[Supplementary Material]
Class-Incremental Learning by Knowledge Distillation with Adaptive Feature Consolidation

Minsoo Kang[†] Jaeyoo Park[†] Bohyung Han[†,§]
ECE[†], ASRI[†], & IPAI[†,§]
Seoul National University
{kminsoo,bellos1203,bhhan}@snu.ac.kr

A. Appendix

This document first derives the upper bound of the expectation of the loss change, $\triangle\mathcal{L}(Z'_{\ell,c})$, in model updates, which is a detailed version of (7), (8), and (9) in the main paper. Secondly, we present the results from two different strategies to maintain exemplar sets. Thirdly, we compare with PODNet based on other metrics and using the random exemplar selection rule. Finally, we discuss the limitation.

A.1. Detailed Derivation of (7), (8), and (9)

By (5) of the main paper, the expected loss change given by model updates is given by:

$$\mathbb{E}\Big[\triangle\mathcal{L}(Z'_{\ell,c})\Big] = \mathbb{E}\Big[\langle\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y), Z'_{\ell,c} - Z_{\ell,c}\rangle_F\Big],$$

where the expectations are taken over $\mathcal{P}^{t-1}_{\text{data}}$. Let $A_{m,n}$ be the element at the m^{th} row and n^{th} column of the matrix A. Then, (8) in the main paper is given by the following derivation:

$$\begin{aligned}
\langle\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y), Z'_{\ell,c} - Z_{\ell,c}\rangle_F &= \text{tr}(\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y)^T(Z'_{\ell,c} - Z_{\ell,c})) \\
&= \sum_{i=1}^{W_\ell}(\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y)^T(Z'_{\ell,c} - Z_{\ell,c}))_{i,i} \\
&= \sum_{i=1}^{W_\ell}\sum_{j=1}^{H_\ell}(\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y)^T)_{i,j}(Z'_{\ell,c} - Z_{\ell,c})_{j,i} \\
&= \sum_{i=1}^{W_\ell}\sum_{j=1}^{H_\ell}(\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y))_{j,i}\cdot(Z'_{\ell,c} - Z_{\ell,c})_{j,i} \\
&\leq \sqrt{\Big(\sum_{i=1}^{W_\ell}\sum_{j=1}^{H_\ell}(\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y))^2_{j,i}\Big)} \times \sqrt{\Big(\sum_{i=1}^{W_\ell}\sum_{j=1}^{H_\ell}(Z'_{\ell,c} - Z_{\ell,c})^2_{j,i}\Big)} \\
&= \|\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell),y)\|_F\cdot\|Z'_{\ell,c} - Z_{\ell,c}\|_F,
\end{aligned} \tag{21}$$

On the other hand, for any given random variable X and Y, the following inequality holds for any real k:

$$\mathbb{E}[(kX+Y)^2] \geq 0 \iff \mathbb{E}[X^2]k^2 + 2k\mathbb{E}[XY] + \mathbb{E}[Y^2] \geq 0. \tag{22}$$

Because the number of all zeros for the above inequality is at most 1, the discriminant must be less than or equal to 0 as follows:

$$\mathbb{E}[XY]^2 - \mathbb{E}[X^2]\mathbb{E}[Y^2] \leq 0 \iff |\mathbb{E}[XY]| \leq \sqrt{\mathbb{E}[X^2]\mathbb{E}[Y^2]}. \tag{23}$$

Table 7. Comparisons between AFC and the state-of-the-art algorithms on CIFAR100 with two strategies for maintaining exemplar sets. One stores 2,000 images for the entire old classes in total, where 2,000 images are equally distributed across all the classes in the previous tasks while the other always keep 20 images for each of old classes, which are referred to as $R_{\text{total}} = 2,000$ and $R_{\text{per}} = 20$, respectively. Note that the results with $R_{\text{per}} = 20$ are copied from our main paper and methods with asterisks (*) denote our reproductions using the official code given by the authors. A bold-faced number represents the best performance in each column.

	CIFAR100			
	$R_{\text{total}} = 2000$		$R_{\text{per}} = 20$	
	50 stages	10 stages	50 stages	10 stages
New classes per stage	1	5	1	5
iCaRL [4]	42.34	56.52	44.20	53.78
BiC [5]	48.44	55.03	47.09	53.21
UCIR (NME) [2]	54.08	62.89	48.57	60.83
UCIR (CNN) [2]	55.20	63.62	49.30	61.22
GDumb* [3]	60.98	61.33	59.76	60.24
PODNet (NME) [1]	62.47	64.60	61.40	64.03
PODNet (CNN) [1]	61.87	64.68	57.98	63.19
PODNet (NME)* [1]	60.53	64.30	56.78	63.27
PODNet (CNN)* [1]	61.86	64.92	57.86	62.78
AFC (NME)	**63.88**	**65.42**	**62.58**	**64.29**
AFC (CNN)	**64.01**	**65.92**	**62.18**	**64.98**

Table 8. Performance comparison between AFC and PODNet on CIFAR100 based on backward transfer metric.

Backward Transfer (%)	CIFAR100			
	50 stages	25 stages	10 stages	5 stages
New classes per stage	1	2	5	10
PODNet (NME)* [1]	-20.05 ± 0.85	**-17.38 ± 1.11**	**-14.49 ± 0.78**	**-11.98 ± 0.78**
PODNet (CNN)* [1]	-29.49 ± 1.65	-27.06 ± 1.71	-25.60 ± 1.32	-23.45 ± 1.58
AFC (NME)	**-15.34 ± 0.54**	**-13.64 ± 0.97**	**-14.86 ± 0.43**	**-12.91 ± 1.90**
AFC (CNN)	**-18.52 ± 1.45**	-17.42 ± 0.77	-18.53 ± 0.80	-17.18 ± 1.19

By (23), we have the upper bound of the expectation of (21), which is given by

$$\begin{aligned} &\mathbb{E}\Big[\|\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell), y)\|_F \cdot \|Z'_{\ell,c} - Z_{\ell,c}\|_F\Big] \\ \leq &\sqrt{\mathbb{E}\Big[\|\nabla_{Z_{\ell,c}}\mathcal{L}(G_\ell(Z_\ell), y)\|_F^2\Big] \cdot \mathbb{E}\Big[\|Z'_{\ell,c} - Z_{\ell,c}\|_F^2\Big]}, \end{aligned} \tag{24}$$

and, we obtain (9) in the main paper.

A.2. Results from Two Different Strategies to Maintain Exemplar sets

In all experiments of the main paper, we store the same number of exemplars for each class in the previous tasks, *e.g*. 20 examples per class ($R_{\text{per}} = 20$). We also run the experiments with another strategy, where we allocate a fixed amount of memory for the entire old tasks, *e.g*. 2,000 examples in total ($R_{\text{total}} = 2000$). Table 7 illustrates the results on CIFAR-100 from the two strategies, which show the outstanding performance of the proposed approach, AFC, in both cases.

A.3. Results by Other Metrics

We report the backward transfer and average accuracy by comparing AFC with PODNet. Table 8 and 9 demonstrate that AFC also outperforms PODNet in terms of the metrics in most cases. Although AFC is marginally worse than PODNet (NME) on lower stage settings (10 and 5 stages) in terms of the backward transfer metric, it clearly outperforms PODNet for 50 and 25 stage settings, which suffer from more catastrophic forgetting.

Table 9. Performance comparison between AFC and PODNet on CIFAR100 based on average accuracy metric.

Average Accuracy (%)	CIFAR100			
	50 stages	25 stages	10 stages	5 stages
New classes per stage	1	2	5	10
PODNet (NME)* [1]	46.80 ± 1.21	49.63 ± 1.19	53.10 ± 0.78	55.77 ± 0.55
PODNet (CNN)* [1]	48.57 ± 0.25	51.30 ± 0.61	52.70 ± 0.36	54.80 ± 0.70
AFC (NME)	**52.30 ± 0.50**	**54.37 ± 0.64**	**54.70 ± 0.72**	**56.73 ± 1.07**
AFC (CNN)	**52.77 ± 0.21**	**54.50 ± 0.44**	**54.93 ± 0.51**	**56.87 ± 0.40**

Table 10. Performance comparison between AFC and PODNet on CIFAR100 using the random selection for exemplars.

Random Exemplar Selection Rule	CIFAR100			
	50 stages	25 stages	10 stages	5 stages
New classes per stage	1	2	5	10
PODNet (NME)* [1]	54.99 ± 0.84	58.11 ± 0.79	61.75 ± 1.04	63.67 ± 1.02
PODNet (CNN)* [1]	55.55 ± 1.83	58.18 ± 1.56	61.12 ± 1.39	63.35 ± 1.01
AFC (NME)	**60.37 ± 0.83**	**62.12 ± 0.68**	**62.65 ± 0.95**	**64.31 ± 0.55**
AFC (CNN)	**59.51 ± 1.04**	**61.55 ± 0.96**	**62.90 ± 1.15**	**64.53 ± 1.10**

A.4. Results by Random Exemplar Selection Rule

Table 10 shows that the nearest-mean of exemplars rule is more effective than the random selection rule for both AFC and PODNet compared with the result of Table 1 in the main paper. Note that AFC also outperforms PODNet combined with the random selection for the exemplars.

A.5. Limitation

Existing methods for class incremental learning essentially require the exemplar sets for the old tasks in order to reduce the catastrophic forgetting problem. Storing the exemplar sets such as the personal medical datasets potentially poses risk in privacy. Moreover, the proposed method and other functional regularization approaches require additional forward passes for the old models, which leads to extra computational cost in terms of FLOPs, memory, and power consumptions. Therefore, it would be important to develop algorithms applicable to the resource-hungry systems.

References

[1] Arthur Douillard, Matthieu Cord, Charles Ollion, and Thomas Robert. PODNet: Pooled Outputs Distillation for Small-Tasks Incremental Learning. In *ECCV*, 2020. 2, 3

[2] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a Unified Classifier Incrementally via Rebalancing. In *CVPR*, 2019. 2

[3] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. GDumb: A Simple Approach that Questions Our Progress in Continual Learning. In *ECCV*, 2020. 2

[4] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental Classifier and Representation Learning. In *CVPR*, 2017. 2

[5] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large Scale Incremental Learning. In *CVPR*, 2019. 2