Adaptive Aggregation Networks for Class-Incremental Learning
Supplementary Materials

Yaoyao Liu[1] Bernt Schiele[1] Qianru Sun[2]

[1]Max Planck Institute for Informatics, Saarland Informatics Campus
[2]School of Computing and Information Systems, Singapore Management University
{yaoyao.liu, schiele}@mpi-inf.mpg.de qianrusun@smu.edu.sg

These supplementary materials include the results for different CIL settings(§A), "strict memory budget" experiments (§B), additional ablation results (§C), additional plots (§D), more visualization results (§E), and the execution steps of our source code with PyTorch (§F).

A. Results for Different CIL Settings.

We provide more results on the setting with the same number of classes at all phases [32] in the second block (**"same # of cls"**) of Table S1. For example, N=25 indicates 100 classes evenly come in 25 phases, so 4 new classes arrive in each phase (including the 0-th phase). Further, in this table, each entry represents an accuracy of the **last phase** (since all-phase accuracies are not comparable to our original setting) averaged over 3 runs; and "update θ_{base}" means that θ_{base} is updated as $\theta_{\text{base}} \leftarrow \phi_i \odot \theta_{\text{base}}$ after each phase. All results are under "strict memory budget" and "all"+"scaling" settings, so ϕ_i indicate the meta-learned weights of SS operators. The results show that 1) "*w/* AANets" performs best in all settings and brings consistent improvements; and 2) "update θ_{base}" is helpful for CIFAR-100 but harmful for ImageNet-Subset.

Last-phase acc. (%)	*CIFAR-100*			*ImageNet-Subset*		
	N=5	10	25	5	10	25
LUCIR (50 cls in Phase 0)	54.3	50.3	48.4	60.0	57.1	49.3
w/ AANets	**58.6**	**56.7**	**53.3**	64.3	58.0	56.5
LUCIR (**same # of cls**)	52.1	44.9	40.6	60.3	52.5	53.3
w/ AANets, update θ_{base}	54.3	47.4	42.4	61.4	52.5	48.2
w/ AANets	52.6	46.1	41.9	**68.8**	**60.8**	**56.8**

Table S1. Supplementary to Table 1. Last-phase accuracies (%) for different class-incremental learning (CIL) settings.

B. Strict Memory Budget Experiments

In Table S2, we present the results of 4 state-of-the-art methods *w/* and *w/o* AANetss as a plug-in architecture, under the "strict memory budget" setting which strictly controls the total memory shared by the exemplars and the model parameters. For example, if we incorporate AANetss to LUCIR [16], we need to reduce the number of exemplars to balance the additional memory introduced by AANetss (as AANetss take around 20% more parameters than the plain ResNets used in LUCIR [16]). As a result, we reduce the numbers of exemplars for AANetss from 20 to 13, 16 and 19, respectively, for CIFAR-100, ImageNet-Subset, and ImageNet, in the "strict memory budget" setting. For CIFAR-100, we use $530k$ additional parameters, so we need to reduce $530k\text{floats} \times 4\text{bytes/float} \div (32 \times 32 \times 3\text{bytes/image}) \div 100\text{classes} \approx 6.9\text{images/class}$, and $\lceil 6.9 \rceil = 7\text{images/class}$. For ImageNet-Subset, we use $12.6M$ additional parameters, so we need to reduce $12.6M\text{floats} \times 4\text{bytes/float} \div (224 \times 224 \times 3\text{bytes/image}) \div 100\text{classes} \approx 3.3\text{images/class}$, and $\lceil 3.3 \rceil = 4\text{images/class}$. For ImageNet, we use $12.6M$ additional parameters, so we need to reduce $12.6M\text{floats} \times 4\text{bytes/float} \div (224 \times 224 \times 3\text{bytes/image}) \div 100\text{classes} \approx 0.3\text{images/class}$, and $\lceil 0.3 \rceil = 1\text{image/class}$. From Table S2, we can see that our approach of using AANetss still achieves the top performances in all CIL settings even if the "strict memory budget" is applied.

C. More Ablation Results

In Table S3, we supplement the ablation results obtained in more settings. "4×" denotes that we use 4 same-type blocks at each residual level. Comparing Row 7 to Row 2 (Row 5) shows the efficiency of using different types of blocks for representing stability and plasticity.

D. Additional Plots

In Figures S2, we present the phase-wise accuracies obtained on CIFAR-100, ImageNet-Subset and ImageNet, respectively. "Upper Bound" shows the results of joint training with all previous data accessible in every phase. We can observe that our method achieves the highest accuracies in almost every phase of different settings. In Figures S3 and S4, we supplement the plots for the values of α_η and α_ϕ learned on the CIFAR-100 and ImageNet-Subset (N=5, 25). All curves are smoothed with a rate of 0.8 for a better visualization.

E. More Visualization Results

Figure S1 below shows the activation maps of a "goldfinch" sample (seen in Phase 0) in different-phase models (ImageNet-Subset, N=5). Notice that the plastic block gradually loses its attention on this sample (i.e., forgets it), while the stable block retains it. AANets benefit from its stable blocks.

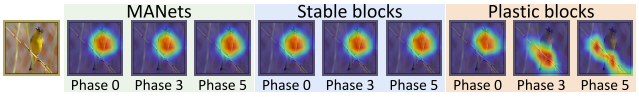


Figure S1. Supplementary to Figure 3. The activation maps of a "goldfinch" sample (seen in Phase 0) in different-phase models (ImageNet-Subset; N=5).

F. Source Code in PyTorch

We provide our PyTorch code on https://class-il.mpi-inf.mpg.de/. To run this repository, we kindly advise you to install Python 3.6 and PyTorch 1.2.0 with Anaconda. Create a new environment and install PyTorch on it:

```
conda create --name e3bm-pytorch python=3.6
conda activate e3bm-pytorch
conda install pytorch=1.2.0
conda install torchvision -c pytorch
```

Install other requirements:

```
pip install tqdm tensorboardX Pillow==6.2.2
```

Running experiments on CIFAR-100:

```
python3 main.py --nb_cl_fg=50 --nb_cl=10 --gpu=0
    --random_seed=1993 --baseline=lucir --
    branch_mode=dual --branch_1=ss --branch_2=
    free --dataset=cifar100
```

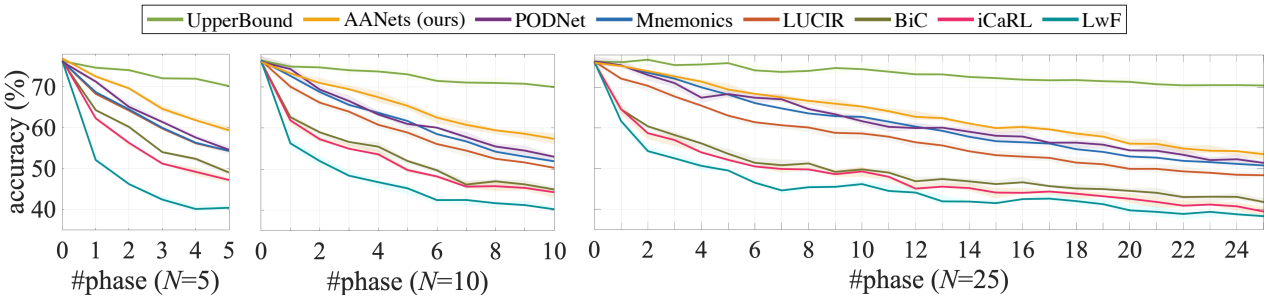
```
python3 main.py --nb_cl_fg=50 --nb_cl=10 --gpu=0
    --random_seed=1993 --baseline=icarl --
    branch_mode=dual --branch_1=ss --branch_2=
    free --dataset=cifar100
```

Method	*CIFAR-100*			*ImageNet-Subset*			*ImageNet*		
	N=5	10	25	5	10	25	5	10	25
iCaRL [34]	57.12$_{\pm 0.50}$	52.66$_{\pm 0.89}$	48.22$_{\pm 0.76}$	65.44$_{\pm 0.35}$	59.88$_{\pm 0.83}$	52.97$_{\pm 1.02}$	51.50$_{\pm 0.43}$	46.89$_{\pm 0.35}$	43.14$_{\pm 0.67}$
w/ AANetss (ours)	63.91$_{\pm 0.52}$	57.65$_{\pm 0.81}$	52.10$_{\pm 0.87}$	71.37$_{\pm 0.57}$	66.34$_{\pm 0.61}$	61.87$_{\pm 1.01}$	63.65$_{\pm 1.02}$	61.14$_{\pm 0.59}$	55.91$_{\pm 0.95}$
	64.22$_{\pm 0.42}$	60.26$_{\pm 0.73}$	56.43$_{\pm 0.81}$	73.45$_{\pm 0.51}$	71.78$_{\pm 0.64}$	69.22$_{\pm 0.83}$	63.91$_{\pm 0.59}$	61.28$_{\pm 0.49}$	56.97$_{\pm 0.86}$
LUCIR [16]	63.17$_{\pm 0.87}$	60.14$_{\pm 0.73}$	57.54$_{\pm 0.43}$	70.84$_{\pm 0.69}$	68.32$_{\pm 0.81}$	61.44$_{\pm 0.91}$	64.45$_{\pm 0.32}$	61.57$_{\pm 0.23}$	56.56$_{\pm 0.36}$
w/ AANetss (ours)	66.46$_{\pm 0.45}$	65.38$_{\pm 0.53}$	61.79$_{\pm 0.51}$	72.21$_{\pm 0.87}$	69.10$_{\pm 0.90}$	67.10$_{\pm 0.54}$	64.83$_{\pm 0.50}$	62.34$_{\pm 0.65}$	60.49$_{\pm 0.78}$
	66.74$_{\pm 0.37}$	65.29$_{\pm 0.43}$	63.50$_{\pm 0.61}$	72.55$_{\pm 0.67}$	69.22$_{\pm 0.72}$	67.60$_{\pm 0.39}$	64.94$_{\pm 0.25}$	62.39$_{\pm 0.61}$	60.68$_{\pm 0.58}$
Mnemonics [25]	63.34$_{\pm 0.62}$	62.28$_{\pm 0.43}$	60.96$_{\pm 0.72}$	72.58$_{\pm 0.85}$	71.37$_{\pm 0.56}$	69.74$_{\pm 0.39}$	64.54$_{\pm 0.49}$	63.01$_{\pm 0.57}$	61.00$_{\pm 0.71}$
w/ AANetss (ours)	66.12$_{\pm 0.00}$	65.10$_{\pm 0.00}$	61.83$_{\pm 0.00}$	72.88$_{\pm 0.00}$	71.50$_{\pm 0.00}$	70.49$_{\pm 0.00}$	65.21$_{\pm 0.76}$	63.36$_{\pm 0.67}$	61.37$_{\pm 0.80}$
	67.59$_{\pm 0.34}$	65.66$_{\pm 0.61}$	63.35$_{\pm 0.72}$	72.91$_{\pm 0.53}$	71.93$_{\pm 0.37}$	70.70$_{\pm 0.45}$	65.23$_{\pm 0.62}$	63.60$_{\pm 0.71}$	61.53$_{\pm 0.29}$
PODNet-CNN [11]	64.83$_{\pm 1.11}$	63.19$_{\pm 1.31}$	60.72$_{\pm 1.54}$	75.54$_{\pm 0.29}$	74.33$_{\pm 1.05}$	68.31$_{\pm 2.77}$	66.95	64.13	59.17
w/ AANetss (ours)	66.36$_{\pm 1.02}$	64.31$_{\pm 1.13}$	61.80$_{\pm 1.24}$	76.63$_{\pm 0.35}$	75.00$_{\pm 0.78}$	71.43$_{\pm 1.51}$	67.80$_{\pm 0.87}$	64.80$_{\pm 0.60}$	61.01$_{\pm 0.97}$
	66.31$_{\pm 0.87}$	64.31$_{\pm 0.90}$	62.31$_{\pm 1.02}$	76.96$_{\pm 0.53}$	75.58$_{\pm 0.74}$	71.78$_{\pm 0.81}$	67.73$_{\pm 0.71}$	64.85$_{\pm 0.53}$	61.78$_{\pm 0.61}$

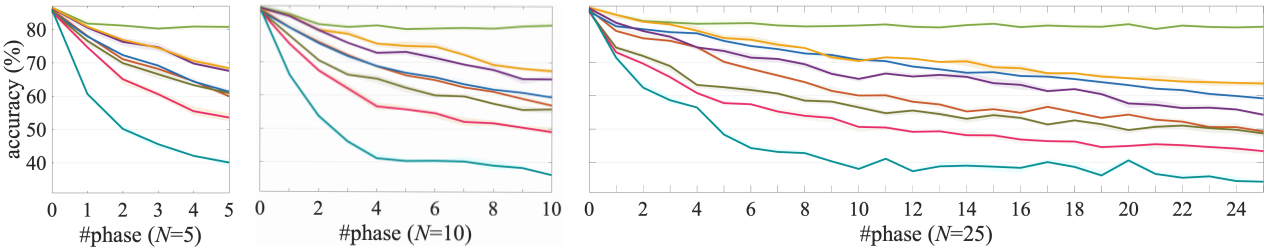
Table S2. Supplementary to Table 2. **Using "strict memory budget" setting.** Average incremental accuracies (%) of four state-of-the-art methods *w/* and *w/o* our AANetss as a plug-in architecture. The red lines are the corresponding results in Table 2 of the main paper.

Row	Ablation Setting	*CIFAR-100* (acc.%)						*ImageNet-Subset* (acc.%)					
		Memory	FLOPs	#Param	N=5	10	25	Memory	FLOPs	#Param	N=5	10	25
1	single-branch "all" [16]	7.64MB	70M	469K	63.17	60.14	57.54	330MB	1.82G	11.2M	70.84	68.32	61.44
2	"all" + "all"	9.43MB	140M	938K	64.49	61.89	58.87	372MB	3.64G	22.4M	69.72	66.69	63.29
3	4× "all"	13.01MB	280M	1.9M	65.13	64.08	59.40	456MB	7.28G	44.8M	70.12	67.31	64.00
4	single-branch "scaling"	7.64MB	70M	60K	62.48	61.53	60.17	334MB	1.82G	1.4M	71.29	68.88	66.75
5	"scaling" + "scaling"	9.43MB	140M	120K	65.13	64.08	62.50	382MB	3.64G	2.8M	71.71	71.07	66.69
6	4× "scaling"	13.01MB	240M	280K	66.00	64.67	63.16	478MB	3.64G	5.6M	72.01	71.23	67.12
7	"all" + "scaling"	9.66MB	140M	530K	**66.74**	**65.29**	**63.50**	378MB	3.64G	12.6M	72.55	69.22	67.60
8	"all" + "frozen"	9.43MB	140M	469K	65.62	64.05	63.67	372MB	3.64G	11.2M	71.71	69.87	67.92
9	"scaling" + "frozen"	9.66MB	140M	60K	64.71	63.65	62.89	378MB	3.64G	1.4M	**73.01**	**71.65**	**70.30**

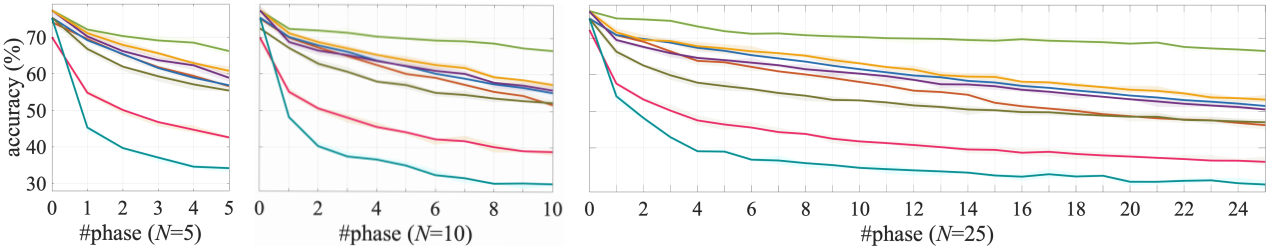
Table S3. Supplementary to Table 1. More ablation study. "4×" denotes that we use 4 same-type blocks at each residual level.



(a) CIFAR-100 (100 classes). In the 0-th phase, θ_{base} is trained on 50 classes, the remaining classes are given evenly in the subsequent phases.

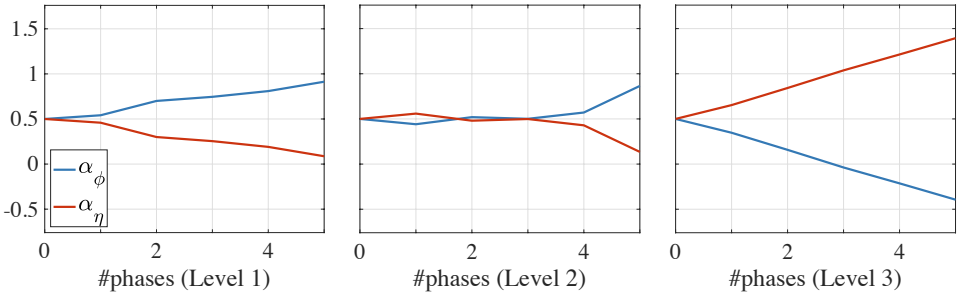


(b) ImageNet-Subset (100 classes). In the 0-th phase, θ_{base} is trained on 50 classes, the remaining classes are given evenly in the subsequent phases.

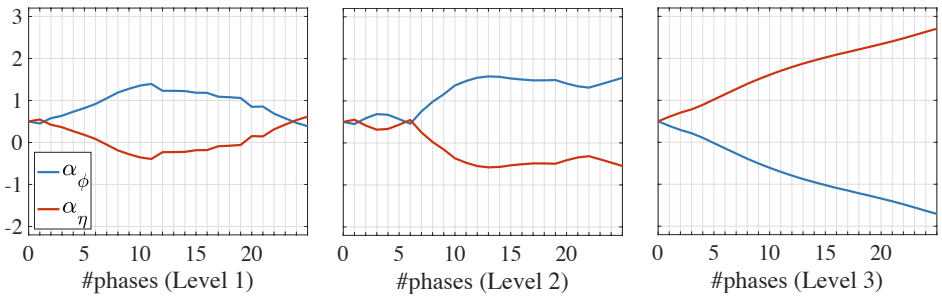


(c) ImageNet (1000 classes). In the 0-th phase, θ_{base} on is trained on 500 classes, the remaining classes are given evenly in the subsequent phases.

Figure S2. Supplementary to Table 2.Phase-wise accuracies (%). Light-color ribbons are visualized to show the 95% confidence intervals. Comparing methods: Upper Bound (the results of joint training with all previous data accessible in each phase); PODNet (2020) [11]; Mnemonics (2020) [25]; LUCIR (2019) [16]; BiC (2019) [48]; iCaRL (2017) [34]; and LwF (2016) [23].

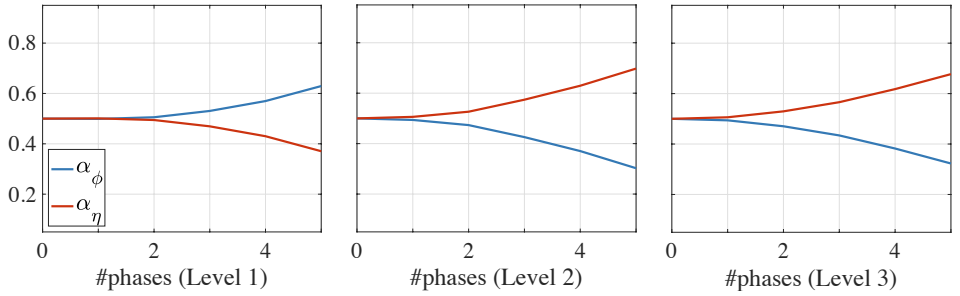


(a) CIFAR-100, N=5

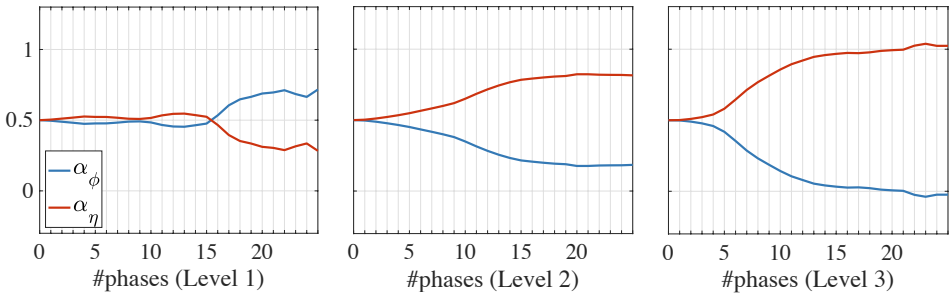


(b) CIFAR-100, N=25

Figure S3. Supplementary to Figure 4. The changes of values for α_η and α_ϕ on CIFAR-100.



(a) ImageNet-Subset, N=5



(b) ImageNet-Subset, N=25

Figure S4. Supplementary to Figure 4. The changes of values for α_η and α_ϕ on ImageNet-Subset.

References

[1] Davide Abati, Jakub Tomczak, Tijmen Blankevoort, Simone Calderara, Rita Cucchiara, and Babak Ehteshami Bejnordi. Conditional channel gated networks for task-aware continual learning. In *CVPR*, pages 3931–3940, 2020. 9

[2] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert gate: Lifelong learning with a network of experts. In *CVPR*, pages 3366–3375, 2017. 9

[3] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *CVPR*, pages 583–592, 2019. 9

[4] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *ECCV*, pages 241–257, 2018. 8, 9

[5] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, pages 532–547, 2018. 9

[6] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *ICLR*, 2019. 9

[7] Zhiyuan Chen and Bing Liu. Lifelong machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(3):1–207, 2018. 9

[8] Guy Davidson and Michael C Mozer. Sequential mastery of multiple visual tasks: Networks naturally learn to learn and forget to forget. In *CVPR*, pages 9282–9293, 2020. 9

[9] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. Continual learning: A comparative study on how to defy forgetting in classification tasks. *arXiv*, 1909.08383, 2019. 9

[10] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *arXiv*, 1909.08383, 2019. 9

[11] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *ECCV*, 2020. 3, 4, 8, 9, 10, 11, 12, 13, 14

[12] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017. 8

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014. 10, 11

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 8, 10, 12

[15] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *arXiv*, 1503.02531, 2015. 9

[16] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *CVPR*, pages 831–839, 2019. 1, 3, 4, 8, 9, 10, 11, 12, 13, 14

[17] Wenpeng Hu, Zhou Lin, Bing Liu, Chongyang Tao, Zhengwei Tao, Jinwen Ma, Dongyan Zhao, and Rui Yan. Overcoming catastrophic forgetting for continual learning via model adaptation. In *ICLR*, 2019. 9

[18] Xinting Hu, Kaihua Tang, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. Distilling causal effect of data in class-incremental learning. In *CVPR*, 2021. 9, 10

[19] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler L. Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. In *AAAI*, pages 3390–3398, 2018. 8

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 12

[21] Xinzhe Li, Qianru Sun, Yaoyao Liu, Qin Zhou, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. In *NeurIPS*, pages 10276–10286, 2019. 10

[22] Yingying Li, Xin Chen, and Na Li. Online optimal control with linear dynamics and predictions: Algorithms and regret analysis. In *NeurIPS*, pages 14858–14870, 2019. 9

[23] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. 4, 8, 9, 14

[24] Yaoyao Liu, Bernt Schiele, and Qianru Sun. An ensemble of epoch-wise empirical bayes for few-shot learning. In *ECCV*, pages 404–421, 2020. 10

[25] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *CVPR*, pages 12245–12254, 2020. 3, 4, 8, 9, 10, 11, 12, 13, 14

[26] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NIPS*, pages 6467–6476, 2017. 9

[27] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pages 109–165. Elsevier, 1989. 8

[28] K. McRae and P. Hetherington. Catastrophic interference is eliminated in pre-trained networks. In *CogSci*, 1993. 8

[29] Martial Mermillod, Aurélia Bugaiska, and Patrick Bonin. The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects. *Frontiers in Psychology*, 4:504, 2013. 8

[30] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *ECCV*, 2020. 9

[31] Jathushan Rajasegaran, Munawar Hayat, Salman H Khan, Fahad Shahbaz Khan, and Ling Shao. Random path selection for continual learning. In *NeurIPS*, pages 12669–12679, 2019. 9

[32] Jathushan Rajasegaran, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Mubarak Shah. itaml: An incremental task-agnostic meta-learning approach. In *CVPR*, pages 13588–13597, 2020. 1, 10

[33] R. Ratcliff. Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological Review*, 97:285–308, 1990. 8

[34] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In *CVPR*, pages 5533–5542, 2017. 3, 4, 8, 9, 11, 12, 13, 14

[35] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *ICLR*, 2019. 9

[36] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *ICLR*, 2019. 10

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 8, 12

[38] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv*, 1606.04671, 2016. 9

[39] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *CVPR*, pages 618–626, 2017. 13, 15

[40] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *NIPS*, pages 2990–2999, 2017. 8, 9

[41] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2018. 8

[42] Qianru Sun, Yaoyao Liu, Zhaozheng Chen, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning through hard tasks. *arXiv*, 1910.03648, 2019. 12

[43] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *CVPR*, pages 403–412, 2019. 10, 12, 13

[44] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. Topology-preserving class-incremental learning. In *ECCV*, 2020. 9, 14

[45] Heinrich Von Stackelberg and Stackelberg Heinrich Von. *The theory of the market economy*. Oxford University Press, 1952. 10

[46] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A. Efros. Dataset distillation. *arXiv*, 1811.10959, 2018. 10

[47] Max Welling. Herding dynamical weights to learn. In *ICML*, pages 1121–1128, 2009. 9

[48] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *CVPR*, pages 374–382, 2019. 4, 8, 9, 10, 14

[49] Ju Xu and Zhanxing Zhu. Reinforced continual learning. In *NeurIPS*, pages 899–908, 2018. 9

[50] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NIPS*, pages 3320–3328, 2014. 11, 14

[51] Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *CVPR*, pages 6982–6991, 2020. 9

[52] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, pages 12203–12213, 2020. 10

[53] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *CVPR*, 2021. 9

[54] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *CVPR*, pages 13208–13217, 2020. 9