Class-Incremental Learning with Cross-Space Clustering and Controlled Transfer: Supplementary Material

Arjun Ashok[1,2], K J Joseph[1], and Vineeth N Balasubramanian[1]

[1] Indian Institute of Technology Hyderabad
[2] PSG College Of Technology, Coimbatore
arjun.ashok.psg@gmail.com, {cs17m18p100001, vineethnb}@iith.ac.in
https://cscct.github.io

1 Results on Different Batch Sizes

Our method leverages examples from the entire batch in order to facilitate training. All our results on CIFAR100 in the original paper are reported using a batch size of 128, following prior work [3, 1]. Here, we analyze the efficacy of our method on different batch sizes used during training. Prior work has observed that increasing the batch size tends to slightly reduce the test accuracy, and have proposed various methods to reduce the drop in accuracy [4, 2, 5]. In our experiments, we adopt the linear scaling rule [2], which scales the learning rate in proportion to the batch size.

Table S1 showcases the results on two different experimental settings, across various batch sizes. It can be seen that increasing the batch size increases the performance of our method. This is because larger batch sizes allow using more samples from the memory as well as the current task, providing a broader view of the feature space, which directly benefits our objectives.

Table S1. Ablation studies on the Batch Size

Settings		$\mathcal{B} = 50$	$\mathcal{B} = \mathcal{C}$
Methods	Batch Size	$\mathcal{C} = 5$	
LUCIR [3]	128	59.4	42.28
LUCIR + CSCCT		$60.01_{+2.61}$	$44.03_{+1.55}$
LUCIR [3]	256	57.06	39.58
LUCIR + CSCCT		$59.71_{+2.91}$	$41.48_{+1.90}$
LUCIR [3]	512	56.19	38.16
LUCIR + CSCCT		$59.02_{+2.83}$	$40.2_{+2.04}$
LUCIR [3]	1024	54.83	37.97
LUCIR + CSCCT		$57.8_{+2.97}$	$40.27_{+2.3}$

2 Results on Different Exemplar Memory Sizes

The memory size specifies the *exemplars-per-class* that the model can store at the end of each phase. Here, we report results varying the exemplar memory size.

Table S2 showcases the results. Enforcing stricter memory constraints causes a performance drop in LUCIR [3], however, our method still provides strong relative improvements across settings. As the memory size increases, our method offers greater relative improvements.

Table S2. Ablation studies on the Memory Size

Settings		$\mathcal{B} = 50$	$\mathcal{B} = \mathcal{C}$
Methods	Memory Size	$\mathcal{C} = 5$	
LUCIR [3]	10	55.83	39.45
LUCIR + CSCCT		57.54 **+1.71**	40.72 **+1.14**
LUCIR [3]	20	59.4	42.28
LUCIR + CSCCT		60.01 **+2.61**	44.03 **+1.55**
LUCIR [3]	30	62.52	46.35
LUCIR + CSCCT		65.05 **+2.53**	48.34 **+1.99**
LUCIR [3]	40	63.60	50.16
LUCIR + CSCCT		66.49 **+2.89**	52.28 **+2.12**

3 Phase-Wise Plots

Figures A1 and A2 showcase phase-wise plots on three class-incremental learning settings on CIFAR100, on top of multiple baseline methods.

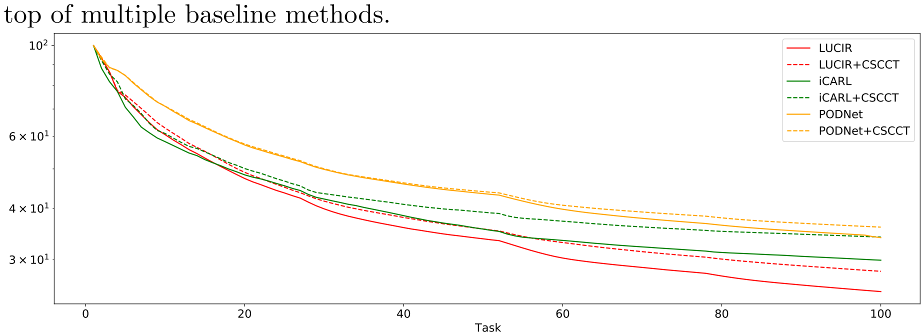

Fig. A1. Phase-wise average incremental accuracies on CIFAR100, on the 100 Task Setting with 1 Class per Task. The y-axis is set to log scale for visual clarity.

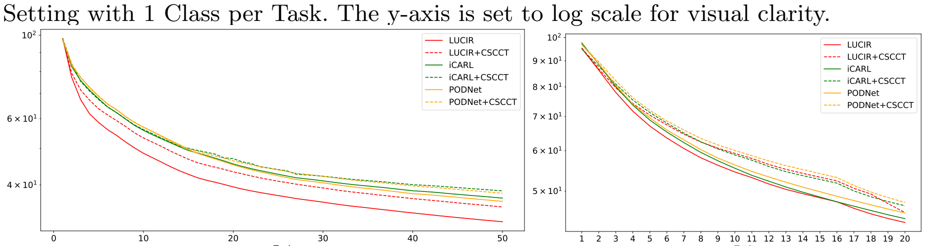

Fig. A2. Phase-wise average incremental accuracies on CIFAR100, on the 50 Task Setting with 2 classes per task (Left) and the 20 Task Setting with 5 classes per task (Right). The y-axis is set to log scale for visual clarity.

References

1. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: Podnet: Pooled outputs distillation for small-tasks incremental learning. In: ECCV (2020) 1

2. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., He, K.: Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677 (2017) 1
3. Hou, S., Pan, X., Loy, C.C., Wang, Z., Lin, D.: Learning a unified classifier incrementally via rebalancing. CVPR (2019) 1, 2
4. Smith, S.L., Kindermans, P.J., Le, Q.V.: Don't decay the learning rate, increase the batch size. In: ICLR (2018) 1
5. You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., Hsieh, C.J.: Large batch optimization for deep learning: Training bert in 76 minutes. arXiv preprint arXiv:1904.00962 (2019) 1