose to encode soft positive correlation among instance neighbors to achieve adaptive contribution, as shown in Fig. 2 (a).

Specifically, for each $\mathbf{x}_i$, we first use the nearest neighbor operator on the projected features to select candidate neighbors. We denote them as $NN(\mathbf{z}_i)_k$ with k as the index. We then pass them and $\mathbf{z}_i$ to an attention module to predict a set of positiveness values $\mathbf{w}_i = \{w_{ik}\} \in (0, 1)$ to weight the contribution of $NN(\mathbf{z}_i)_k$ to the loss. Accordingly, the soft neighborhood contrastive loss is defined as:

$$\mathcal{L}_i^{soft} = -\frac{1}{|NN(\mathbf{z}_i)|} \sum_{k \in NN(\mathbf{z}_i)} \log \frac{w_{ik} \cdot \exp(\mathbf{z}_i \cdot \mathbf{z}_k/\tau)}{\sum_n \mathbb{I}_{[n \neq i]} \exp(\mathbf{z}_i \cdot \mathbf{z}_n/\tau)} \tag{4}$$

Candidate neighborhood. For each batch of data, we first compute their features $\mathcal{F}$ from the projection head at once. For each reference view $\mathbf{x}_i$, we retrieve nearest neighbors by comparing the cosine similarity to a threshold ϵ as:

$$NN(\mathbf{z}_i) = \{\mathbf{F}\}, \text{ for } \mathbf{F} \text{ in } \mathcal{F}, \text{ if } \cos(\mathbf{z}_i, \mathbf{F}) \geq \epsilon \tag{5}$$

where $\mathbf{z}_i$ and $\mathbf{F} \in \mathcal{F}$ are normalized before computation.

Positiveness generation. Fig. 2 (a) shows an intuitive example of the candidate neighbors. The first two neighbors belong to the same category as the reference '*tiger*' sample, while the 3^{rd} and 4^{th} neighbors are partially related (*i.e.*, they belong to the '*lion*' and '*leopard*' categories, but not '*tiger*'). The remaining instances are not related to '*tiger*'. Therefore, the first four instances tend to be selected and they should contribute adaptively to the loss. We propose to learn an attention module to measure the soft correlations between the selected neighbors and the reference instance (instead of the binary form in [47]). Given two inputs $\mathbf{z}_i$ and $NN(\mathbf{z}_i)_k$, we can calculate the positiveness score as:

$$\mathbf{w}_i = \text{Softmax}[f_1(\mathbf{z}_i) \times f_2(NN(\mathbf{z}_i)_k)^T] \tag{6}$$

where $f_1(\cdot)$ and $f_2(\cdot)$ are the new projection layers and $\times$ denotes the cross attention operator. Eq. 6 is then normalized so that $\mathbf{w}_i$ has a max value as 1. Note that $f_1(\cdot)$ and $f_2(\cdot)$ can also be the non-parametric identity mappings, which are empirically found to be more effective. This observation may be attributed to the self-supervised learning paradigm, where the objective is to train the encoder effectively. Simplifying the attention module leads to less overfitting and improves learning attentive features.

3.2. Learning to incrementally discover categories

The main limitation of the prior works is that the labeled set $\mathcal{S}^0$ is not fully exploited [7]. Instead of performing only representation learning on $\mathcal{S}^0$ [21, 45], we borrow the meta-learning paradigm (in particular MAML [12]) to learn how to continually discover new classes. In few-shot learning, during meta-training, MAML constructs few-shot tasks to mimic the meta-testing scenario to achieve learning to quickly adapt. In our C-GCD case, the online continual class discovery tasks can be viewed as the "meta-testing" stage. Therefore, we propose to simulate the continual setting using $\mathcal{S}^0$ during offline training, as shown in Fig. 2 (b). We aim to produce a model *initialization* that is trained by aligning the training and evaluation objectives so that it is endowed with the capability to effectively discover novel classes with less forgetting during evaluation.

Algorithm 1 The optimization procedure of MetaGCD

Require: α, β, γ: learning rates
Require: $\mathcal{S}^0$: training set of *labeled* classes
randomly initialize parameters θ
while not converged **do**
  $\mathcal{D} = \{(\mathcal{D}_{tr}^j, \mathcal{D}_{te}^j)\}_{j=0}^T$
    ▷ sample a pseudo incremental sequence
  $\mathcal{P} = \varnothing$ ▷ empty cumulative pseudo test set
  $\theta^{E,P} \leftarrow \theta^{E,P} - \gamma \nabla_{\theta^{E,P}} \mathcal{L}_{labeled}(\mathbf{x}_{tr}^0, \mathbf{y}_{tr}^0; \theta)$
    ▷update parameters using pseudo labeled classes
  $\mathcal{P} = \mathcal{P} \cup \mathcal{D}_{te}^0$ ▷ accumulate test set of sess-0
  for $j = 1, \cdots, T$ **do**
    $\tilde{\theta}^{E,P} = \theta^{E,P} - \alpha \nabla_{\theta^{E,P}} \mathcal{L}_{soft}(\mathbf{x}_{tr}^j; \theta)$
    ▷ compute adapted params with unlabeled samples
    $\mathcal{P} = \mathcal{P} \cup \mathcal{D}_{te}^j$ ▷accumulate test set of sess-j
    $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{(\mathcal{X},\mathcal{Y}) \in \mathcal{P}} \mathcal{L}_{scl}(\mathcal{X}, \mathcal{Y}; \tilde{\theta}^E, \tilde{\theta}^P)$
      ▷ update meta-params θ to new session
  end for
end while

Sequential task sampling. To mimic the evaluation process, we sample sequential learning tasks from $\mathcal{S}^0$ [7]. Specifically, we first randomly separate the $\mathcal{S}^0$ into *pseudo labeled* and *pseudo unlabeled classes* without overlapping. Next, we sample a sequence of $T+1$ sessions, $\mathcal{D} = \{(\mathcal{D}_{tr}^j, \mathcal{D}_{te}^j)\}_{j=0}^T$, where $\mathcal{D}_{tr}^j$ and $\mathcal{D}_{te}^j$ are the training and test set for the j^{th} session. For the training splits $\{\mathcal{D}_{tr}^j\}$, we follow the evaluation protocol to only allow the first session to contain labeled data, (*i.e.*, $\mathcal{D}_{tr}^0 = \{\mathbf{x}_{tr}^0, \mathbf{y}_{tr}^0\}$) and the rest with unlabeled data (*i.e.*, $\mathcal{D}_{tr}^j = \{\mathbf{x}_{tr}^j\}$, for $j > 0$). We also set the first session to contain a larger number of samples, *i.e.*, $\left|\mathcal{D}_{tr}^0\right| \gg \left|\mathcal{D}_{tr}^{j>0}\right|$ to simulate the C-GCD setting where the model is first trained during an offline training stage with a large amount of data. For the test splits $\{\mathcal{D}_{te}^j\}$, all of them contain labels that will be used during the optimization (*i.e.*, $\mathcal{D}_{te}^j = \{\mathbf{x}_{te}^j, \mathbf{y}_{te}^j\}$, $\forall j$). Note that $\mathcal{D}_{te}^j$ only contains the test data belonging to the current session j.

Meta-training. For each sampled sequence $\mathcal{D}$, we let the model continually explore the incoming unlabeled data in

an unsupervised manner. To reduce the forgetting issue due to learning new knowledge, we utilize the bi-level optimization [12, 8, 7] to directly formulate incrementally discovering without forgetting as the meta-objective. The meta-learning procedure is illustrated in Alg. 1 and Fig. 2 (b). Concretely, we decouple the network as $\theta = \{\theta^E, \theta^P\}$, where θ^E and θ^P are the encoder and projection layers. At each incremental session, we aim to evaluate all the classes that have encountered so far. Hence, at the beginning of each sequence, we define an empty cumulative pseudo test set $\mathcal{P}$ to store the test samples from previous sessions. After that, we first train θ on the pseudo labeled classes ($j = 0$) using the unsupervised and supervised contrastive loss (Eq. 3). At each j^{th} session ($j > 0$), we update θ on unlabeled samples $\mathcal{D}_{tr}^j = \{\mathbf{x}_{tr}^j\}$ via a few gradient steps:

$$\tilde{\theta}^{E,P} = \theta^{E,P} - \alpha \nabla_{\theta^{E,P}} \mathcal{L}_{soft}(\mathbf{x}_{tr}^j; \theta) \tag{7}$$

where $\mathcal{L}_{soft}(\cdot)$ is the proposed soft neighborhood contrastive loss (Eq. 4). It aims to discriminate uncorrelated samples while absorbing correlated ones. By thoroughly exploring the unlabeled data, it maintains comprehensive old knowledge while efficiently discovering novel classes.

Eq. 7 mimics how the model discovers novel classes on the incoming unlabeled data at test-time. Ideally, we like the adapted $\tilde{\theta}^{E,P}$ to perform well on all encountered classes. The test data from previous sessions and the current session separately reflect the catastrophic forgetting robustness and novel class discovery capability. Thus, we append $\mathcal{D}_{te}^j$ to $\mathcal{P}$. Accordingly, the meta-objective is defined as follows for the outer loop of the meta-level optimization:

$$\min_{\theta^E, \theta^P} \sum\nolimits_{(\mathcal{X}, \mathcal{Y}) \in \mathcal{P}} \mathcal{L}_{scl}(\mathcal{X}, \mathcal{Y}; \tilde{\theta}^E, \tilde{\theta}^P) \tag{8}$$

where $\mathcal{L}_{scl}(\cdot)$ is the supervised contrastive loss in Eq. 1. Note that the optimization is performed on θ, although $\mathcal{L}_{scl}(\cdot)$ is a function of $\tilde{\theta}^E$ and $\tilde{\theta}^P$. The meta-objective in Eq. 8 is then optimized using gradient descent, as shown in Line 13 of Alg. 1. We empty $\mathcal{P}$ when all $T + 1$ sessions are iterated. After meta-training, we obtain an initialization of the model θ which has been specifically trained to discover and learn novel objects from a sequence of unlabeled data.

Meta-testing. It is worth mentioning that the procedure in Alg. 1 aligns with the evaluation protocol. After discovering novel classes at each incremental session, the model is evaluated on all encountered classes. Our meta-objective optimizes the model towards what it is supposed to do at evaluation to maximize the performance. In addition, despite some uncertainties that may occur for unsupervised learning, the model is constrained by a fully supervised meta-objective. Thus, when training converges, the meta-model θ is ready to discover novel classes while maintaining the old knowledge by only running Line 10 of Alg. 1.

Dataset	Labeled Set		Unlabeled Set	
	#class	#image	#class	#image
CIFAR10	7	28000	10	22000
CIFAR100	80	32000	100	18000
Tiny-ImageNet	150	60000	200	40000

Table 1: Datasets used in our experiments. We show the number of classes in the labeled and unlabeled sets, as well as the number of samples.

4. Experiments

4.1. Dataset and setup

Dataset. We construct the C-GCD benchmark using three widely used datasets as in NCD [40, 35, 47], *i.e.*, CIFAR10 [23], CIFAR100 [23] and Tiny-ImageNet [25]. Each dataset is split into two subsets, 1) large-scale labeled samples accounting for 80% of the known classes data constitute a labeled set for offline training; and 2) the remaining data containing known and novel classes are used as an unlabeled set for continual object discovery. In Tab. 1, we summarize the dataset splits used in our training.

Session-wise data split. All labeled samples $\mathcal{S}^0$ are used for offline training in our setting. During the online incremental learning stage, the unlabeled samples are dynamically added (*i.e.*, sessions $t \geq 1$). Specifically, CIFAR10 is divided into 3 incremental sessions. In the $t^{th}(t{>}0)$ session, 3000 unlabeled images from 1 novel class and 2000 unlabeled images from $7 + (t-1) \times 1$ known classes are added. CIFAR100 is divided into 4 sessions, in which 1500 unlabeled images from 5 novel classes and 2000 unlabeled images from $80 + (t-1) \times 5$ known classes are added in the t^{th} session. The Tiny-ImageNet consists of 5 incremental sessions, each containing 3000 unlabeled images from 10 novel classes and 3000 unlabeled images from $150 + (t-1) \times 10$ known classes.

Sequential task sampling. During offline training, we use $\mathcal{S}^0$ to sample sequential tasks. We first split $\mathcal{S}^0$ into non-overlapping *pseudo labeled* and *novel* classes (4/3 for CIFAR10, 60/20 for CIFAR100 and 100/50 for Tiny-ImageNet). For each task, the *pseudo labeled* set is first used to warm up the model , followed by T incremental sessions of unlabeled samples containing the *pseudo labeled* and *novel* classes. Both the session number and the number of novel classes in each offline incremental session are consistent with the online incremental learning scenario.

Evaluation metrics. After learning the model on unlabeled samples at every online incremental stage, we follow [40] to measure the clustering accuracy between the ground truth labels y_i and the model's predictions $\hat{y}_i$ as:

$$ACC = \max_{p \in \mathcal{P}(\mathcal{Y})} \frac{1}{N} \sum (\mathbb{1}\{y_i = p(\hat{y}_i)\}), \tag{9}$$

Methods	**CIFAR10** (Session Number)									Final Impro.		
	1			2			3					
	All	Old	New	All	Old	New	All	Old	New	All	Old	New
RankStats	69.31	70.20	58.63	65.23	67.86	51.20	38.16	50.01	35.94	+54.50	+47.22	+48.77
FRoST	73.92	81.17	66.45	69.56	79.73	58.04	67.73	70.84	51.13	+24.93	+26.39	+33.58
VanillaGCD	89.24	97.97	81.80	85.13	96.67	74.60	86.41	95.03	76.75	+6.25	+2.20	+7.96
GM	90.00	98.41	77.40	87.39	**99.01**	73.46	87.86	97.15	78.93	+4.80	+0.08	+5.78
MetaGCD(ours)	**95.38**	**99.07**	**89.15**	**93.34**	98.81	**85.39**	**92.66**	**97.23**	**84.71**			

Methods	**CIFAR100** (Session Number)												Final Impro.		
	1			2			3			4					
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New
RankStats	62.33	64.22	31.60	55.01	58.55	26.85	51.77	56.70	25.47	47.51	54.59	17.20	+27.05	+23.01	+43.93
FRoST	67.14	68.57	50.73	67.01	68.82	52.60	62.35	65.48	45.67	55.84	59.06	42.95	+18.72	+18.54	+18.18
VanillaGCD	76.78	77.91	58.60	73.67	75.29	60.70	72.77	74.72	62.33	71.44	74.75	58.20	+3.12	+2.85	+2.93
GM	78.29	**79.91**	66.00	77.58	**79.64**	61.13	74.56	77.60	58.14	72.02	75.98	56.32	+2.54	+1.62	+4.81
MetaGCD(ours)	**78.96**	79.36	**72.60**	**78.67**	79.41	66.81	**76.06**	**78.20**	**64.87**	**74.56**	**77.60**	**61.13**			

Methods	**Tiny-ImageNet** (Session Number)															Final Impro.		
	1			2			3			4			5					
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New	All	Old	New
RankStats	62.39	64.54	35.01	55.89	52.23	34.20	49.88	46.17	28.33	44.20	42.87	24.50	36.09	35.20	15.76	+34.15	+36.33	+42.7
FRoST	64.92	67.84	46.28	59.50	61.86	40.60	57.86	60.63	39.14	55.68	59.71	36.55	50.49	53.76	33.37	+19.75	+17.77	+29.09
VanillaGCD	75.92	78.17	62.15	74.53	77.73	56.12	73.64	74.85	57.31	70.69	71.13	54.35	66.15	67.17	54.43	+4.09	+4.36	+4.03
GM	76.32	**79.55**	63.60	75.43	78.10	57.40	72.63	76.29	54.80	70.54	**76.80**	51.50	67.31	**72.08**	50.90	+2.93	-0.55	+7.56
MetaGCD(ours)	**78.67**	79.41	**66.80**	**77.89**	**79.95**	**61.40**	**75.23**	**77.86**	**61.20**	**72.00**	75.61	**57.55**	**70.24**	71.53	**58.46**			

Table 2: **Performance (in %) comparisons with the state-of-the-art methods on CIFAR10, CIFAR100, Tiny-ImageNet datasets**. The results of other methods are obtained by running their released codes under the C-GCD setting.

where N is the total number of test samples and $\mathcal{P}(\mathcal{Y})$ is the set of all permutations of the class labels $\mathcal{Y}$ encountered so far. The optimal permutation can be obtained via the Hungarian algorithm [24]. Our main metric is ACC on 'All' classes, indicating the accuracy across all accumulated test sets so far. To decouple the evaluation on *forgetting* and *discovery*, we further report accuracy for both the 'Old' classes subset (samples in the test set belonging to previous known classes) and 'New' classes subset (samples in the test set belonging to novel classes).

Implementation details. Following [40], we employ a vision transformer (ViT-B-16) [10] pretrained on ImageNet [9] with DINO [3] as the feature extractor throughout the paper. We use the Adam optimizer and the learning rates in Alg. 1 are set as $\gamma = 0.1$, $\alpha = 0.001$ and $\beta = 0.0001$. We use a batch size of 256 and $\lambda = 0.35$ to balance the losses in Eq. 3. Unless otherwise stated, we select the threshold ϵ to be 0.85 in Eq. 5. At the meta-training stage, we first perform training on *pseudo labeled* set for 50 epochs, followed by 10 inner and 1 outer gradient updates for incremental sessions. At the meta-test stage, we directly perform 20 gradient updates to adapt using *unlabeled* samples. Furthermore, we follow standard practice in self-supervised learning to use the same projection head as in [3] and discard it at test-time.

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

Since this paper considers a new problem setting, there is no prior work that we can directly compare. Nevertheless, we choose SOTA methods on NCD and run their codes under our C-GCD setting, including RankStats [16], VanillaGCD [40], and recent continual NCD models FRoST [35], GM [45]. Both RankStats and FRoST train two classifiers on top of a shared feature representation. The first head is fed images from the labeled set and is trained with the cross-entropy loss, while the second head sees only images from unlabeled images of novel classes. In order to adapt RankStats and FRoST to C-GCD, we train them with a single classification head for the total number of classes in the dataset. The sequential version of VanillaGCD is adopted and serves as the baseline for our model. We leverage the original training mechanism for GM.

In Tab. 2, we report the *All*/*Old*/*New* class accuracy per incremental session for all methods, and the relative improvement for the final session. As we can see, the proposed method consistently outperforms all other methods on all three datasets among the most incremental sessions. Specifically, our MetaGCD surpasses the most recent method GM by 5.78%, 4.81% and 7.56% on CIFAR10, CIFAR100 and Tiny-ImageNet datasets for the final *New* classes accuracy.

Methods	**CIFAR100** (Session Number)												Average Acc		
	1			2			3			4					
	All	Old	New	All	Old	New	All	Old	New	All	Old	New	mA	mO	mN
Baseline	76.78	77.91	58.60	73.67	75.29	60.70	72.77	74.62	62.33	71.44	74.75	58.20	73.67	75.64	59.96
+$\mathcal{L}_{CN}$	77.29	78.53	65.44	75.20	77.67	63.08	74.55	75.91	63.63	72.62	75.43	59.40	74.92	76.89	62.89
+$\mathcal{L}_{SP}$	77.92	78.45	68.78	76.53	78.22	64.71	74.86	77.24	64.09	73.34	76.87	60.60	75.66	77.70	64.55
+ Meta-learning	78.96	79.36	72.60	78.67	79.41	66.81	76.06	78.20	64.87	74.56	77.60	61.13	77.06	78.64	66.35

Table 3: **Ablation study of various components of our MetaGCD on the CIFAR100 dataset**. We report 'All', 'Old' and 'New' class accuracy for each incremental session, and the average of all sessions such as mean 'All' (mA), mean 'Old' (mO) and mean 'New' accuracy (mN). Here **CN** denotes candidate neighbors and **SP** denotes soft positiveness.

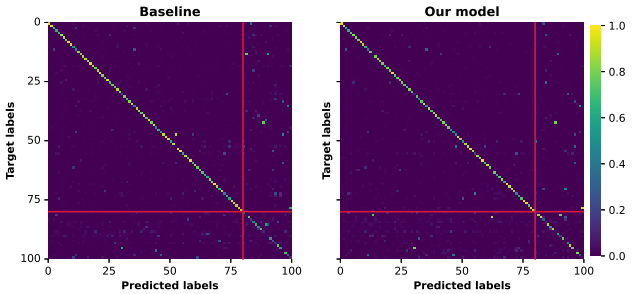

Figure 3: **Class-wise performance on the CIFAR100 dataset**. The confusion matrices show that our model significantly improves the baseline for both *known* and *novel* classes (separated by the red line). Especially for novel classes, the confusion matrix of our method has more concentrated values along the diagonal.

Besides, our model outperforms the baseline VanillaGCD by 6.25%, 3.12% and 4.09% for the final *All* classes. We also report the class-wise performance via the confusion matrices shown in Fig. 3. It is obvious that the baseline performs poorly, especially on the *novel* classes. However, the proposed model has a significant gain in discovering *novel* classes. Moreover, less forgetting is observed in our method, as more values are concentrated on the diagonal.

4.3. Ablation Study

We conduct ablation studies on the CIFAR100 dataset to evaluate each component in our proposed framework.

Importance of neighborhood. In our contrastive learning framework, we compute the soft correlation to allow more positive pairs to contribute to the loss. As reported in the second row of Tab. 3, considering the neighborhood achieves a performance gain of 1.25% (*i.e.*, 74.92% *v.s.* 73.67% on the mean accuracy of *All* classes). The performance gain may come from the abundant comparisons from positive samples, which facilitates the current instance to align with more highly-correlated samples. We also conduct

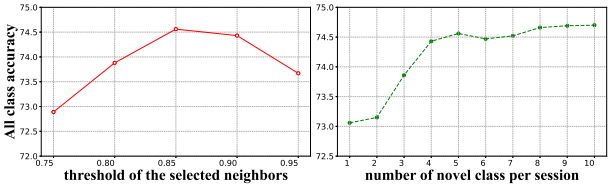

Figure 4: **Hyper-parameter analysis on the CIFAR100 dataset regarding feature similarity threshold (left) and various numbers of novel classes (right)**. An appropriate threshold or the number of new classes helps to stabilize the training process and improve performance.

experiments to assess the sensitivity of threshold ϵ when selecting the positive neighbor instances. Increasing ϵ allows more strict positive pairs, but some partially related samples might be ignored. On the other hand, reducing ϵ increases the likelihood of introducing true negative samples, which may negatively impact model performance. As empirically found in the left side of Fig. 4, a trade-off should be made, and a threshold of 0.85 achieves the best performance.

Importance of soft positiveness. We then analyze the importance of soft positiveness to the recognition performance in the third row of Tab. 3. When we compute a correlation weight for each selected neighbor, the clustering accuracy on *Novel* classes increases from 62.89% to 64.55%. It indicates that the binary labeling strategy is insufficient to measure the correlation at the feature space, thus causing the backbone network to produce less discriminative representations compared with soft labeling methods. In the lower part of Fig. 2 (a), we show the correlation weight of selected neighbors with an input instance. The high score corresponds to the same category while the low score corresponds to less-correlated categories, which shows that our attention module is effective in modeling correlations between the input instance and each candidate neighbor.

Effectiveness of meta-learning. Our meta-learning optimization strategy further improves *All* classes accuracy to 74.56% for the final incremental session in the last row

of Tab. 3. It demonstrates the effectiveness of the proposed method where the meta-objective specifically forces the model to discover novel classes without forgetting old classes. Additionally, we analyze the impact of the number of novel classes that are sampled during meta-training. To investigate this, we train separate models by setting the number of novel classes in the range of $\{1, 10\}$. As illustrated on the right side of Fig. 4, a larger number of classes for sequence tasks is more optimal. When there are fewer classes, the model is at a higher risk of overfitting to certain classes rather than learning how to incrementally learn.

5. Conclusion

In this paper, we propose a more realistic setting for real-world applications, namely C-GCD. The ultimate goal of C-GCD is to discover novel classes while keeping the old knowledge without forgetting. We propose a meta-learning based optimization strategy to directly optimize the network to learn how to incrementally discover with less forgetting. In addition, we introduce a soft neighborhood contrastive learning to utilize the soft positiveness to adaptively support the current instances from their neighbors. Extensive experiments on three datasets demonstrate the superiority of our method over state-of-the-art methods.

6. Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (No. 2022JBZY019), the National Key Research and Development Project (No. 2018AAA0100300) and an NSERC Discovery grant.

References

[1] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in Neural Information Processing Systems*, 32, 2019.

[2] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14493–14502, 2020.

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9650–9660, 2021.

[4] Can Chen, Yingxueff Zhang, Jie Fu, Xue Steve Liu, and Mark Coates. Bidirectional learning for offline infinite-width model-based optimization. *NeurIPS*, 2022.

[5] Can Chen, Shuhao Zheng, Xi Chen, Erqun Dong, Xue Steve Liu, Hao Liu, and Dejing Dou. Generalized dataweighting via class-level gradient manipulation. *NeurIPS*, 2021.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the IEEE International Conference on Machine Learning*, pages 1597–1607, 2020.

[7] Zhixiang Chi, Li Gu, Huan Liu, Yang Wang, Yuanhao Yu, and Jin Tang. Metafscil: A meta-learning approach for few-shot class incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14166–14175, 2022.

[8] Zhixiang Chi, Yang Wang, Yuanhao Yu, and Jin Tang. Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9137–9146, 2021.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the IEEE International Conference on Learning Representations*, 2021.

[11] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9284–9292, 2021.

[12] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of International Conference on Machine Learning*, pages 1126–1135, 2017.

[13] Li Gu, Zhixiang Chi, Huan Liu, Yuanhao Yu, and Yang Wang. Improving protonet for few-shot video object recognition: Winner of orbit challenge 2022. *arXiv preprint arXiv:2210.00174*, 2022.

[14] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

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

[16] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Autonovel: Automatically discovering and learning novel visual categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6767–6781, 2021.

[17] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8401–8409, 2019.

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual rep-

resentation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

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

[20] Khurram Javed and Martha White. Meta-learning representations for continual learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[21] KJ Joseph, Sujoy Paul, Gaurav Aggarwal, Soma Biswas, Piyush Rai, Kai Han, and Vineeth N Balasubramanian. Novel class discovery without forgetting. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, pages 570–586, 2022.

[22] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[23] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4), 2009.

[24] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[25] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[26] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017.

[27] Hanwen Liang, Niamul Quader, Zhixiang Chi, Lizhe Chen, Peng Dai, Juwei Lu, and Yang Wang. Self-supervised spatiotemporal representation learning by exploiting video continuity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1564–1573, 2022.

[28] Huan Liu, Zhixiang Chi, Yuanhao Yu, Yang Wang, Jun Chen, and Jin Tang. Meta-auxiliary learning for future depth prediction in videos. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 5756–5765, 2023.

[29] Huan Liu, Li Gu, Zhixiang Chi, Yang Wang, Yuanhao Yu, Jun Chen, and Jin Tang. Few-shot class-incremental learning via entropy-regularized data-free replay. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, pages 146–162, 2022.

[30] Huan Liu, Zijun Wu, Liangyan Li, Sadaf Salehkalaibar, Jun Chen, and Keyan Wang. Towards multi-domain single image dehazing via test-time training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2022.

[31] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020.

[32] Yiwei Lu, Frank Yu, Mahesh Kumar Krishna Reddy, and Yang Wang. Few-shot scene-adaptive anomaly detection. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, pages 125–141, 2020.

[33] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11205–11214, 2021.

[34] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2017.

[35] Subhankar Roy, Mingxuan Liu, Zhun Zhong, Nicu Sebe, and Elisa Ricci. Class-incremental novel class discovery. In *Proceedings of the IEEE Conference on European Conference on Computer Vision*, pages 317–333, 2022.

[36] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *International Conference on Machine Learning*, pages 1842–1850, 2016.

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2014.

[38] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, 30, 2017.

[39] Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho. Meta-transfer learning for zero-shot super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3516–3525, 2020.

[40] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7492–7501, 2022.

[41] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019.

[42] Yanan Wu, Tengfei Liang, Songhe Feng, Yi Jin, Gengyu Lyu, Haojun Fei, and Yang Wang. Metazscil: A meta-learning approach for generalized zero-shot class incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10408–10416, 2023.

[43] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12455–12464, 2021.

[44] Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in Neural Information Processing Systems*, 34:23664–23678, 2021.

[45] 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. *Advances in Neural Information Processing Systems*, 2022.

[46] Tao Zhong, Zhixiang Chi, Li Gu, Yang Wang, Yuanhao Yu, and Jin Tang. Meta-dmoe: Adapting to domain shift by meta-distillation from mixture-of-experts. *Advances in Neural Information Processing Systems*, 35:22243–22257, 2022.

[47] Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10867–10875, 2021.