Incremental Generalized Category Discovery

Citation for published version:
https://doi.org/10.1109/ICCV51070.2023.01753

Digital Object Identifier (DOI):
10.1109/ICCV51070.2023.01753

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published in:
2023 IEEE/CVF International Conference on Computer Vision (ICCV)

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
Incremental Generalized Category Discovery

Bingchen Zhao  Oisin Mac Aodha
University of Edinburgh
https://bzhao.me/iNatIGCD

Figure 1: We explore the task of discovering novel categories over time. Here we illustrate visual differences between the same categories across different continents, where the continent colors represent four different incremental learning stages from our new iNatIGCD dataset. Each box contains two instances of the same category, where the images are from different geographical locations. We can see that the background and appearance can change dramatically across different locations.

Abstract

We explore the problem of Incremental Generalized Category Discovery (IGCD). This is a challenging category-incremental learning setting where the goal is to develop models that can correctly categorize images from previously seen categories, in addition to discovering novel ones. Learning is performed over a series of time steps where the model obtains new labeled and unlabeled data, and discards old data, at each iteration. The difficulty of the problem is compounded in our generalized setting as the unlabeled data can contain images from categories that may or may not have been observed before. We present a new method for IGCD which combines non-parametric categorization with efficient image sampling to mitigate catastrophic forgetting. To quantify performance, we propose a new benchmark dataset named iNatIGCD that is motivated by a real-world fine-grained visual categorization task. In our experiments we outperform existing related methods.

1. Introduction

The wealth and complexity of visual information potentially observable by artificial systems deployed in the real world vastly exceeds the comparative simplicity of our carefully curated benchmark vision datasets. To operate safely and reliably in challenging environments, these systems need to be able to correctly recognize previously learned concepts, not confuse these known concepts with novel ones, and be able to differentiate novel concepts so that they can be grouped and efficiently learned. As humans, we excel at this type of flexible learning in such dynamic settings [1, 34], and it is clear that we need to endow our artificial systems with similar desirable abilities.

In the context of visual categorization, there is a rich body of work that has moved beyond the traditional supervised setting into more open-ended learning paradigms. For example, in semi-supervised learning, in addition to labeled data, during training we also have access to unlabeled
We address these issues via a novel IGCD method that combines a non-parametric classifier with a density-based sampling mechanism that efficiently enables the selection of informative examples for both classification and past memorization.

Benchmarking in the incremental discovery setting to date has largely been restricted to artificial category-based splits of conventional image categorization datasets [32, 59, 31, 39]. To encourage future progress on IGCD, we also present a new dataset called iNatIGCD. iNatIGCD more faithfully simulates a real-world fine-grained incremental learning setting. It is motivated by a real-world visual categorization use-case whereby the categories present at different stages are selected using spatio-temporal metadata from the community science platform iNaturalist [26]. As a result, iNatIGCD naturally includes challenges such as appearance shifts, in addition to both fine and coarse-grained differences between old and novel categories present in images sourced from different geographical regions.

In summary, we present the following contributions: (i) A new approach for IGCD that combines non-parametric classification with efficient incremental learning via density-based support set selection. (ii) iNatIGCD, a new in-the-wild dataset for benchmarking IGCD that features a multi-stage training split motivated by a real-world fine-grained visual categorization use-case. (iii) A thorough evaluation on IGCD where we outperform recent methods.

2. Related Work

Here we review existing work in semi-supervised learning and both standard and incremental category discovery. 

Semi-Supervised Learning. Conventional semi-supervised learning assumes that we have access to both labeled and unlabeled data at training time [65, 11, 44]. Many works have been proposed to tackle this task using pseudo-labeling [46], consistency regularization [6, 53, 52, 33], density-based label propagation [35], or non-parametric categorization [3]. It is typically assumed that the unlabeled data contains instances from the same categories that are present in the labeled set. Recent works have extended semi-supervised learning by removing the assumption that the categories in the unlabeled and labeled sets are the same [50, 25, 61], but their focus is still on the performance on the labeled categories and they do not evaluate the clustering accuracy on the novel categories in the unlabeled set.

Open-set and Open-world recognition. Open-set recognition considers the case where novel classes can appear during testing, and the model needs to reject those novel classes [51, 18, 5]. Thus open-set recognition method cannot be directly used for category discovery as they only rejects the novel categories. Similarly, open-world recognition methods deal with the novel categories by incrementally soliciting human labels for the novel cate-
gories [4, 7, 41, 42]. While in category discovery, we require the models to automatically discover the novel categories without the human-in-the-loop.

**Category Discovery.** In contrast to semi-supervised learning, Novel Category Discovery (NCD) [20] addresses an alternative setting whereby there is no overlap between the categories in the labeled and unlabeled sets. Here, the goal is to automatically discover the novel categories in the unlabeled data. This can be viewed as a semi-supervised clustering problem [23, 24, 20].

Recently, Generalized Category Discovery (GCD), a more realistic and challenging version of the discovery problem has been proposed [9, 57]. In this setting, the unlabeled data can consist of images from both seen and unseen (i.e. novel) categories. Earlier works showed that self-supervised pretraining can aid category discovery [19]. Starting from a backbone initialized using self-supervised pretraining [12], ORCA [9] assumes the number of novel categories is known and proposed a three-component loss to train a deep classifier. The loss is comprised of a supervised component, a pairwise loss that uses high-confidence pseudo-labels to ensure that similar unlabeled instances are grouped together, and a regularization term to ensure that all instances are not assigned to the same category. GCD [57] also make use of self-supervised pretraining using [10], but foregoes the need for pseudo-labels by instead using a clustering-based approach. They perform a contrastive finetuning step using image pairs from the same category for the labeled data and augmented pairs for the contrastive finetuning step using image pairs from the same category for the labeled data and augmented pairs for the unlabeled data. Finally, they perform unsupervised clustering using k-means to assign unlabeled images to categories.

Building on GCD [57], SimGCD [60] investigated the impact of different design choices on downstream performance. Their final approach does not use an explicit clustering step but instead makes use of learned category prototypes inspired by self-distillation methods [2], resulting in improved performance compared to GCD. MIB [13] employs a similar pretraining phase to GCD to train their feature extractor, in addition to a cross-entropy and conditional entropy loss for the labeled and unlabeled data respectively. We take inspiration from these methods, but explore a different setting of the discovery problem, that of IGCD.

**Incremental Category Discovery.** Category-incremental learning is a learning setting whereby the number of categories in a dataset increases over time, e.g. over a set of discrete learning stages. The challenge in the incremental setting is that it is not possible to store all of the previously observed training data from earlier stages during each subsequent learning stage. Some of the difficulties that arise from this setting include catastrophic forgetting [40, 30] (i.e. a dramatic decrease in performance on old categories as a result of training on new ones) and susceptibility to distributional shifts in the data. Multiple different approaches have been explored in the literature to address this, including storing past training examples in a replay buffer [47] or storing distilled exemplars [45, 8]. Taxonomic variants of the problem have also been explored [36, 13]. For a recent survey on the topic we refer readers to [64].

Most relevant to our work are a recent set of approaches that explore category discovery in the incremental/continuous setting, e.g. [48, 27, 62, 37]. [48] proposed a one-stage category-incremental setting termed class-iNCD. Here labeled data is initially available to learn a representation, then discarded, and finally a set of unlabeled data containing only novel categories is provided. Their goal is to train a model that performs well on both sets of categories. However, here there is no overlap between the categories in the labeled and unlabeled sets. This setting has also been explored in [38], where their focus was only on the category discovery performance. FRoST [48] retains feature prototypes learned from labeled data which are replayed during the discovery phase to prevent forgetting of the old categories. NCDwF [27] also explore the same setting and propose a method that uses pseudo-latent supervision, feature distillation, and a mutual information-based regularizer to maintain performance on the discarded labeled categories and assist in discovering novel ones. In the MSc-iNCD setting explored in [37], the model only obtains unlabeled data at each time step which is assumed to come from previously unseen (i.e. novel) categories.

Grow and Merge (GM) [62] was recently introduced and applied to several different incremental discovery settings. It performs two key phases at each learning stage. First in the growing phases, it performs novelty detection to separate novel from seen categories and then trains a dynamic network to perform NCD. Then in the merging phases, it combines the newly discovered categories with the previously known ones into a single model. Later, we compare to GM and show superior performance.

With the exception of GM, who only perform one related experiment on CIFAR-100 [32], other existing methods assume that unlabeled data can only contain novel categories. In contrast, we explore the more challenging incremental generalized setting where, at each stage, new data can come from either previously seen or novel categories. Unlike [62], in our setting, all previously seen categories are not guaranteed to be present in each subsequent stage.

**Benchmarking Category Discovery.** There are several common datasets used to evaluate category discovery methods. Earlier works [20, 19] typically use standard image categorization datasets like CIFAR10/100 [32] and ImageNet [49] by creating artificial splits for discovery evaluation. [63] argue that more challenging fine-grained datasets like CUB [59] are more suitable for evaluating discovery performance as the labeled and unlabeled categories share more visual similarity. Recently, the Semantic-Shift Bench-
mark (SSB) [58, 57] was proposed to better evaluate the task of detecting semantic novelty. SSB uses the existing CUB [59], Stanford Cars [31], and FGVC-Aircraft [39] datasets where the category splits are designed to have a clear ‘axes of semantic variation’ as well as a coherent definition of categories in the labeled and unlabeled sets.

Recent work in incremental discovery [48, 27, 62] evaluate performance using artificial data splits on CIFAR-10/100 [48, 62, 27], TinyImageNet [48], ImageNet [27], or the fine-grained SSB datasets [62]. However, these artificial data splits may not reflect the real-world performance of IGCD methods. For example, it has been shown that the category splits in the labeled dataset play an important role NCD performance [16]. To address this gap, we introduce a new ‘real-world’ data split derived from the publicly available fine-grained iNaturalist species categorization dataset [56]. We make it applicable to the incremental setting by leveraging metadata, such as the date and location of each observation. From this, we are able to better simulate real-world IGCD by creating data and category splits that mimic those that naturally arise over multiple time steps in the real-world. Note that the geographical domain shift has also been studied in GeoNet [28], where the main task is unsupervised domain adaptation. Besides the task difference, our iNatIGCD dataset also contains 10× more images, from more geographical locations, and fine-grained concepts that are challenging for category discovery.

3. The iNatIGCD Benchmark

Here we outline iNatIGCD, our new benchmark for IGCD. Our dataset is based on the large-scale fine-grained iNat2021 visual categorization dataset [56], which contains images from 10,000 different categories of plant and animal species sourced from the community science platform iNaturalist [26]. Each of the images posted to iNaturalist contains the precise capture date and time, along with the geographical location as metadata [26]. This creates a natural stream of data over time that contains category distribution shifts between observations from around the globe as the iNaturalist community grows across different regions.

iNatIGCD is motivated by the scenario where one can develop an initial image classifier from the currently available labeled data, which only contains images and labels captured before a certain time and is restricted to a certain region (e.g. a specific continent). As time progresses, new unlabeled data is obtained which can also include images from other regions (e.g. a different continent). This scenario is close to the real-world situation of the iNaturalist platform where we have access to labeled image observations at a given time point, but we cannot easily obtain reliable labels for new observations that are shared each day as labeling images takes time. We can view the existing collection of observations as the labeled dataset and the new observations, which can contain a combination of old and new categories, as the unlabeled dataset. Similar to the common assumption in continual learning [45, 8], all the previous labeled data can be difficult, or even impossible, to store during learning. Thus as time progresses, we assume that we do not have access to all the past labeled data when re-training/updating the model.

Our new IGCD benchmark is structured as follows: At each time step, we have a labeled dataset and an unlabeled one. Our goal is to correctly discover any novel categories in the unlabeled dataset while also classifying examples belonging to known categories. When we progress to a new time step, the labeled dataset from the previous step becomes unavailable. However, we now have a new labeled dataset, which is actually the previous unlabeled dataset, along with a new unlabeled dataset which is used for category discovery. This simulates the real-world process as it takes time for the iNaturalist community to reach a consensus regarding which category is present in a newly uploaded image. See Fig. 2 for an illustration of the process.

**Dataset Construction.** To generate our data splits, we leverage metadata from the iNat2021 dataset [56]. We first sort and group all the images according to the time when they were captured. To make the data more evenly distributed, we cluster them into four temporal stages: 2008–2016, 2017, 2018, and 2019. We further split the data according to the location of the observation, only including data from North America and Europe in the first stage and then including Oceania, Asia, and finally Africa and South America in the second, third, and fourth stages respectively. This simulates the growth of the species recognition community on iNaturalist from continent to continent over time.

Our new iNatIGCD dataset is summarized in Fig. 3. It poses several unique challenges for category discovery methods: (i) we have an order of magnitude more fine-grained categories in each of the incremental stages compared to existing datasets which will challenge the ability of category discovery methods, (ii) at each stage, we have more images that require categorization, which can give a...
more reliable measure of the discovery performance of a model, and (iii) the categories that appear in each stage are based on the natural distribution of species rather than an artificial split as in previous benchmarks.

4. Method

In this section we present a new approach targeting the IGCD setting. At each stage \( t \) (i.e. time step), we have two datasets to train the model on, \( D_{\text{lab}}^t \) and \( D_{\text{unlab}}^t \). Our goal is to learn a classifier that can correctly classify and discover categories in \( D_{\text{lab}}^t \) and \( D_{\text{unlab}}^t \), in addition to maintaining its performance on the categories in the current stage when we advance to later stages.

In the IGCD setting, it is assumed that it is not possible to store all the previously observed data. As a result, the model must learn incrementally. A common incremental learning technique is to avail of the concept of example replay [45, 8]. This involves saving a few examples of each category and then ‘replaying’ them to the model during training in later stages. However, there are challenges associated with applying replay approaches in our IGCD setting due to two primary issues: (i) we have novel categories in the unlabeled dataset in IGCD, thus it is not straightforward to select examples for these novel categories as previous works typically assume full supervision, and (ii) SoTA GCD methods use parametric classifiers [60, 17], which we show in experiments can overfit to the few replay examples. We tackle these technical challenges by using a non-parametric classifier combined with a density-based example selection mechanism which we introduce below.

4.1. Problem Setting

Formally, we denote the labeled dataset at the initial stage as \( D_{\text{lab}}^0 = \{(x_i^0, y_i^0)\} \). This is used to train an initial model to recognize the set of categories \( \{1, 2, \ldots, K^0\} \). For incremental learning at a later stage \( t \), we have two datasets, a labeled dataset \( D_{\text{lab}}^t = \{(x_i^t, y_i^t)\} \) and an unlabeled dataset \( D_{\text{unlab}}^t = \{x_i^t\} \). We denote the category sets in \( D_{\text{lab}}^t \) and \( D_{\text{unlab}}^t \) as \( C_{\text{lab}}^t \) and \( C_{\text{unlab}}^t \). In our generalized setting, \( C_{\text{lab}}^t \) and \( C_{\text{unlab}}^t \) can be different and may only have partial overlap. When we progresses to stage \( t + 1 \), \( D_{\text{unlab}}^t \) will receive its labels which are used as \( D_{\text{lab}}^{t+1} \) for the next stage. The goal is to learn a model that can not only discover novel categories in the unlabeled dataset \( D_{\text{unlab}}^t \) at stage \( t \), but also maintain the ability to recognize the categories in \( C_{\text{lab}}^t, C_{\text{lab}}^{t+1}, \ldots, C_{\text{lab}}^t \). Similar to the category-incremental learning setting [45, 8], one main added challenge of our case, compared to static GCD, is that the model can exhibit catastrophic forgetting [47] on the categories that are not present in \( C_{\text{lab}}^t \) and \( C_{\text{unlab}}^t \).

Note, our setting is related, but different from the Mixed Incremental (MI) one in [62]. In contrast to us, [62] assume that only the unlabeled data \( D_{\text{unlab}}^t \) at stage \( t \) is available for training the model. In our experiments, we also compare models that are only trained with \( D_{\text{unlab}}^t \). Furthermore, in the MI setting, the category set \( C_{\text{unlab}}^t \) always includes all previously seen categories. In our more difficult and realistic setting, the category set at each time step may only have partial overlap with the previous category sets.

4.2. Classifier Learning

For each input image \( x_i \), we use a neural network \( f \) to extract its corresponding feature representation \( h_i = f(x_i) \). From this, we use a non-parametric classifier \( g \) to assign the image to a category probability vector \( p_i = g(h_i) \), where \( \sum_c p_{i,c} = 1 \). For \( f \) to learn an informative representation, we adopt a supervised contrastive loss \( L_{\text{SupCon}} \) from [29] for learning from \( D_{\text{lab}}^t \) and use a self-supervised contrastive loss \( L_{\text{SelfCon}} \) from [12] for learning from \( D_{\text{unlab}}^t \). These two losses are combined via a balancing factor \( \lambda_{\text{rep}} \).

\[
L_{\text{rep}} = \lambda_{\text{rep}}L_{\text{SupCon}} + (1 - \lambda_{\text{rep}})L_{\text{SelfCon}}.
\]

We use a non-parametric Soft Nearest-neighbor (SNN) classifier for \( g \) that computes the distribution over category labels for an input image \( x_i \),

\[
p_i = \text{SNN}(h_i, S, \tau) = \sum_{k=1}^{N^S} \frac{\exp(h_i \cdot h_k^i/\tau)}{\sum_{h_k^i \in S} \exp(h_i \cdot h_k^i/\tau)} y_k.
\]

\( S = \{(h_k^i, y_k)\} \) is a support set containing features of all support samples \( h_k^i \) and their corresponding one-hot labels \( y_k \). \( N^S = |S| \) is the number of support samples in \( S \), and \( K^S \) is the number of categories in \( S \). \( h_i \) is the representation of \( x_i \) from \( f \), and \( \tau \) is a temperature parameter controlling the sharpness of the prediction \( p_i \). The support samples are representative examples of each of the categories and are used to softly assign a label to the input example \( h_i \) by averaging all the one-hot labels \( y_k \) in \( S \) weighted by the softmax similarity. For the labeled categories, we randomly select the support examples. Later we discuss the selection of support examples for the novel categories.

Importantly, as our classifier is non-parametric, it is easy to extend it so that it can classify more categories by simply adding additional examples from the new categories to the support set. At test time, we just use the saved support examples to perform categorization for an input image by assigning the labels via measuring the distance between the input image and support examples. The classifier is trained with two cross-entropy losses \( L_{\text{ce}} \), and makes use of an entropy-regularizer,

\[
L_t = \frac{1}{|B_t|} \sum_{i \in B_t} L_{\text{ce}}(y_i^t, \hat{p}_i^t),
\]

\[
L_u = \frac{1}{|B_u|} \sum_{i \in B_u} L_{\text{ce}}(\hat{p}_i^u, \bar{p}_i^u) - \epsilon H(\bar{p}),
\]
where $B^l$ and $B^u$ are batches of indices from the labeled and unlabeled datasets, $y^l_i$ is the ground-truth label for the example $x^l_i$, and $\hat{p}^l_i$ and $\hat{p}^u_i$ are the predictions for two augmented views of the same input example $x^u_i$ with different temperatures. These two losses together form the loss function we use for classifier learning,

$$L_{cls} = \lambda_{cls}L_l + (1 - \lambda_{cls})L_u,$$

where $\lambda_{cls}$ is a weighting factor. The entropy regularization term, $H(p) = - \sum p \log p$, regularizes the mean prediction $p$ over a mini-batch which is computed as

$$p = \frac{1}{|B^u|} \sum_{i \in B^u} (\hat{p}^u_i + \hat{p}^l_i).$$

This above entropy term has been adopted in previous works in GCD [60] and semi-supervised learning [3] to calibrate predicted category distribution and to avoid empty clusters, as observed in [57, 60]. The overall loss for training the model is

$$L = L_{rep} + L_{cls}.$$  

4.3. Support Sample Selection

Our SNN classifier function is similar to the one defined in the semi-supervised approach of [3]. However, in its original form, it cannot be used to learn from novel categories from the unlabeled data if they do not already appear in the labeled set. We extend this approach via a novel modification to make it applicable to the generalized category discovery setting. Specifically, for the potentially novel categories in the unlabeled dataset, we propose a new density-based selection mechanism to select a subset of examples from the unlabeled dataset and then pseudo-label them with $g$ to use them as support samples in $S$.

We start by estimating the density $d_i$ of a sample $x^u_i$ based on its $k$-nearest neighbors, such that

$$d_i = \frac{1}{K} \sum_{j \in N_{K}^{x^u_i}} \frac{h^u_j - h^u_i}{\|h^u_j\|_2 \|h^u_i\|_2}.$$  

Here we iterate through the top-$K$ neighbors and average the cosine similarities of the feature representations, where $N_{K}^{x^u_i}$ contains the indices of the $K$ nearest neighbors of $x^u_i$. Intuitively, a larger density value $d_i$ indicates that the corresponding prediction for $x^u_i$ is a more reliable pseudo-label as it indicates that $x^u_i$ is similar to its neighbors in the learned representation space of $f$, see [35]. Inspired by this, we propose a method for automatically selecting reliable samples from $D_{unlab}$ using this density definition.

First, we select the density peaks from the dataset which are the points whose density is higher than all other $k$-nearest neighbors around it [35]. Specifically, the set of density peaks can be defined as

$$\mathcal{P} = \{x^u_i | \forall k \in N^K_{x^u_i}, d_i > d_k\},$$

where $\mathcal{P}$ is a set of selected images $x_i$. Second, it is not guaranteed that there is only one density peak per class. To address this, we use an intersection-over-union (IoU) score on the nearest neighbor set between two density peaks to measure their similarity. We then remove redundant peaks using a similar procedure to non-maximum-suppression [43].

Thus, for two density peaks $x^u_i$ and $x^u_j$ in $\mathcal{P}$, we measure their similarity by calculating the IoU score of their $K$-nearest neighbor sets $N^K_{x^u_i}$ and $N^K_{x^u_j}$. If the IoU score is higher than a threshold $T$, we only keep the peak with the higher density value. The selected density peaks are used to pseudo-label their close neighbors based on distance, and then the density peaks and their closer neighbors are added as support samples for novel categories to $S$. For the labeled categories, we select a number of support examples from the labeled dataset using the same density selection approach.
mechanism to serve as their support samples. This procedure is outlined in detail in the supplementary material.

The benefit of our proposed method is that we automatically have an estimate of the number of novel categories in the unlabeled set based on the number of selected density peaks. This is in contrast to previous methods for estimating the number of categories which make use of the $k$-means algorithm, e.g. [20, 57]. Our approach is more closely integrated with the representation learning step and is also more computationally efficient.

4.4. Incremental Update

Earlier we introduced our method for learning a classifier that leverages non-parametric category assignments. Now we discuss how to extend this model to the incremental setting. To do this, we focus on learning between a pair of stages, i.e. from stage $t$ to $t+1$.

After training on $D_{lab}^t$ and $D_{unlab}^t$ with the objectives defined in Sec. 4.2, we have a model that can classify the labeled and unlabeled categories, $C_{lab}^t$ and $C_{unlab}^t$. To handle subsequent stages, we draw inspiration from the classic category-incremental learning method iCaRL [45]. Specifically, we store a set of representative examples $(x_i, y_i)$ to a memory buffer $\mathcal{R}^t$ for replay to be used for training at later stages. Note, this is not to be confused with the support set $S$ used by our non-parametric classifier defined earlier.

Different from iCaRL which selects examples using category centroids, we use our proposed density selection mechanism described in Sec. 4.3 to select the examples to save to $\mathcal{R}^t$. Thus at stage $t+1$, we concatenate the training set $D_{lab}^{t+1}$ with the memory buffer $\mathcal{R}^t$ and then optimize the training objectives in Sec. 4.2. The memory buffer created at stage $t+1$ is also concatenated with $\mathcal{R}^t$ to ensure that the knowledge from all previous stages is maintained.

5. Experiments

5.1. Implementation Details

Model Training. We use a ResNet-18 [22] initialized using MoCo [21] on ImageNet-1k as our feature extractor $f$. We freeze its parameters up to the last residual block to prevent the model from overfitting on the labeled data as in [19, 63]. We train for 100 epochs on the labeled data from the initial stage $D_{lab}^0$, and then train on each subsequent stage $t$ with $D_{lab}^t$ and $D_{unlab}^t$ for 40 epochs each. We provide additional vision transformer [15] results and details of the training settings in the supplementary material.

Evaluation Metrics. We use our iNatIGCD dataset to evaluate the performance of different IGCD methods, using two different settings: (i) IGCD-l, where we assume the labels of $D_{unlab}^t$ will become available at the end of stage $t$ and thus can be used as $D_{lab}^{t+1}$, and (ii) IGCD-u, where we assume no data is labeled in any of the incremental stages, and at stage $t$ only $D_{unlab}^t$ is available to train the model. We remove the supervised losses accordingly in IGCD-u.

For our experiments on iNatIGCD, we report the clustering accuracy for categories in $C_{lab}^t$ and $C_{unlab}^t$ at stage $t$ as ‘Old’ and ‘New’, and ‘All’ as the performance on both. We also report the accuracy on the categories in previous stages $0, 1, \ldots, t-1$ that do not appear in the current stage as ‘S-$t$’. As in [62], we report maximum forgetting $M_{f}$, which is the maximum difference between the clustering accuracy from the stage 0 categories and any later stage $t$. We also report final discovery $M_{d}$, which is the final clustering accuracy of the model on all the categories from all stages.

5.2. Results

Here we describe the results of relevant methods on different IGCD benchmarks. We include comparisons to GM [62], a recent IGCD method. We also compare to FRoST [48], which is a one-stage method that can only handle one incremental stage. We extend it to the multi-stage setting by repeating the second discovery phase of FRoST [48] for each new stage. We also compare to SimGCD [60], a recent SoTA GCD method and apply it to IGCD by rerunning it on the data from each new stage. To enable it to maintain performance on previous categories, we combine it with a memory buffer as in iCaRL [45]. Finally, to establish a performance upper bound on iNatIGCD, we train a fully supervised model on all the data and labels at each stage $t$ (including all previous stages $t-1, \ldots, 0$). Additional results are in the supplementary material.

5.2.1 iNatIGCD Results

IGCD-l Setting. We first present results for the incremental setting with labels in Tab. 1 and Fig. 5. Although the non-incremental SimGCD [60] achieves comparable performance at recognizing labeled categories (‘Old’) and discovering novel categories (‘New’) at each stage, it fails to maintain good performance on the categories it has seen before but that are not present in the current stage (see ‘S-0’ and ‘S-1’). Combining SimGCD with iCaRL [45] helps alleviate this forgetting problem, which we see as an increase in ‘S-0’ and ‘S-1’ scores compared to SimGCD alone. However, the performance on ‘Old’ categories, which have labeled data during training, drops for SimGCD + iCaRL. We argue that this is because SimGCD employs a parametric classifier, but iCaRL [45] performs non-parametric categorization at test time. We can see from the Stage-0 performance of SimGCD + iCaRL that this mismatch between training and testing results in a performance drop even in the initial fully supervised scenario. In contrast, our SNN classifier achieves a balance between preventing the category information from previous stages from being forgotten (‘S-0’ and ‘S-1’) as well as obtaining good performance on
Table 1: Results on iNatIGCD in the IGCD-I setting (i.e. where labels are available at the end of each stage). Higher numbers are better, with the exception of $M_f$ where lower is better.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Stage-0</th>
<th>Stage-1</th>
<th>Stage-2</th>
<th>Stage-3</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All</td>
<td>Old</td>
<td>New</td>
<td>All</td>
<td>Old</td>
</tr>
<tr>
<td>Supervised upper-bound</td>
<td>42.3</td>
<td>62.8</td>
<td>68.7</td>
<td>59.8</td>
<td>65.8</td>
</tr>
<tr>
<td>SimGCD [60]</td>
<td>42.3</td>
<td>24.1</td>
<td>35.8</td>
<td>10.5</td>
<td>25.6</td>
</tr>
<tr>
<td>SimGCD + iCaRL [45]</td>
<td>41.5</td>
<td>23.7</td>
<td>33.6</td>
<td>11.9</td>
<td>23.5</td>
</tr>
<tr>
<td>FRoST [48]</td>
<td>42.3</td>
<td>20.5</td>
<td>24.6</td>
<td>9.4</td>
<td>12.4</td>
</tr>
<tr>
<td>GM [62]</td>
<td>42.3</td>
<td>18.7</td>
<td>26.1</td>
<td>8.6</td>
<td>19.5</td>
</tr>
<tr>
<td>Ours</td>
<td>41.8</td>
<td>25.6</td>
<td>34.6</td>
<td>14.5</td>
<td>25.7</td>
</tr>
</tbody>
</table>

Figure 5: The category discovery performance (measured by ‘All’ class clustering accuracy) and the forgetting (measured by accuracy on stage-0 classes) at each stage of the training. We can see that our proposed method obtains strong discovery performance, similar to SimGCD [60], and at the same time exhibits less forgetting.

5.2.2 Mixed Incremental Results

In Tab. 3 we present results on the Mixed Incremental (MI) scenario proposed in [62]. At each incremental stage $t$, the model is trained on a dataset $D_{\text{unlab}}^t$ of unlabeled examples which contains both novel categories and the categories the model has already learned before. The goal is to classify both novel and seen categories at the same time. The difference between MI and our IGCD-u setting is that in MI it is assumed that at each stage there are always instances from the previous stage’s categories in the unlabeled images.

GM [62] only report results for the MI setting on CIFAR-100. For fairness, we re-train their model using the same hyper-parameters as ours and observe improved performance for their method compared to their paper. We also present additional results on CUB [59]. We include comparisons to the strong GCD baseline SimGCD [60], which cannot learn incrementally, and our incremental extension of their method. We observe that SimGCD [60] exhibits catastrophic forgetting, but adding iCaRL [45] can alleviate this issue. Our proposed method achieves better performance than GM [62], with the exception of the $M_f$ performance on the smaller CIFAR-100 dataset.

5.2.3 Ablations

Estimating the Number of Novel Categories. One of the unique challenges in GCD is the task of estimating the number of novel categories in the unlabeled data. [57] described a baseline that uses semi-supervised $k$-means to estimate it. GM [62] adopted this method to estimate the number at
Table 2: Results on iNatIGCD in the IGCD-u setting (i.e. where labeled data is not provided during the incremental stages). Higher numbers are better, with the exception of $M_f$ where lower is better.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Stage-0</th>
<th>Stage-1</th>
<th>Stage-2</th>
<th>Stage-3</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All</td>
<td>New S-0</td>
<td>New S-1</td>
<td>New S-2</td>
<td>S-1</td>
</tr>
<tr>
<td>Supervised upper-bound</td>
<td>42.3</td>
<td>59.8</td>
<td>48.7</td>
<td>62.3</td>
<td>63.9</td>
</tr>
<tr>
<td>SimGCD [60]</td>
<td>42.3</td>
<td>8.4</td>
<td>5.1</td>
<td>12.6</td>
<td>2.3</td>
</tr>
<tr>
<td>SimGCD + iCaRL [45]</td>
<td>41.5</td>
<td>10.5</td>
<td>16.7</td>
<td>13.4</td>
<td>8.4</td>
</tr>
<tr>
<td>FRoST [48]</td>
<td>42.3</td>
<td>6.7</td>
<td>12.3</td>
<td>9.2</td>
<td>7.4</td>
</tr>
<tr>
<td>GM [62]</td>
<td>42.3</td>
<td>8.6</td>
<td>18.4</td>
<td>13.8</td>
<td>9.1</td>
</tr>
<tr>
<td>Ours</td>
<td>41.8</td>
<td>12.7</td>
<td>22.1</td>
<td>15.4</td>
<td>10.7</td>
</tr>
</tbody>
</table>

Table 3: Results on the MI setting introduced in [62].

<table>
<thead>
<tr>
<th>Methods</th>
<th>CIFAR-100</th>
<th>CUB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$M_f$ ↓</td>
<td>$M_d$ ↑</td>
</tr>
<tr>
<td>SimGCD [60]</td>
<td>58.7</td>
<td>28.3</td>
</tr>
<tr>
<td>SimGCD + iCaRL [45]</td>
<td>9.4</td>
<td>29.4</td>
</tr>
<tr>
<td>GM [62]</td>
<td>3.6</td>
<td>30.6</td>
</tr>
<tr>
<td>Ours</td>
<td>4.0</td>
<td>31.2</td>
</tr>
</tbody>
</table>

Table 4: Estimation of the number of novel categories in the unlabeled data at each stage.

Impact of the Selection Method for $R$. Compared to the centroid selection in iCaRL [45], our method uses the concept of density peaks to select the examples to save in the replay buffer $R$. In Tab. 5 we compare centroid selection and our proposed density-based selection. We observe that our method achieves superior performance.

Impact of the Size of $S$. One of the key factors influencing the performance of our SNN classifier is the size of $S$. In Tab. 6, we study the influence of this hyper-parameter by varying the number of examples per category ($N^S/K^S$). When only using one example per category, our model exhibits forgetting and inferior discovery performance. Increasing the number of examples per category increases the performance, until it starts to plateau after five examples per category. To reduce the cost of saving too many examples to $S$, we set $N^S/K^S$ to five in our experiments.

Table 5: Impact of the replay buffer selection method.

Table 6: Impact of the size of the support set $S$.

6. Conclusion

We explored the problem of incremental generalized category discovery (IGCD). To do this, we constructed a new dataset, iNatIGCD, motivated by a real-world fine-grained visual discovery task and used it to benchmark the performance of recent category discovery methods. Through our experiments, we showed that our new approach which combines non-parametric categorization with a density-based sample selection technique is superior to existing methods. While promising, our results show that the IGCD problem, especially in our large-scale fine-grained setting, remains challenging for existing methods. We hope that our work opens the door to future progress on this task.

Acknowledgements. We thank the iNaturalist community for their data collection efforts.
References


[16] Yixin Fei, Zhongkai Zhao, Siwei Yang, and Bingchen Zhao. Xcon: Learning with experts for fine-grained category discovery. In BMVC, 2022. 4


[33] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D object representations for fine-grained categorization. In Workshop on 3D Representation and Recognition, 2013. 2, 4

[34] Alex Krizhevsky. Learning multiple layers of features from tiny images. University of Toronto, 2009. 2, 3


[47] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D object representations for fine-grained categorization. In Workshop on 3D Representation and Recognition, 2013. 2, 4


