AllGCD: Leveraging All Unlabeled Data for Generalized Category Discovery

Supplementary Material

8. Data Detail

Table 7. Datasets overview. We provide the quantities of labeled ($\mathcal{D}^l$) and unlabeled ($\mathcal{D}^u$) images, along with their respective class distributions.

Dataset	Balanced	Labeled ($\mathcal{D}^l$)		Unlabeled ($\mathcal{D}^u$)	
		#Image	#Class	#Image	#Class
CUB-200-2011 [43]	✓	1.5k	100	4.5k	200
Stanford Cars [18]	✓	2.0k	98	6.1k	196
FGVC-Aircraft [25]	✓	1.7k	50	5.0k	100
CIFAR100 [19]	✓	20.0k	80	30.0k	100
ImageNet-100 [38]	✓	31.9k	50	95.3k	100
Herbarium 19 [37]	✗	8.9k	341	25.4k	683

Table 7 provides a comprehensive overview of the dataset statistics used in our experiments. Notably, generic datasets such as CIFAR-100 and ImageNet-100 contain substantially larger volumes of unlabeled data (e.g., 95.3k in ImageNet) compared to fine-grained datasets like CUB (4.5k) and Stanford Cars (6.1k). This discrepancy in unlabeled sample availability directly influences the selection process governed by the confidence thresholds δ and γ, thereby affecting the overall efficacy of both Intra-CL and Inter-CU in facilitating supervised contrastive learning (S-CL) and novel category discovery.

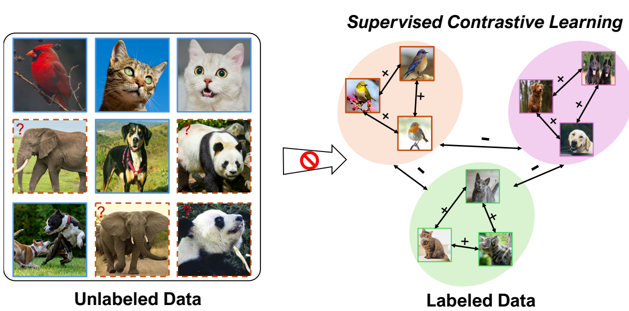


Figure 8. **Supervised contrastive learning (S-CL) without unlabeled data.** Traditional S-CL excludes unlabeled data, which contains both known and novel samples.

Fig. 8 illustrates the data view in traditional supervised contrastive learning (S-CL). Since unlabeled data lacks annotations, S-CL is typically applied only to labeled data. However, labeled data includes only known classes (*e.g.*, bird, dog, cat) and excludes novel classes (*e.g.*, elephant, panda), leading to poor performance on novel classes in parametric GCD methods [3, 44, 45].

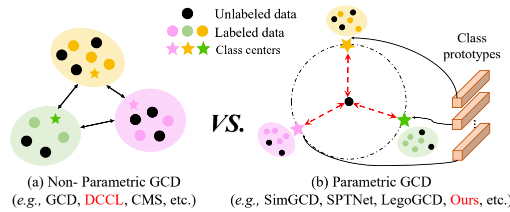


Figure 9. Supervised Contrastive Learning (S-CL) in GCD: Non-Parametric *vs.* Parametric.

9. Paradigm of Our AllGCD

GCD baselines are generally divided into parametric and non-parametric methods. Vaze *et al*. introduced GCD using both unsupervised and supervised contrastive learning (S-CL). Non-parametric approaches like DCCL [51] and CMS [5] follow this setup. In contrast, AllGCD is built on the parametric GCD framework [45], which integrates CL into classifier learning and has shown superior performance. Below, we clarify key distinctions from related works.

Compared to DCCL [51]: (1) **Setting:** DCCL is non-parametric; AllGCD targets the more effective parametric setting (Fig.9). (2) **Motivation:** DCCL models inter-class concepts; we reveal S-CL's limitations in parametric GCD due to scarce labels. (3) **Mechanism:** DCCL aligns semantic concepts, while AllGCD enhances prototype learning via voting-based S-CL using unlabeled data.

Compared to OpenCon [35]: (1) We show S-CL is less effective for novel class discovery in **parametric** GCD (Fig. 9), unlike in non-parametric settings. (2) We are the first to identify this limitation and propose a voting strategy to address it.

10. Hyperparameter Sensitivity Analysis

We analyze the sensitivity of δ and γ on fine-grained and generic datasets. (1) Fig. 10 shows δ works best in $[0.7, 0.75]$ for CUB/SCars and $[0.8, 0.85]$ for CIFAR100/ImageNet-100, while γ in $[0.7, 0.75]$ across all. (2) This shift reflects dataset scale—fine-grained datasets (e.g., 1.5k in Sars *vs.* 31.9k in ImageNet-100) yield fewer confident samples at a higher threshold (e.g., δ=0.85). (3) **Thus**, δ and γ can be selected by dataset type (fine-grained or generic) without dataset-specific tuning. (4) **Meanwhile**, our method is SOTA (except for LegoGCD on ImageNet-100) based on mean within the [0.65–0.85], confirming its robustness.

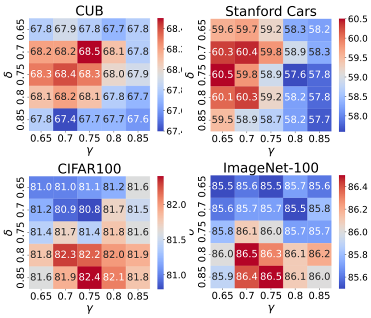


Figure 10. 'All' Acc. w.r.t. δ and γ.

Table 8. 'All' Acc. comparison (with Variance).

Method	Scars	Img-100
GCD	39.0	74.1
DCCL	-	80.5
PromptCAL	50.2	83.1
SimGCD	53.8	83.0
SPTNet	<u>59.0</u>	85.4
LegoGCD	57.3	**86.3**
CMS	56.0	84.7
InfoSieve	55.7	80.5
Ours (Mean)	**59.1**	<u>85.90</u>
Ours (Var.)	$\pm$0.83	$\pm$0.09

Table 9. Classification results on CUB and Stanford Cars with DINOv2 [27].

Method	CUB			Stanford Cars		
	All	Old	New	All	Old	New
k-means	67.6	60.6	71.1	29.4	24.5	31.8
GCD [41]	71.9	71.2	72.3	65.7	67.8	64.7
SimGCD [45]	71.5	78.1	68.3	71.5	81.9	66.6
μGCD [42]	74.0	75.9	73.1	76.1	**91.0**	68.9
SPTNet [44]	76.3	79.5	74.6	-	-	-
AllGCD (ours)	**78.4**	**82.8**	**76.2**	**76.2**	88.3	**70.4**

Table 10. Comparison of parameters and training time across parametric GCD methods.

Method	Parameters				Training Time		
	Backbone	Classifier	Projector	Extra	CUB	Stanford Cars	CIFAR100
SimGCD [45]	✓	✓	✓	✗	4 h 15 m	6 h 2 m	31 h 13 m
LegoGCD [3]	✓	✓	✓	✗	4 h 25 m	6 h 58 m	31 h 40 m
SPTNet [44]	✓	✓	✓	✓	20 h 34 m	41 h 43 m	115 h 2 m
AllGCD (Ours)	✓	✓	✓	✗	5 h 52 m	7 h 41 m	34 h 41 m

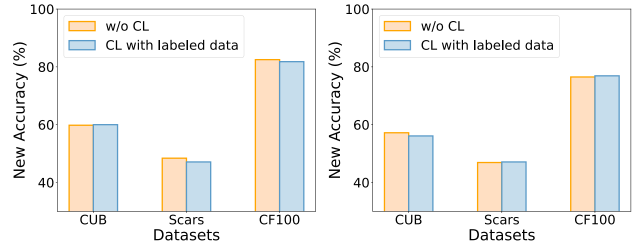
11. Results with DINOv2

We further validate the effectiveness of AllGCD by employing the ViT-B/14 model pre-trained with DINOv2 [27] on the CUB and Stanford Cars datasets. Our method consistently outperforms recent state-of-the-art approaches, such as μGCD and SPTNet, demonstrating its robustness across different architectures. Specifically, on CUB, AllGCD improves novel class discovery accuracy by **7.9%** compared to the baseline SimGCD and by **6.9%** relative to SPTNet. On Stanford Cars, it enhances novel class discovery by **3.8%** and surpasses μGCD by **1.5%** in 'New' accuracy. These results highlight the adaptability and superior performance of AllGCD, reinforcing its efficacy in novel category discovery even when applied to alternative backbone.

12. Results on Other Parametric Methods

12.1. Limited S-CL in parametric methods

In this section, we explore novel class discovery with and without supervised contrastive learning (S-CL) in two competitive parametric methods, LegoGCD [3] and SPTNet [44]. Overall, both methods fail to improve the accuracy of novel classes when CL is restricted to labeled data. From the orange and blue bars in Fig. 11a and Fig. 11b, we can see that the 'New' class accuracy does not improve with CL in either LegoGCD or SPTNet. Furthermore, we plot the 'New' class accuracy during training in Fig. 11a and Fig. 11b. Obviously, the accuracy with CL shows little difference compared to without CL, as the lines nearly overlap. Notably, we reproduced SPTNet using SimGCD models, achieving 'Old'/'New' results of 78.4%/56.1% on CUB, 78.9%/47.4% on Stanford Cars, and 83.1%/76.0% on CIFAR100. CL was then removed from these models to generate Fig. 11b.



(a) LegoGCD (b) SPTNet

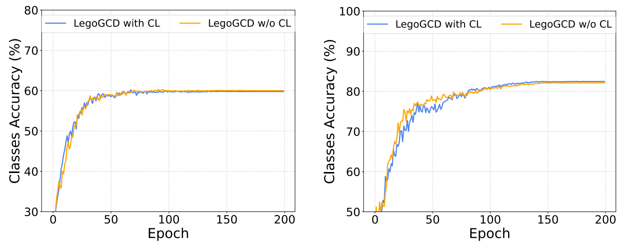
Figure 11. **'New' class accuracy in LegoGCD and SPTNet with and without supervised contrastive learning (S-CL)**. The accuracy remains nearly unchanged, highlighting that CL using only labeled data is insufficient for classifying novel classes in previous parametric GCD methods.

12.2. Integration of AllGCD with LegoGCD

To further substantiate the effectiveness of AllGCD, we integrate its proposed components into other parametric methods, such as LegoGCD [3]. As presented in Table 11, our approach consistently enhances the accuracy of both 'Old' and 'New' classes across fine-grained and generic datasets. Specifically, AllGCD achieves performance gains of 1.6% and 3.5% on CUB, 1.5% and 3.1% on Stanford Cars, and 1.6% and 0.3% on CIFAR-100 compared to LegoGCD. These results underscore the importance of leveraging a larger volume of unlabeled data, which substantially strengthens contrastive learning and enhances novel category discovery.

13. Parameters and Training time Analysis

Table 10 provides a comprehensive comparison of the parameter count and training efficiency across various parametric methods, including LegoGCD and SPTNet. A ✓ indicates shared components, while SPTNet incorporates additional data prompts. While our approach moderately increases training time—requiring 1 h 37 m on CUB, 1 h 39 m on Stanford Cars, and 3 h 18 m on CIFAR-100 compared to



(a) CUB

(b) CIFAR100

Figure 12. **'New' class accuracy during training in LegoGCD with and without S-CL.** The accuracy shows no significant change regardless of the use of S-CL in both CUB and CIFAR100.

Table 11. Classification results of LegoGCD[3] combined with our components on fine-grained and generic datasets.

Datasets	AllGCD	*ACC*		
		All	Old	New
CUB		63.8	71.9	59.8
CUB	✓	**66.7**+2.9	**73.5**+1.6	**63.3**+3.5
Stanford Cars		57.3	75.7	48.4
Stanford Cars	✓	**59.9**+2.6	**77.2**+1.5	**51.5**+3.1
CIFAR100		81.8	81.4	82.5
CIFAR100	✓	**82.5**+0.7	**83.1**+1.6	**82.8**+0.3

the baseline SimGCD—it remains substantially more efficient than SPTNet, which requires over **five times** the training time of SimGCD. Despite this efficiency, our method achieves highly competitive performance, striking an optimal balance between classification accuracy and computational cost.