Generalized Category Discovery for LiDAR Semantic Segmentation

Supplementary Material

This supplementary material provides additional details to complement the main paper: (i) a detailed summary of the loss terms and pseudo code; (ii) an overview of the clustering approaches used in the OSS Baseline (REAL†); (iii) the criteria for selecting novel categories and the dataset split design to ensure a fair evaluation of GCDLSS; (iv) further explanations on the baseline methods used for comparison; (v) additional qualitative results, and (vi) detailed implementation details.

A. Loss Function Summary and Pseudo Code

Loss	Purpose
$\mathcal{L}_{\text{sup}}$	Supervised CE loss on labeled data, used to both pretrain the initial representations and continuously maintain the performance of the known category classifier ϕ_{kn}.
$\mathcal{L}_{\text{semi}}$	Semi-supervised CE loss on augmented data using teacher-generated pseudo-labels (unlabeled) and original labels (labeled) to update the student known classifier (ϕ^{s}_{kn}) and NCC (ϕ^{s}_{ncc}).
$\mathcal{L}_{\text{mse}}$	Mean squared error loss between student and teacher logits on unlabeled data for consistency regularization.
$\mathcal{L}_{\text{cal}}$	CE-based calibration loss on known category points within LaserMix and resize-augmented data, used to enhance the NCC's prediction confidence for novel categories by ranking the novel category second to the true category.
$\mathcal{L}_{\text{adapt}}$	Margin-based loss for learning the adaptive threshold (θ_{adapt}) that separates known and novel categories.
$\mathcal{L}_{\text{unov}}$	Unsupervised CE loss on reliable novel features for training novel classifier (ϕ_{nov}) with clustering-based pseudo-labels.
$\mathcal{L}_{\text{uncc}}$	Unsupervised CE loss on reliable novel features for updating NCC (ϕ_{ncc}).
$\mathcal{L}_{\text{snov}}$	Supervised CE loss on labeled data for scaling the logits of the known and novel classifiers relative to each other.

Table 1. Losses and their training objectives.

In this section, we summarize all loss terms 1 used in our framework, including their functional forms and training objectives, as well as pseudo-code for the GCDLSS framework. CE denotes Cross Entropy loss, ε denotes the feature extractor, and superscript s, t stands for the student and teacher model. NCC denotes Novel Candidate Classifier.

Algorithm 1 Pseudo-code for GCDLSS Training Procedure

Require: Labeled dataset D_{L}, Unlabeled dataset D_{U}, Number of novel categories C_{nov}

Ensure: A model that can (1) distinguish novel points in unlabeled data, and (2) classify novel points into C_{nov} discovered categories, while preserving known category segmentation performance.

Stage 1: Pretraining
Train known classifier ϕ_{kn} on D_{L} using CE loss ($\mathcal{L}_{\text{sup}}$)
Stage 2: Initialization
Initialize mean teacher model $\varepsilon^{\text{t}} \leftarrow \varepsilon^{\text{s}}$ (EMA sync)
Initialize novel feature queue $\mathbb{Q} \leftarrow \emptyset$
Set adaptive threshold parameter $\theta_{\text{adapt}} \leftarrow 0$
for each training iteration **do**
  (a) Loss Computation and Student Update
  LaserMix + Resize augmentation using $D_{\text{L}} \cup D_{\text{U}}$
  Compute $\mathcal{L}_{\text{semi}}$, $\mathcal{L}_{\text{mse}}$, $\mathcal{L}_{\text{cal}}$, and $\mathcal{L}_{\text{adapt}}$
  Update student model ε^{s} via gradient descent
  (b) Teacher Model Update (EMA Update)
  Update teacher model: $\varepsilon^{\text{t}}_t \leftarrow \tau\varepsilon^{\text{t}}_{t-1} + (1-\tau)\varepsilon^{\text{s}}_t$
  (c) Novel Candidate Extraction
  Compute anomaly scores via NCC: $s(x) = \phi_{\text{ncc}}(x)$
  Select novel candidates $\mathbb{S} = \{x \in D_{\text{U}} \mid s(x) > \theta_{\text{adapt}}\}$
  (d) Novel Candidate Filtering
  Cluster $\mathbb{S} \cup \mathbb{Q}$ into $C_{\text{nov}} + \alpha$ clusters using K-means
  Score centroids with known classifier ϕ_{kn} and discard top-K known clusters (K=α) to secure reliable novel clusters only
  Update novel feature queue $\mathbb{Q}$ (retain 95% old + add new reliable clusters)
  (e) Pseudo-label Assignment and Training
  Assign pseudo-labels to filtered clusters via Hungarian matching
  Train novel classifier ϕ_{nov} using $\mathcal{L}_{\text{unov}}$
  Train NCC using $\mathcal{L}_{\text{uncc}}$
  Jointly train ϕ_{kn} and ϕ_{nov} using $\mathcal{L}_{\text{snov}}$ for logit scaling with D_{L}
end for
Return: Trained ϕ_{kn} and ϕ_{nov} for inference. *Note*: ϕ_{ncc} is used only during training.

B. Mini Batch K-means and Stream K-means Clustering for OSS Baseline

To adapt the REAL† to the GCDLSS setting, we introduce a clustering-based pipeline for discovering novel categories from large-scale LiDAR point clouds. Similar to

GCDLSS, we first perform pretraining to learn the initial representations of known categories. Following pretraining, we fine-tune the model in an OSS setting, where the focus is on enabling the discovery of novel objects. This pipeline leverages Mini-Batch K-Means for initial centroid initialization and Stream K-Means to incrementally refine the centroids during training. During testing, the KNN (K-Nearest Neighbor) algorithm [3] is used for novel category predictions.

1. Novel Centroid Initialization via Mini-Batch K-Means. We first initialize novel category centroids using resize-augmented training data. Given the large scale of the LiDAR dataset, performing traditional K-Means clustering all at once is computationally expensive and impractical. To address this, we use Mini-Batch K-Means, which provides a more efficient way to initialize centroids by processing small batches of data at a time. This step allows us to identify initial cluster centroids quickly. After initialization, we then use Stream K-Means to refine the centroids incrementally during training.

2. Centroid Fitting and Refinement via Stream K-Means. After initial centroid initialization, we perform a second pass over the data to refine the cluster assignments and update centroid positions using Stream K-Means. We select features with high residual confidence (e.g., softmax probability for the novel category) and incrementally fit the centroids using the Stream K-Means algorithm. This process efficiently builds a set of refined cluster centroids representing potential novel categories without loading the entire dataset into memory.

3. Novel Classification with KNN. At test time, features from novel candidate points are extracted using the trained model. We assign novel category labels using KNN classification based on the fitted centroids. This enables non-parametric, instance-level classification of novel points. Known category predictions are generated via argmax over the known-class softmax outputs, and all predictions are merged for evaluation.

4. Cluster Matching and Evaluation. To account for the arbitrary ordering of discovered clusters, we align the predicted clusters with ground-truth novel labels using Hungarian matching on the cost matrix derived from category-wise counts, as in GCDLSS. This matching is used to compute final evaluation metrics, such as mean IoU for known and novel categories.

C. Details on Novel Category Selection

As mentioned previously, our novel category selection is initially motivated by the criteria proposed in NOPS [4]. Specifically, NOPS defines the following principles: (1) maintain semantic diversity by avoiding highly similar known–novel pairs, (2) ensure that novel categories are inferable from known ones (e.g., *bike* $\rightarrow$ *rider*), and (3) avoid category imbalance, where a single frequently occurring novel category disproportionately dominates training.

Building upon these principles, we refine the criteria for our GCDLSS setting to address unique challenges in LiDAR semantic segmentation. In particular, since the amount of labeled data can drastically change depending on the selected novel categories, we prioritize selecting category-wise distinctive novel categories that are semantically separable.

Prior 2D GCD setups [5, 6] typically adopt a sample-level labeled-to-unlabeled ratio of 80:20 or 50:50. However, in LiDAR semantic segmentation (LSS), maintaining such a high labeling ratio becomes significantly more challenging due to the dense nature of point-level annotations. As shown in Fig. 1, each point cloud frame in LiDAR datasets contains a significantly higher number of labeled instances compared to image classification or 2D segmentation. For example, frequently occurring categories—such as *pole* and *fence* in SemanticKITTI, or *manmade* and *car* in nuScenes—appear so often across the dataset that selecting them as novel categories would drastically reduce the proportion of usable labeled data. This leads to a severe imbalance in the labeled data, ultimately undermining the stability and representativeness of model training.

To better reflect these constraints, we limit the proportion of labeled scans to 25%. This protocol captures practical supervision constraints while preserving enough labeled data to enable reliable discovery of novel categories.

Finally, we exclude specific categories from the novel set if their oracle performance (Fig.2)—i.e., fully supervised accuracy on the entire dataset—is significantly low. These categories (e.g., *motorcyclist*, *other-ground*) often exhibit poor segmentation quality even under full supervision. Including such categories would make it difficult to distinguish whether low performance arises from model limitations or intrinsic ambiguity, thereby compromising the fair evaluation of novel category performance.

D. Additional Explanations for the Baseline

In this section, we provide additional details on the design of baseline experiments that were not previously covered.

D.1. 2D Generalized Category Discovery

We conducted a SimGCD [6]of 2D GCD for the GCDLSS baseline. For SimGCD†, we apply supervised contrastive learning and self-contrastive learning. To avoid computational overheads, we use sampling for contrastive learning. We sampled 1024 points for each dataset.

D.2. Open-Set Semantic Segmentation for LiDAR

Introducing REAL [1], one of the methods for open semantic segmentation (OSS), it is impossible to classify novel categories directly, necessitating a clustering approach. In

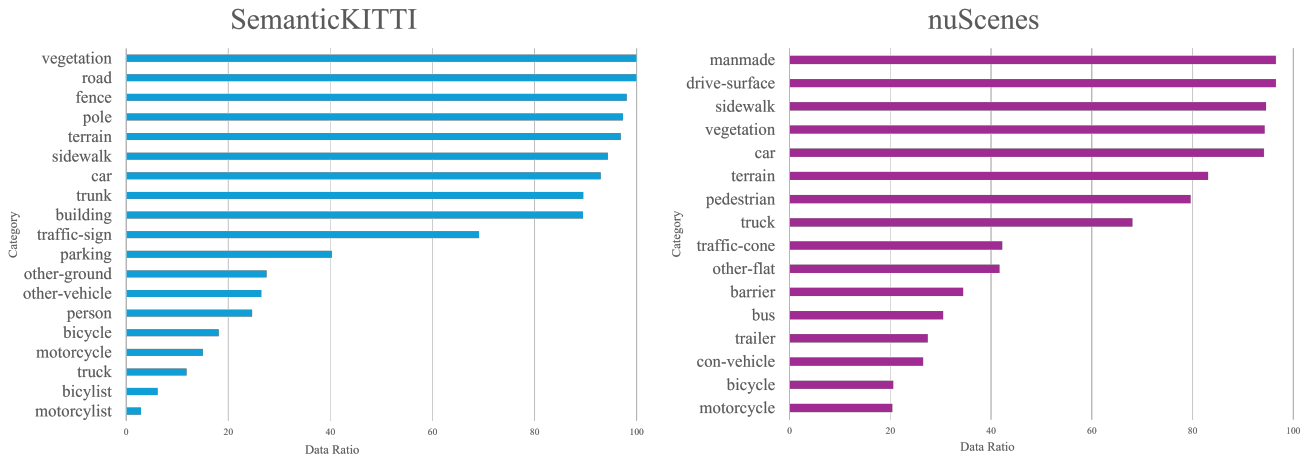

Figure 1. Category-wise data ratio statistics on SemanticKITTI and nuScenes. Frequent categories such as *fence* and *pole* (SemanticKITTI), and *manmade* and *truck* (nuScenes), dominate the labeled point clouds, making them unsuitable as novel categories due to their disproportionate impact on the available labeled supervision. This analysis supports our decision to exclude such categories and to design novel splits with more balanced data distributions.

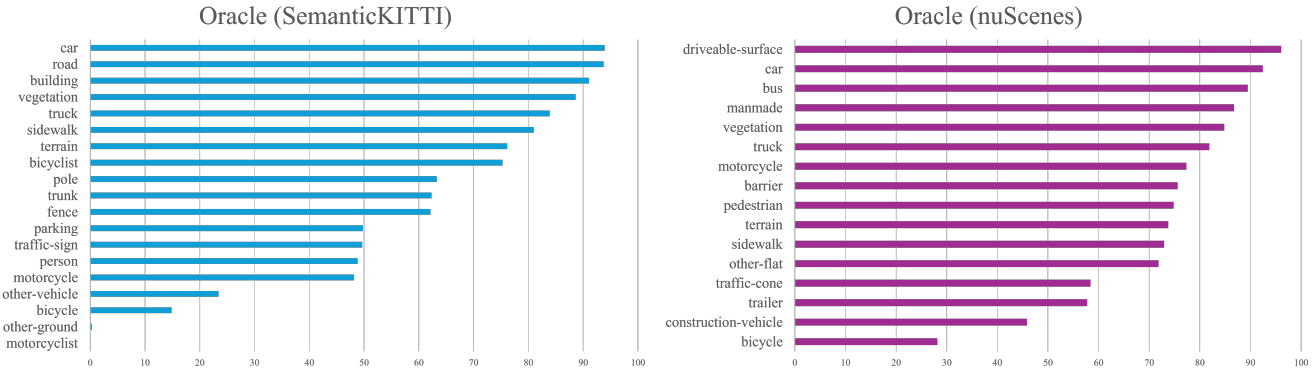

Figure 2. Oracle denotes the upper-bound performance obtained via fully supervised learning using the entire dataset. We observe that certain categories, such as *motorcyclist* and *other-ground* in SemanticKITTI, exhibit consistently poor performance even under oracle settings. In the GCDLSS setting, where labeled data is significantly limited compared to the full dataset, such categories are excluded from the novel set to ensure meaningful and fair evaluation of novel category discovery. Notably, most categories in nuScenes show reasonable oracle performance, and this criterion is applied only to SemanticKITTI.

this case, clustering is performed using a method similar to the one introduced in the "Stream K-means Clustering for OSS Baseline" section. Since known classifiers are trained in REAL, we only need to create the centroids of novel categories and fit them with the test dataset. We apply Mini-batch K-means clustering, setting batch size 2×10^5, and reassignment ratio 0.01.

E. Additional Qualitative Results

This section presents qualitative results on the other splits, which were not covered in the main paper.

E.1. SemanticKITTI Split

As illustrated in Fig. 3 for the *KITTI1* split, our model successfully identified the novel categories. It clearly distinguished *other-vehicle* and also showed better performance on the *bicyclist* category compared to other baselines. In contrast, REAL† and SimGCD† often confused *other-vehicle* with *car*, leading to mixed predictions. NOPS† performed worse, failing almost entirely to detect the *other-vehicle* category. This performance extended to the *KITTI2* split, where our model again excelled on the *other-vehicle* category. Moreover, it effectively discovered the additional novel category *person* with higher accuracy than the baselines. While other models often misclassified *person* as

Figure 3. Qualitative comparison results on the SemanticKITTI dataset (*KITTI1*, *KITTI2*, *KITTI3* splits). The red dashed circles indicate points belonging to the novel category.

geometrically similar objects, such as *pole* or *trunk*, our model, although not perfectly delineating the shape, correctly identified the points as belonging to the *person* category. Throughout these evaluations, our model also maintained strong performance on known categories.

However, the *KITTI3* split showed a different trend. In this challenging setting, REAL† achieved higher overall accuracy on novel categories than our model. While our model showed high quantitative performance on the novel *parking* category, a qualitative inspection revealed its tendency to confuse these areas with adjacent objects. Despite this mixed result on novel categories, our model still maintained the best performance on known categories, showcasing its robustness in preserving existing knowledge.

E.2. nuScenes Split

As illustrated in Fig. 4 for the *NUSC1* split, our model demonstrated strong performance on the *trailer*, and it showed better differentiation of the *bicycle* compared to other baselines. Both REAL† and NOPS† tended to confuse the *bicycle* with *manmade*, while SimGCD† showed a tendency to confuse it with *pedestrian*. In the *NUSC2* split, our model effectively discovered the novel categories, *bus* and *motorcycle*, with higher accuracy than other models.

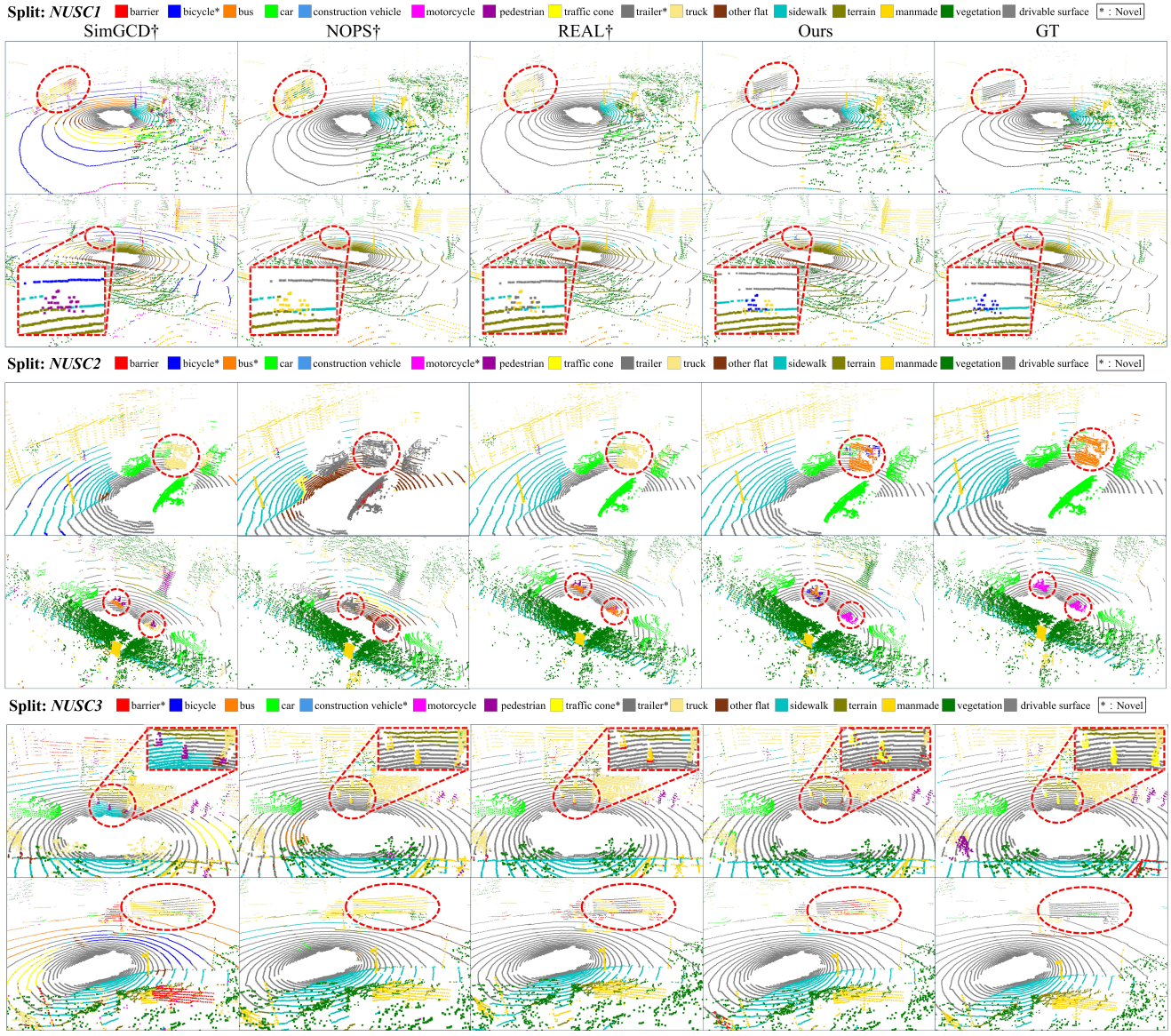

Figure 4. Qualitative comparison results on the nuScenes dataset (*NUSC1*, *NUSC2*, *NUSC3* splits). The red dashed circles indicate points belonging to the novel category.

Moreover, as shown in the first row of the qualitative results for *NUSC2*, our model also demonstrated high accuracy in predicting the *car* adjacent to the *bus*, highlighting its strong performance on known categories as well.

In the more challenging *NUSC3* split, which includes four novel categories and only 25% labeled data, our approach continues to outperform the baselines in both known and novel category performance. As illustrated in Fig. 4, our method shows clear improvements, especially in detecting fine-grained novel categories, such as *traffic cones* and *trailers*, where baselines struggle. While the performance on some other novel categories was lower, our model still exhibited superior accuracy on known categories compared to the baselines. By inspecting regions near the *trailer*, it was evident that our model was particularly adept at predicting known categories, further emphasizing its robustness in challenging scenarios.

However, despite these improvements, our model still faces limitations in consistently excelling across all novel categories. While it performs well on specific categories, a more balanced and generalized performance across all novel categories remains an area for further improvement.

F. Implementation Details

We use the MinkowskiUNet-34C architecture [2] as our backbone. Optimization is performed with SGD (momentum 0.9, weight decay 1×10^{-4}). A learning rate scheduler with linear warm-up and cosine annealing is used, with a maximum learning rate of 1×10^{-2}, a minimum of 1×10^{-5}, and a batch size of 4. The adaptive threshold θ_{adapt} is initialized to 0 in logit space.

Loss Hyperparameters. The loss weights used for each dataset are summarized below:

Dataset	λ_{sup}	λ_{adapt}	λ_{cal}	λ_{semi}	λ_{mse}	λ_{unsup}
SemanticKITTI	1	0.2	0.05	0.1	200	0.1
nuScenes	1	0.5	0.1	0.1	200	0.1

Classifiers. Each classifier is implemented as a linear layer that maps per-point features to logits over the corresponding category set ($\phi_{\text{kn}}, \phi_{\text{ncc}}, \phi_{\text{nov}}$). For the novel candidate classifier, we employ a multi-head architecture (with three heads) to capture diverse notions of novelty [1]. The outputs from the multiple heads are aggregated using a `max` operation across heads. Since the `mean` or `sum` behaves like a logical AND operation—requiring consensus across heads—whereas `max` acts more like a logical OR, allowing any single head to activate on a novel candidate, which encourages the classifier to capture a broader range of potential novel features.

Hardware and Training Configuration. All experiments were conducted using an NVIDIA RTX 3090 GPU with 24GB of memory. To ensure reproducibility, we fixed the random seed to 1234 for all experiments. Since the availability of labeled data differs across datasets, we trained the model for approximately 30 epochs. In cases where the labeled data proportion is smaller, we train the model for a longer period to account for the larger amount of unlabeled data, ensuring sufficient learning from the labeled samples.

References

[1] Jun Cen, Peng Yun, Shiwei Zhang, Junhao Cai, Di Luan, Mingqian Tang, Ming Liu, and Michael Yu Wang. Open-world semantic segmentation for lidar point clouds. In *European Conference on Computer Vision*, pages 318–334. Springer, 2022. 2, 6

[2] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019. 6

[3] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009. 2

[4] Luigi Riz, Cristiano Saltori, Elisa Ricci, and Fabio Poiesi. Novel class discovery for 3d point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9393–9402, 2023. 2

[5] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7492–7501, 2022. 2

[6] Xin Wen, Bingchen Zhao, and Xiaojuan Qi. Parametric classification for generalized category discovery: A baseline study. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16590–16600, 2023. 2