Supplementary Material for PSS: Progressive Sample Selection for Open-World Visual Representation Learning

Tianyue Cao* , Yongxin Wang† , Yifan Xing , Tianjun Xiao , Tong He , Zheng Zhang , Hao Zhou , and Joseph Tighe

Amazon Web Services
vanessa_@sjtu.edu.cn, {yongxinw, yifax, tianjux, htong, zhnzhe, zhouho, tighej}@amazon.com

1 Compare with More Existing Methods

We compare our method with more semi-supervised learning and novel class discovery baselines, including PRL$^+$[14], ORCA$^+$[2] and GCD[12]. Also, we apply different backbone architectures on our method to compare with different existing methods fairly. Furthermore, we validate the effectiveness of PSS over the more advanced self-supervised methods for model pretraining, such as SwAV [4] and DINO[5]. Lastly, we experiment with other clustering algorithms of HAC/DBSCAN to be used within our PSS progressive design.

Baselines with ResNet-50 backbone architecture We train PRL$^+$ and ORCA$^+$ with a ResNet-50 backbone. We adapt PRL to the open-world representation learning setting and denote it as PRL$^+$. PRL uses in-degrees of nodes in a k-nearest-neighbors (k-NN) graph built over input features as the sample selection criterion. Thus, we replace the sample selection criterion of PSS to node in-degrees for PRL$^+$. For ORCA$^+$, we train an ORCA model to generate pseudo labels for $\mathcal{U}$ and re-train the feature extractor with Smooth-AP loss on $\mathcal{L}_{\text{in}} \cup \mathcal{U}$. We adapt ORCA to our open-world representation learning setting in this way because it only has $\mathcal{C}_{\text{in}}$ and $\mathcal{C}_{\text{out}}$ classification heads but not the $\mathcal{C}_{test}$ classification head and in our practical setting, test set novel classes and unlabeled samples are unavailable during training. Table 2 shows the retireval performance of PSS, PRL$^+$, and ORCA$^+$ over the iNaturalist benchmark, all of which use ResNet-50 as the feature extractor backbone. Benefiting from the progressive pipeline and sample selection method, PSS outperforms the other two existing methods. Also, though PRL$^+$ also improves across iterations (Recall@1: 0.5376 $\rightarrow$ 0.5433 $\rightarrow$ 0.5458 $\rightarrow$ 0.5465), PSS with its homogeneity density metric significantly outperforms this alternative density metric, achieving Recall@1 0.5714 as opposed to 0.5465, demonstrating the effectiveness of our *sample selection* method.

*Currently at The Shanghai Jiao Tong University. Work conducted while at AWS.
†Corresponding author.

Table 1: iNaturalist challenging dataset split: attributes of the labeled, unlabeled and test set

split	#images	#classes	#super-classes
$\mathcal{L}_{\text{in}}$	25,440	948	7
$\mathcal{U}$	300,406	5,690	14
test	136,093	2,452	13

Table 2: Retrieval performance of PSS and existing methods using ResNet-50 backbone on iNaturalist

method	backbone	Recall@1	Recall@4	Recall@16	Recall@32
PRL$^+$[14]	ResNet-50	0.5465	0.7172	0.8352	0.8788
ORCA$^+$[2]	ResNet-50	0.5413	0.7187	0.8395	0.8830
PSS	ResNet-50	0.5714	0.7357	0.8501	0.8914

Baselines with ViT backbone architecture We train GCD with a 16 patch size ViT-Base backbone pretrained with DINO[5] on ImageNet. To compare with GCD fairly, we also train PSS with the same backbone. Using the same dataset split in the main paper, the retrieval performance of the supervised baseline (with $\mathcal{L}_{\text{in}}$ only) in Recall@1 is 0.6822, which is already similar to the oracle performance (0.6823) using all ground truth of $\mathcal{L}_{\text{in}} \cup \mathcal{U}$. We believe it is due to the randomly sampled $\mathcal{L}_{\text{in}}$ from all of the super categories in iNaturalist already being a strongly representative training set over a higher-capacity vision transformer backbone and more generalized DINO pretrained feature. Thus, we extend our dataset split to a even more practical and challenging setting as follows.

The fine-grain categories over all data in iNaturalist are from 14 super-classes, we randomly sample about 16% of the training classes in 7 super-classes instead of all super-classes as the seen classes $\mathcal{C}_{\text{in}}$, and take 60% of the images from each class in $\mathcal{C}_{\text{in}}$ as the labeled training set $\mathcal{L}_{\text{in}}$. The attributes of the new dataset split is shown in Table 1. Compared with the previous dataset split, $\mathcal{L}_{\text{in}}$ in the new split has larger distribution gap with $\mathcal{U}$ and the test set, thus more challenging.

We validate the performance of GCD and PSS using ViT backbone on this more challenging dataset split, see Table 3. PSS outperforms GCD and the progressive method generalizes to the stronger vision transformer backbone. It also validates the robustness of PSS over different backbones. Note that the supervised baseline outperforms GCD because different feature extractor training losses are used (smooth-ap [1] for the supervised baseline as opposed to noise contrastive [8] and supervised contrastive [9] for GCD).

More advanced self-supervised methods for pretraining We also apply PSS over supervised baselines pretrained with more advanced self-supervised approaches such as SwAV [4] and DINO[5] to test its generalization. In particular, we adopt these unsupervised methods via self-supervised pretraining followed-by fine-tuning over our labeled train set, similar to the adaptation of DINO for the supervised baseline over ViT backbones in GCD [12]. The nature image retrieval performance improvement of PSS (0.6699) over the supervised baseline (DINO[5] pretrained, 0.6550) shown in Table 3 illustrates the superior generalization capability of PSS over stronger pretrained features. Furthermore, Table 4 demonstrates the progressive retrieval performance improvement in PSS over the supervised baseline pretrained with SwAV [4] through self-supervision. Here

we use the same challenging dataset split as the one in DINO pretraining and the GCD [12] experiment above. It shows that PSS is able to generalize over SwAV[4] pretrained supervised baseline models as well.

Other clustering methods We also experimented with other clustering algorithms of HAC/DBSCAN in the PSS progressive design (Table 5). Here, we keep all other design choices the same and only alter the clustering method. It is observed that alternative clustering methods (such as HAC, achieving Recall@1: $0.5376 \rightarrow 0.5479 \rightarrow 0.5502$ across iterations) are also applicable to our PSS framework for progressive performance improvements while HiLANDER brings the most gain. In addition, high computational efficiency is essential for PSS because it runs clustering and feature refinement over multiple iterations. With HAC/DBSCAN, the clustering run-time is prohibitively expensive across iterations. Thus, we select HiLANDER to be used with PSS due to its high clustering pseudo-label quality and fast runtime.

Table 3: Retrieval performance of PSS and GCD using ViT-Base/16 backbone on iNaturalist. PSS outperforms GCD

method	backbone	Recall@1	Recall@4	Recall@16	Recall@32
sup. baseline (DINO[5] pretrain)	ViT-Base/16	0.6550	0.7993	0.8895	0.9214
GCD[12]	ViT-Base/16	0.6203	0.7763	0.8774	0.9130
PSS	ViT-Base/16	0.6699	0.8126	0.9006	0.9306

Table 4: Retrieval performance of PSS over the supervised baseline with SwAV[4] pretraining on iNaturalist. PSS generalizes over SwAV[4] pretrained models

method	backbone	Recall@1	Recall@4	Recall@16	Recall@32
sup. baseline (SwAV[5] pretrain)	ResNet-50	0.5384	0.7021	0.8218	0.8678
PSS iter1	ResNet-50	0.5385	0.7023	0.8226	0.8684
PSS iter2	ResNet-50	0.5402	0.7047	0.8235	0.8696
PSS iter3	ResNet-50	0.5415	0.7051	0.8251	0.8706

Table 5: Retrieval performance of PSS with other clustering methods over iNaturalist

method	backbone	Recall@1	Recall@4	Recall@16	Recall@32
sup. baseline	ResNet-50	0.5376	0.7135	0.8359	0.8817
PSS w. DBSCAN	ResNet-50	0.5477	0.7227	0.8409	0.8847
PSS w. HAC	ResNet-50	0.5502	0.7230	0.8435	0.8864
PSS w. HiLANDER	ResNet-50	0.5714	0.7357	0.8501	0.8914

Table 6: Training efficiency and accuracy comparison of PSS and prior works on the nature species retrieval and face verification benchmarks

Benchmark	**Method**	$\mathcal{U}$ **selection criteria**	**Total Training Samples**	**Performance (R@1 / FNMR@FMR1e-4)**
iNaturalist	PL [10]	All	354,857	0.5447
	Hi-LANDER [13]	All	354,857	0.5421
	UNO [7]	All	651,692	0.5372
	Deep Clustering [3]	All	1,006,549	0.5548
	PRL$^+$ [14]	Node Indegree	507,055	0.5465
	ORCA$^+$ [2]	All	651,692	0.5413
	PSS	Density	**262,937**	**0.5714**
IJBC	DBSCAN [6]	All	8,554,382	0.4203
	GCN-V [15]	All	8,554,382	0.2508
	Hi-LANDER [13]	All	8,554,382	0.2472
	RoyChowdhury et al [11]	Identity Disjoint Set	**2,706,271**	0.2706
	Deep Clustering [3]	All	15,983,890	0.2234
	PSS	Density	6,939,329	**0.2165**

2 Efficiency and Accuracy Comparison

In Table 6, we list the unlabeled set sample selection criteria, total number of training samples and corresponding accuracy for all compared prior works. Total training samples stands for a summation of samples used across all training steps (one-time sample selection or iterative) of pseudo-labeler and feature extractor. For instance, PL [10] trains a pseudo-labeler with $29,011$ images from $\mathcal{L}_{\text{in}}$ and a feature extractor with $325,846$ images from $\mathcal{L}_{\text{in}} \cup \mathcal{U}$, ending up with a total number of $354,857$. Table 1 of the main paper shows the dataset attributes over nature species retrieval and face verification. With the same feature extraction backbone, these numbers are proportional to the training efficiency of compared methods. On iNaturalist, PSS is the most training efficient method while achieving the highest Recall@1. Similarly, on the IJBC face verification benchmark, PSS utilizes the second least number of training samples to obtain the lowest FNMR. Note that RoyChowdhury [11] selects from unlabeled set $\mathcal{U}$ only once.

3 Qualitative Visualization

We qualitatively visualize some training set samples $\mathcal{T}$ and selected samples $\mathcal{U}_{\text{selected}}$ which are "far" and "close" to $\mathcal{T}$ in the 1st iteration of PSS, see Figure 1.

We use the new dataset split in Section 1 for the qualitative visualization here. Note that "far" and "close" $\mathcal{U}_{\text{selected}}$ has large and small distance-to-closest-labeled-class respectively. It is observed that some "far" $\mathcal{U}_{\text{selected}}$ samples are from super-classes *Aves, Animalia*, and *Plantae*, which are not in the existing labeled training set $\mathcal{T}$. It shows that our density-based sample selection method has the ability to include some far-away samples to the existing labeled training set, thus can expand the feature space and improve representation generalization.

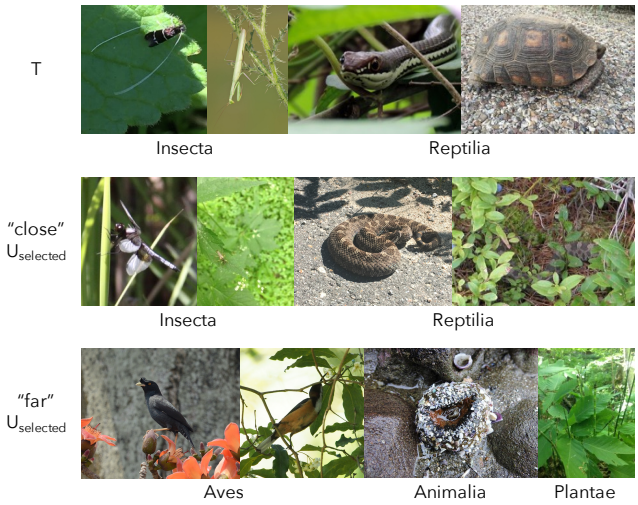

Fig. 1: Some qualitative visualization of selected samples $\mathcal{U}_{\text{selected}}$ which are "far" and "close" to the training set $\mathcal{T}$ in terms of distance-to-closest-labeled-class

4 FAQ

Why does DeepClustering for face verification stop after iteration 2? It's highly costly to run DeepClustering for face verification beyond iteration 2. Even at the 2nd iteration, DeepClustering is already greater than 2 times more costly than PSS as it requires more than twice number of samples over its iterative feature training process, as shown in Table 6. We also found that if we run DeepClustering to its original stopping criterion for face verification, though the error rate can get further reduced (0.2234 to 0.2014 FNMR@FMR1e-4), the cost grows further, with 3 times (23,413,398 training samples) of the cost of PSS (6,939,329 training samples).

References

1. Brown, A., Xie, W., Kalogeiton, V., Zisserman, A.: Smooth-ap: Smoothing the path towards large-scale image retrieval. In: European Conference on Computer Vision. pp. 677–694. Springer (2020)
2. Cao, K., Brbic, M., Leskovec, J.: Open-world semi-supervised learning. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=O-r8LOR-CCA
3. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: ECCV (2018)
4. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments (2020)
5. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the International Conference on Computer Vision (ICCV) (2021)
6. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. p. 226–231. KDD'96, AAAI Press (1996)
7. Fini, E., Sangineto, E., Lathuilière, S., Zhong, Z., Nabi, M., Ricci, E.: A unified objective for novel class discovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9284–9292 (October 2021)
8. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 297–304. JMLR Workshop and Conference Proceedings (2010)
9. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. Advances in Neural Information Processing Systems **33**, 18661–18673 (2020)
10. Lee, D.H., et al.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning, ICML. vol. 3, p. 896 (2013)
11. RoyChowdhury, A., Yu, X., Sohn, K., Learned-Miller, E., Chandraker, M.: Improving face recognition by clustering unlabeled faces in the wild. In: European Conference on Computer Vision. pp. 119–136. Springer (2020)
12. Vaze, S., Han, K., Vedaldi, A., Zisserman, A.: Generalized category discovery. arXiv preprint arXiv:2201.02609 (2022)
13. Xing, Y., He, T., Xiao, T., Wang, Y., Xiong, Y., Xia, W., Wipf, D., Zhang, Z., Soatto, S.: Learning hierarchical graph neural networks for image clustering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3467–3477 (October 2021)
14. Yan, X., Chen, R., Feng, L., Yang, J., Zheng, H., Zhang, W.: Progressive representative labeling for deep semi-supervised learning. arXiv preprint arXiv:2108.06070 (2021)
15. Yang, L., Chen, D., Zhan, X., Zhao, R., Loy, C.C., Lin, D.: Learning to cluster faces via confidence and connectivity estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)