Supplementary Materials for
DyRep: Bootstrapping Training with Dynamic Re-parameterization

A. More Ablation Studies

A.1. Performance on different operation spaces

Compared to the DBB [2] using 4 branches to build the augmented network, DyRep further adds 3 operations for more flexible structures. We now conduct experiments to compare our DyRep with DBB using the same operation spaces for fair comparisons. As summarized in Table 8, we train ResNet-50 using DyRep and DBB with 4 operations and 7 operations on ImageNet and report their validation accuracies. When using the same operation space as DBB, our DyRep can also enjoy significant efficiency and performance improvements. Besides, if we adopt DBB on our larger operation space, its training cost will be higher, and our superiority will be more significant. Comparing 4 branches and 7 branches, the larger one achieves better accuracy because of more diverse representations.

Table 8. Evaluation results of ResNet-50 on ImageNet dataset using different numbers of Rep operations. *: our implementation.

Rep method	#operations	FLOPs	Params	Training cost	ACC (%)
Origin	1	4.09	25.6	7.5	76.14
DBB	4	6.79	40.7	13.7	76.71
DyRep	4	**4.93** (-27.4%)	**30.2** (-25.8%)	**8.1** (-40.9%)	**76.98** (+0.27%)
DBB*	7	8.02	48.3	17.3	76.87
DyRep	7	**5.05** (-37.0%)	**31.5** (-34.8%)	**8.5** (-50.9%)	**77.08** (+0.21%)

A.2. Effects of different update intervals t of DyRep

We update the structures of the network in every t epoch. If the structures are expanded more frequently, the final network will be larger. As summarized in Table 9, we train the models with different update intervals t, and the results show that for a small t, the accuracy will be further improved, but the training cost also increases accordingly. For efficiency consideration, we choose a moderate frequency $t = 15$ on CIFAR-10.

Table 9. Evaluation results of VGG-16 on CIFAR-10 with different update interval t.

Rep method	Update interval t	Cost (GPU hours)	FLOPs (M)	Params (M)	ACC (%)
origin	-	2.4	313	15.0	94.68
DBB	-	9.4	728	34.7	94.97
DyRep	5	13.3	1575	83.4	95.39
DyRep	10	8.8	992	33.6	95.33
DyRep	15	6.9	597	26.4	95.22
DyRep	30	5.8	522	23.7	94.91
DyRep	50	4.1	430	20.3	94.82

A.3. Effects of different scoring metrics in our Rep

Many metrics [14, 24, 25] are proposed to measure the saliency score of weights in network pruning. In our paper, to choose the most suitable metric, we conduct experiments to evaluate these scoring metrics in DyRep. The experimented scoring metrics are summarized as follows.

For one operation with weights $\boldsymbol{\theta}$, its saliency score can be represented as following metrics:

- *random*: the score of each operation is generated randomly.

$$\mathcal{S}_o(\boldsymbol{\theta}) \overset{\text{iid}}{\sim} \mathbb{R}^1. \tag{11}$$

- *grad_norm*: A simple baseline of summing the Euclidean norm of the gradients.

$$\mathcal{S}_o(\boldsymbol{\theta}) = \left\| \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right\|_2. \tag{12}$$

- *snip* [14]:

$$\mathcal{S}_o(\boldsymbol{\theta}) = \sum_i^n \left| \frac{\partial \mathcal{L}}{\partial \theta_i} \odot \theta_i \right|. \tag{13}$$

- *grasp*: [25] aims to improve *snip* by approximating the change in gradient norm (instead of loss), therefore its *grasp* metric is computed as

$$\mathcal{S}_o(\boldsymbol{\theta}) = \sum_i^n -(H \frac{\partial \mathcal{L}}{\partial \theta_i} \odot \theta_i), \tag{14}$$

  where H denotes the Hessian matrix.

- *synflow*: SynFlow [24] proposes a modified version (*synflow*) to avoid layer collapse when performing parameter pruning.

$$\mathcal{S}_o(\boldsymbol{\theta}) = \sum_i^n \frac{\partial \mathcal{L}}{\partial \theta_i} \odot \theta_i. \tag{15}$$

- *vote*: Inspired by [**?**], which leverages the above metrics to vote the decisions, we also provide a result of picking operations with the most votes.

We measure all the metrics above on CIFAR-10 dataset, as summarized in Table 10. The random baseline even worsens the performance as it could introduce some unnecessary disturbances to the original weights, showing that it is important in choosing operations. Besides, *synflow* achieves the best performance compared to other metrics, and we thus adopt it as the scoring metric in our DyRep.

Table 10. Accuracies of VGG-16 using different scoring metrics in DyRep. *Origin* denotes the original results of model without Rep.

Dataset	origin	random	grad_norm	snip	grasp	synflow	vote
CIFAR-10	94.68	94.25	94.71	94.97	94.82	**95.22**	95.03
CIFAR-100	74.10	74.03	74.19	74.56	74.73	**74.91**	74.79

A.4. Effects of training with DyRep for different epochs

To validate the effectiveness of DyRep in boosting training, we adopt DyRep for the first 20,40,60,80, and 100 epochs, then train the remained epochs with fixed structures. As illustrated in Figure 7, our DyRep can dynamically adapt the structures thus is more steady compared to training with fixed structures. Besides, adopting Rep can always obtain higher accuracy than fixed structures, showing that DyRep can improve performance in the whole training process.

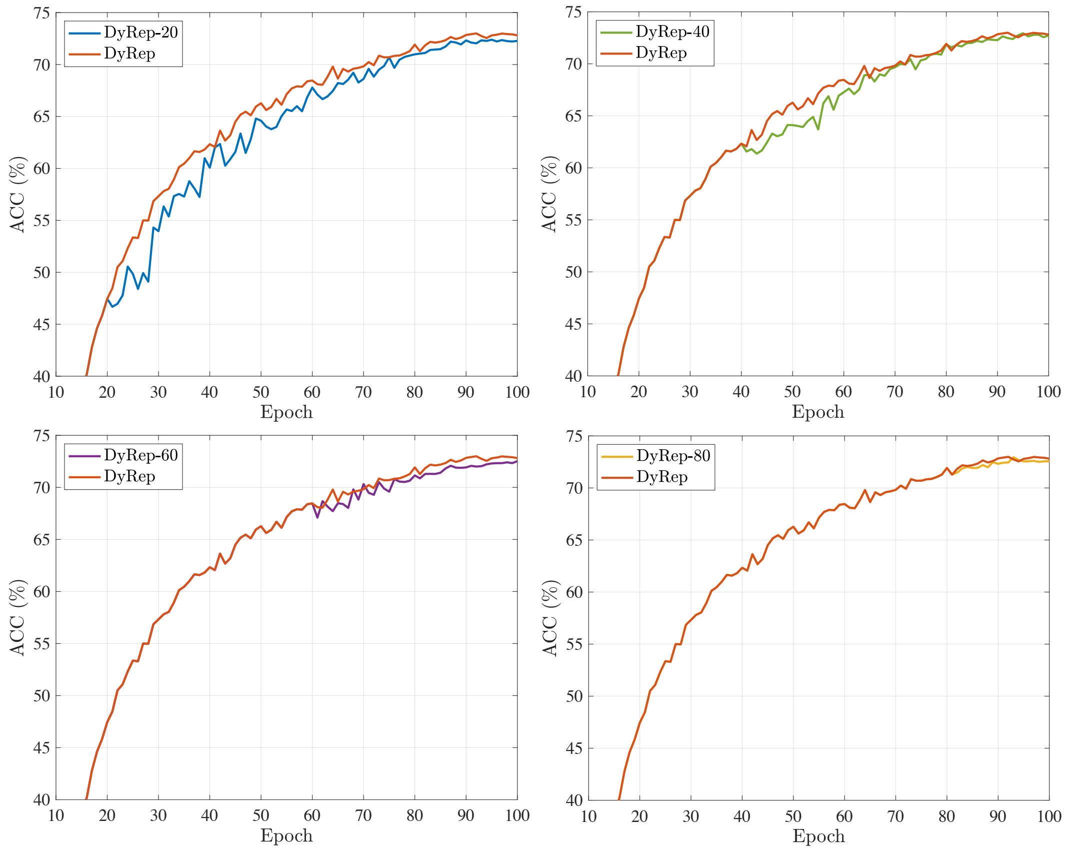

Figure 7. Training curves of adopting DyRep for different epochs on CIFAR-100. DyRep-N denotes training with DyRep for the first N epochs then fixing the structure for the latter $100 - N$ epochs. DyRep with red line means adopting DyRep in the whole training.

References

[1] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.

[2] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Diverse branch block: Building a convolution as an inception-like unit. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10886–10895, 2021. 1

[3] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021.

[4] Chengyu Dong, Liyuan Liu, Zichao Li, and Jingbo Shang. Towards adaptive residual network training: A neural-ode perspective. In *International conference on machine learning*, pages 2616–2626. PMLR, 2020.

[5] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6910–6919, 2021.

[6] Ariel Gordon, Elad Eban, Ofir Nachum, Bo Chen, Hao Wu, Tien-Ju Yang, and Edward Choi. Morphnet: Fast & simple resource-constrained structure learning of deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1586–1595, 2018.

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[10] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[11] Zehao Huang and Naiyan Wang. Data-driven sparse structure selection for deep neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 304–320, 2018.

[12] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[13] Roxana Istrate, Adelmo Cristiano Innocenza Malossi, Costas Bekas, and Dimitrios Nikolopoulos. Incremental training of deep convolutional neural networks. *arXiv preprint arXiv:1803.10232*, 2018.

[14] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. Snip: Single-shot network pruning based on connection sensitivity. In *International Conference on Learning Representations*, 2018. 1, 2

[15] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 510–519, 2019.

[16] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[17] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. In *International Conference on Learning Representations*, 2018.

[18] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Proceedings of the IEEE international conference on computer vision*, pages 2736–2744, 2017.

[19] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[21] Xiu Su, Tao Huang, Yanxi Li, Shan You, Fei Wang, Chen Qian, Changshui Zhang, and Chang Xu. Prioritized architecture sampling with monto-carlo tree search. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10963–10972. IEEE Computer Society, 2021.

[22] Xiu Su, Shan You, Tao Huang, Fei Wang, Chen Qian, Changshui Zhang, and Chang Xu. Locally free weight sharing for network width search. In *International Conference on Learning Representations*, 2020.

[23] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[24] Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2

[25] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. In *International Conference on Learning Representations*, 2019. 1, 2

[26] Tao Wei, Changhu Wang, and Chang Wen Chen. Modularized morphing of neural networks. *arXiv preprint arXiv:1701.03281*, 2017.

[27] Tao Wei, Changhu Wang, Yong Rui, and Chang Wen Chen. Network morphism. In *International Conference on Machine Learning*, pages 564–572. PMLR, 2016.

[28] Wei Wen, Feng Yan, Yiran Chen, and Hai Li. Autogrow: Automatic layer growing in deep convolutional networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 833–841, 2020.

[29] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[30] Jianbo Ye, Xin Lu, Zhe Lin, and James Z Wang. Rethinking the smaller-norm-less-informative assumption in channel pruning of convolution layers. In *International Conference on Learning Representations*, 2018.

[31] Shan You, Tao Huang, Mingmin Yang, Fei Wang, Chen Qian, and Changshui Zhang. Greedynas: Towards fast one-shot nas with greedy supernet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1999–2008, 2020.

[32] Mingyang Zhang, Xinyi Yu, Jingtao Rong, Linlin Ou, and Feng Gao. Repnas: Searching for efficient re-parameterizing blocks. *arXiv preprint arXiv:2109.03508*, 2021.

[33] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[34] Mingkai Zheng, Fei Wang, Shan You, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. Weakly supervised contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10042–10051, 2021.

[35] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.