Supplementary Material
PT4AL: Using Self-Supervised Pretext Tasks for Active Learning

This supplementary material includes contents which are not included in the main paper due to space limit.

A. Hyper-parameters Image Classification Experiments

Table 1 presents the hyper-parameters used in Section 5.1 of the main paper.

Table 1. Hyperparameters used in image classification experiments for PT4AL. lr is the learning rate for the pretext and main tasks. Batch size and epochs are also divided into pretext task and main task parameters

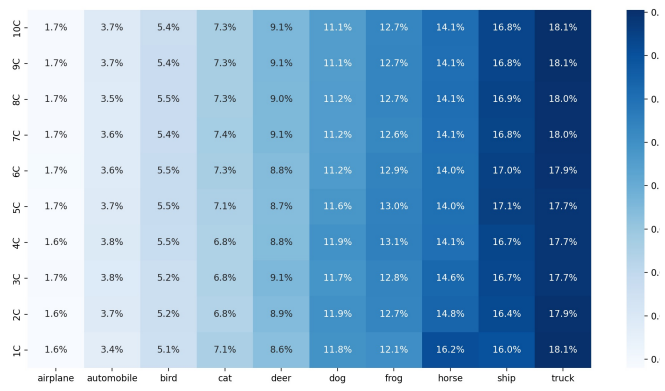
Dataset	lr (pretext/main)	batch size (pretext/main)	epochs (pretext/main)	initial/budget	image size
CIFAR10	0.1 / 0.1	256 / 128	120 / 200	1000 / 50000	32 x 32 (padded)
Caltech-101	0.01 / 0.1	64 / 64	50 / 100	1000 / 8046	224 x 224 (cropped)
ImageNet	0.1 / 0.1	256 / 256	150 / 100	127986 / 1279867	224 x 224 (cropped)

B. Details on Imbalanced Dataset Experiment

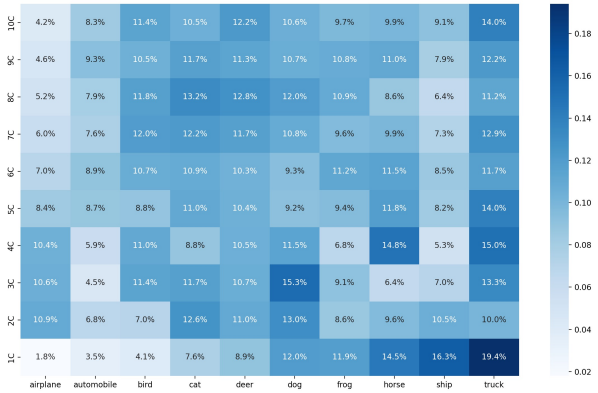
This section corresponds to Sec. 5.3 of the main paper.

Fig. 1 is a heatmap of the class distribution cumulative distribution of data extracted in each cycle by Random, VAAL [8], and PT4AL (Rotation [3]) methods in imbalanced CIFAR10. The x axis represents each of the ten classes, and the y axis represents each active learning cycles. As explained in the main paper, the number of data for each class increases as we go from class "airplane" to "truck".

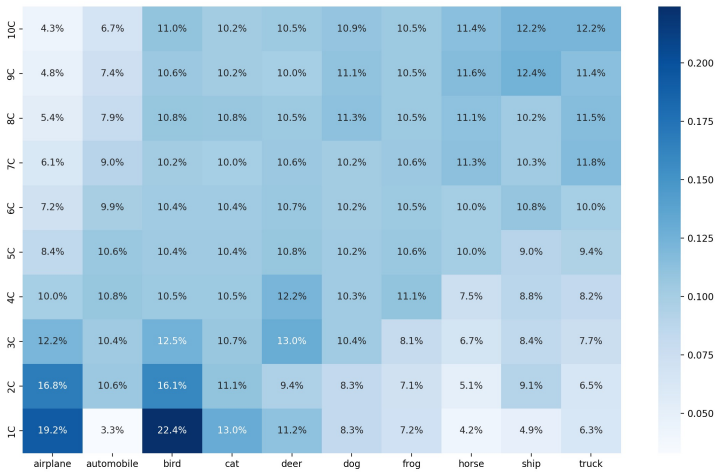
As shown in the figure, Random sampling demonstrates that the distribution of data extracted in every cycle follows the class distribution of the unlabeled imbalanced dataset. On the other hand, VAAL extracts data while relatively considering class balance, but does not completely solve the class imbalance problem. Unlike existing methods, PT4AL extracts data with severe class imbalance in the initial cycles, and moves on to sample in a class-balanced way as the cycle progresses. Our PT4AL method can alleviate the class imbalance problem by using the loss-based sampling method to match class balance in imbalanced datasets.



(a) Random



(b) VAAL



(c) PT4AL(Rotation)

Figure 1. Cumulative class distribution heatmap of (a) Random, (b) VAAL [8] and (c) PT4AL (Rotation [3]) methods in the imbalanced CIFAR10 [6] dataset

C. Cold Start

This section corresponds to Sec. 5.4 of the main paper.

Fig. 2 presents a box plot using the results of 20 experiments in the first iteration of the PT4AL (Rotation) and Random sampling methods on the imbalanced CIFAR10 dataset. As shown in the figure, our method has a smaller difference between the minimum and maximum accuracies than the random method, and it can be seen that the difference between Q1, Q2 and Q3 is also small. Therefore, our PT4AL (Rotation) confidently alleviates the cold start problem by having better starting accuracy with small variance, compared to the existing methods that use random sampling in the first sampling iteration.

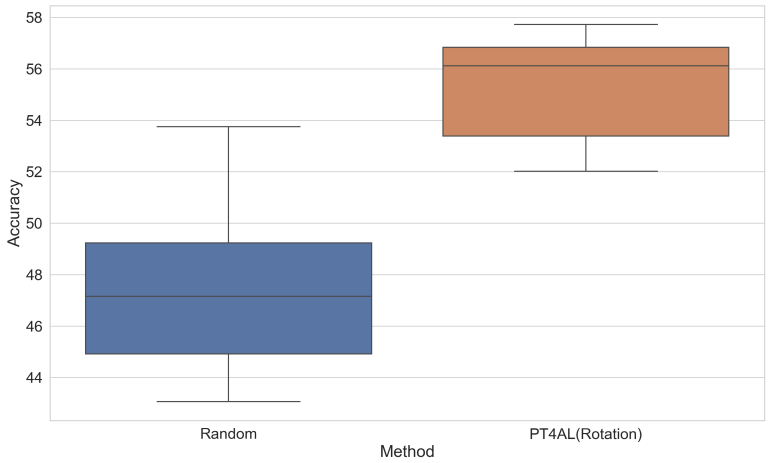


Figure 2. Results of first iteration main task accuracies across 20 experiments for random sampling and PT4AL(Rotation) on CIFAR10

D. Class Distribution for PT4AL(SimSiam [2])

This section corresponds to Sec. 6.2 of the main paper. In the main paper we compare (Fig. 6.a, 6.b) different pretext tasks used for PT4AL. From the comparison we discover that SimSiam [2], one of the most recent papers that use contrastive learning [4] (or joint embedding since it does not use negative examples) to learn visual features, performs poorly on our main task compared to simple hand-designed pretext tasks. [3, 7, 9] The contrastive loss used in SimSiam displays very weak correlation ($\rho = -0.001$) to the main task loss. To examine the cause of this behavior we examine the class distribution of each of the ten batches created for sampling. Fig. 3 illustrates the class distribution for the first, fifth, and tenth batches. Each batches contain 5000 images sampled by their pretext loss rank. Batch 1 has images with the highest ranking losses, and batch 10 has the lowest losses. We can examine that most samples are focused on a few classes, and the biased classes differ for every batch. We discover such behavior from all ten batches in the CIFAR10 [6] experiment. This biased class distribution results in data from a select few classes being sampled in each iteration, which is detrimental for the main task learning performance. PT4AL using SimSiam performs worse than the random baseline which evenly samples data from all clsses. We discover that unlike other pretext tasks, SimSiam and other works [1, 5] that use contrastive learning have loss ranks that vary by class. We contribute this behavior to the losses' tendency to bring positive image pairs closer and repel negative pairs, which explains the loss "clustering" of specific classes. We also suspect that this class-biased tendency is exacerbated with SimSiam's implementation because it does not use negative sets: it only looks at augmentations of the same image to minimize contrastive loss. Due to the class bias, combined with high variance from image augmentations and long training time, we deem contrastive learning tasks are not fit for PT4AL's pretext task model.

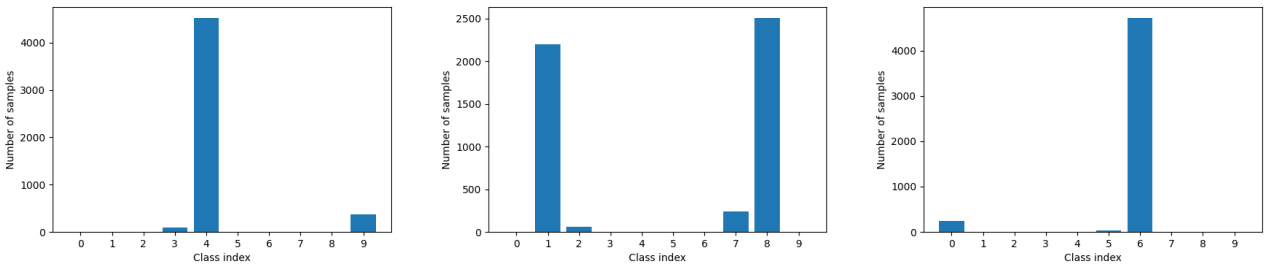


Figure 3. Class distribution of sampled data from the first, fifth, and tenth iterations using the losses extracted from the SimSiam pretext task learner

E. Details on Sampling Strategies

E.1. Sampling in the first iteration

This section corresponds to Sec. 6.3 of the main paper. Fig. 4 demonstrates the samples extracted through top-k loss and uniform sampling from the batch with the lowest rotation pretext task loss in the CIFAR10 dataset. When using the top-k method, images with similar visual features are extracted, while uniform samples more diverse data. In particular, in classes such as deer, airplane, and bird, it can be empirically confirmed that the Top-k method tends to extract visually overlapping samples. Uniform sampling, in contrast, samples relatively diverse data points with different color, shape, and orientation. For example, in the Truck class in Fig. 4 we can observe that top-k samples blueish trucks facing left, while images sampled by uniform sampling face different directions and have different colors. Since the top-k method extracts data with overlapping information, uniform sampling is used in the first iteration of PT4AL to avoid this problem.

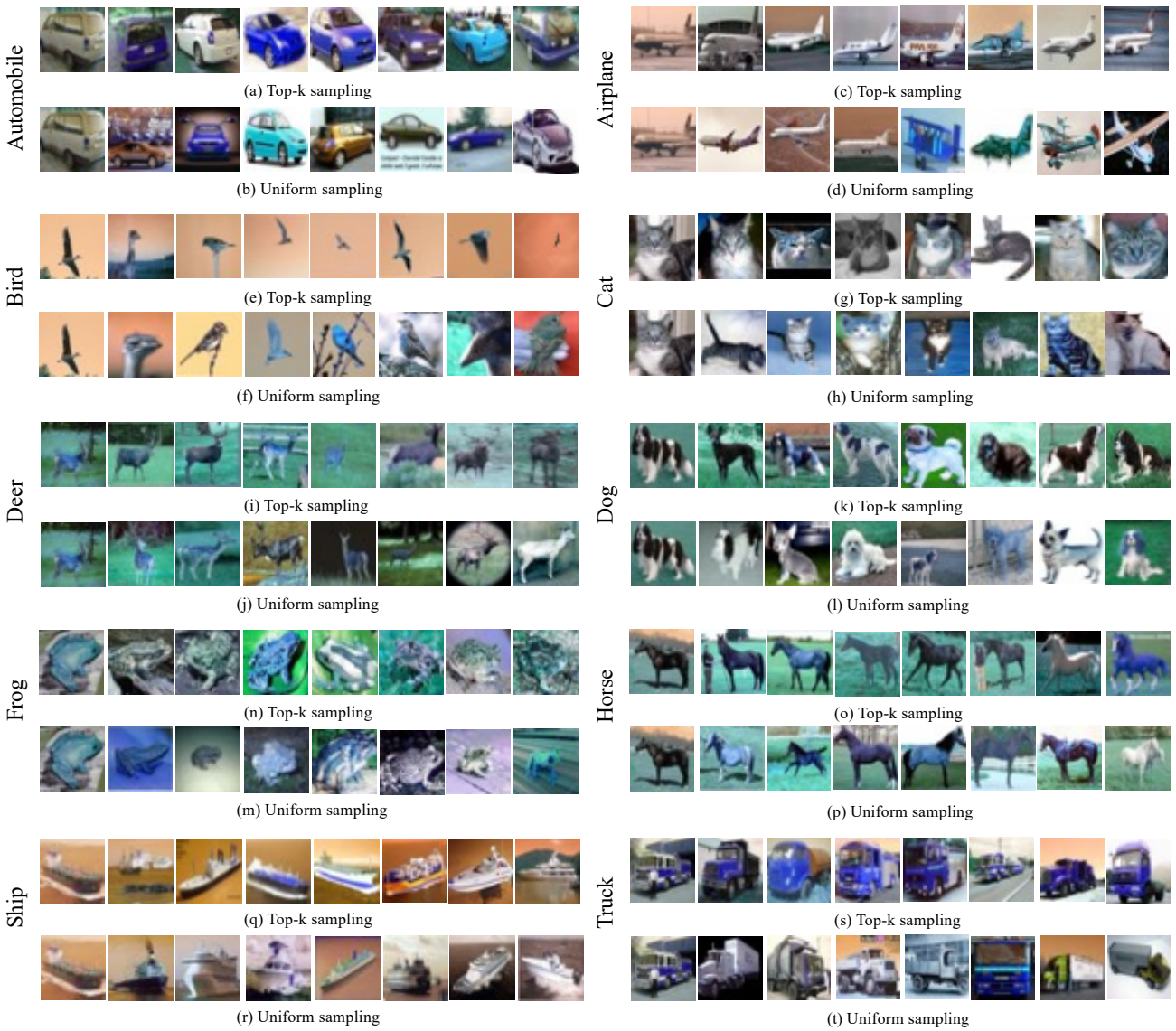


Figure 4. Image data extracted using top-k and uniform sampling from the batch with the lowest pretext task losses. Best viewed in color

E.2. Sampling Method Comparison

This section supplements Sec. 6.3 of the main paper. In this section different sampling methods are explored. Unlike sampling in the first iteration, the methods explored is to be used in the second iteration onwards, when the trained main task model from the previous iteration is available. We compare three different sampling methods: class balanced sampling, confidence based sampling, and entropy based sampling. Class balanced sampling simply samples data from the given batch in a class-balanced way. It uses pseudo-labels from the previous main task model to assign each data to a class, and samples 100 data points from each of the ten classes. If there are less than 100 data points in a class, the sampler supplements the remaining for data from another class. Confidence sampling uses the top-1 posterior probability from each data point and samples data with the lowest 100 top-1 probability. Entropy based sampling samples 100 data with the highest entropy. The main task model is used to extract both top-1 probability and entropy for the methods. Fig. 5 displays the main task results for the three sampling methods. We can see that both entropy and confidence based methods perform similarly, while class balanced sampling under-performs the others. We choose to use confidence sampling since it queries data in the decision boundary of the main task model. Fig. 6 illustrates main task results for confidence-based sampling using batches sorted in ascending or descending order. Overall, sampling from batches with high pretext task losses first outperforms the low loss batch first sampling method.

E.3. High loss first vs Low loss first sampling

Fig. 6 demonstrates the implementations in the main paper using high to low loss sampling and low to loss sampling. While the performances are similar, we observe that sampling from batches with high pretext task loss values and moving on to batches with lower losses perform better especially in the initial iterations. Thus, we use the high-to-low sampling method in the main paper.

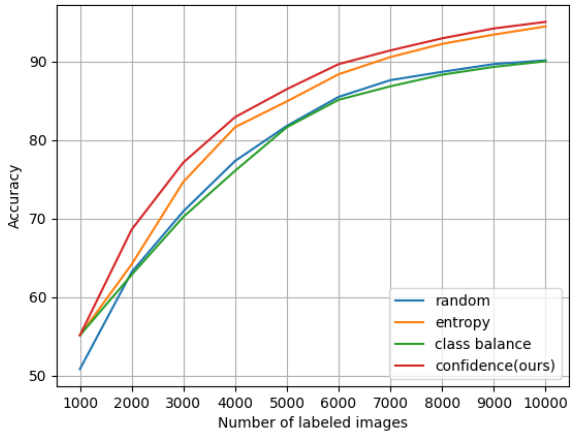


Figure 5. Results of PT4AL using different in-batch sampling methods on CIFAR10

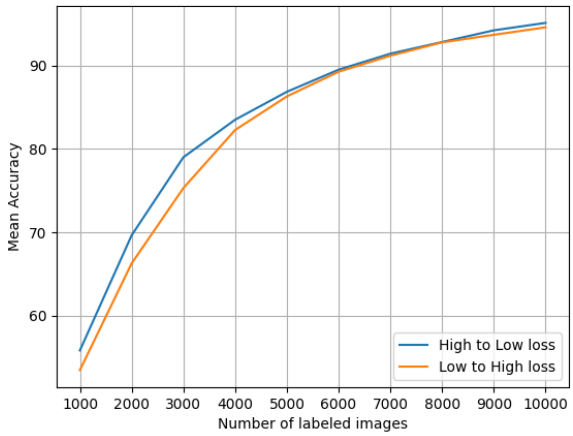


Figure 6. Comparison of main task performance between PT4AL(rotation) with losses sorted by low loss first and high loss first

F. Combination of Pretext Tasks

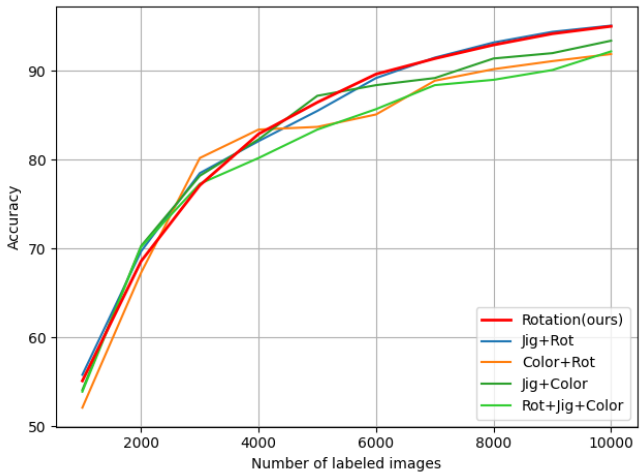


Figure 7. Comparison of different pretext task combinations on CIFAR-10

Fig. 7 shows an experiment of different combinations of pretext tasks on CIFAR-10. The three pretext tasks (jig: jigsaw puzzle, rot: rotation, color: colorization) are combined by summing the pretext task loss rankings. The results indicate that only jigsaw + rotation shows a marginal improvement over the baseline which only uses rotation prediction. Although combining multiple pretext tasks may be synergistic, we think the cost of training multiple tasks is not valuable due to the marginal performance benefit.

References

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 4

[2] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021. 4

[3] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018. 2, 4

[4] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006. 4

[5] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 4

[6] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 4

[7] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016. 4

[8] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981, 2019. 2

[9] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016. 4