Supplementary Materials for *Shadow Knowledge Distillation: Bridging Offline and Online Knowledge Transfer*

Lujun Li[1,2,✉], **Jin Zhe**[1,✉]
School of Artificial Intelligence, Anhui University, China
Chinese Academy of Science, China
lilujunai@gmail.com; jinzhe@ahu.edu.cn

1 Main Experimental Settings

In this section, we provide detailed settings of the classification experiments and extended experiments.

1.1 Experiments on CIFAR

Dataset. CIFAR [3] is the most widely used classification dataset for evaluating the performance of distillation methods. It includes 50,000 training and 10,000 test images.

Implementation. In the comparison experiments with other offline KD methods, we use the same training settings of CRD [11] to implement various KD methods [5, 6, 4, 7, 12], whose training epochs are 240. We used a 32×32 random crop after padding with 4 pixels and a random horizontal flip, and we optimized the models with the stochastic gradient descent (SGD) algorithm with a learning rate of 0.05 and applied learning rate decay in 150, 180, and 210 epochs, for a total of 240 epochs. As a student model, the initial learning rate in ShuffleNetV1 and ShuffleNetV2 is set to 0.01. We used a 5e-4 weight decay, a momentum of 0.9, and a batch size of 64.

1.2 Experiments on ImageNet

Dataset. We also perform experiments on the ImageNet dataset (ILSVRC12) [10], which is regarded as the most difficult classification task. It has approximately 1.2 million training images and 50,000 validation images, with each image belonging to one of 1,000 categories.

Implementation. In the ImageNet experiments, the student models (*i.e.*, ResNet-18 [1] and MobileNet [2]) are trained with 100 epochs. For the data augmentation, we employ the standard data augmentation technique, which includes random cropping, random horizontal flipping, and brightness adjustment. We used the SGD algorithm for the optimizer, with a Nesterov momentum of 0.9, weight decay of 0.0001, and an initial learning rate of 0.1. Other KD methods are implemented using the hyperparameter settings of original paper. The SHAKE's detailed settings are same to the CIFAR-100.

1.3 Experiments on vision transformer

Implementation. We utilize the same data augmentation and regularization methods described in DeiT for fair comparison (*e.g.*, Auto-Augment, Rand-Augment, mixup). We use AdamW as the optimizer, with a learning rate of 0.001 and a weight decay of 0.05. The entire training procedure consists of 300 epochs. The first five epochs are for warm-up, and the learning rate follows a cosine decay function in the remaining epochs. SHAKE, like DeiT, uses the distillation token with the shadow head as the proxy model. Furthermore, SHAKE incorporates mutual distillation between the shadow head and the classification head, yielding significantly higher gains than DeiT.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

1.4 Experiments on object detection.

Dataset. We evaluate SHAKE on MS-COCO dataset [8] , which contains more than 120K images, covering 80 categories. All performance is evaluated on the MS-COCO validation set.

Implementation. We apply SHAKE Faster R-CNN [9]) and initialize the backbone with weights pre-trained on ImageNet [10]. Horizontal image flipping is utilized in data augmentation. For SHAKE, we build an extra shadow head with the same architecture as the original classification head, which performs distillation in the fine-tuning detector stage. Following most of the output logits distillation on the detection, our SHAKE conducts distillation on the classification and regression output predictions. For classification, we exploit the KL divergence for multi-label distillation. For regression, our SHAKE minimizes the bounding box distance between teacher-student. The two distillation loss terms are combined with the summation.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[2] 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.

[3] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Tech Report*, 2009.

[4] Lujun Li. Self-regulated feature learning via teacher-free feature distillation. In *European Conference on Computer Vision (ECCV)*, 2022.

[5] Lujun Li, Liang Shiuan-Ni, Ya Yang, and Zhe Jin. Boosting online feature transfer via separable feature fusion. In *IJCNN*, 2022.

[6] Lujun Li, Liang Shiuan-Ni, Ya Yang, and Zhe Jin. Teacher-free distillation via regularizing intermediate representation. In *IJCNN*, 2022.

[7] Lujun Li, Yikai Wang, Anbang Yao, Yi Qian, Xiao Zhou, and Ke He. Explicit connection distillation. 2020.

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint, arXiv:1506.01497*, 2015.

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

[11] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *ICLR*, 2020.

[12] Liu Xiaolong, Li Lujun, Li Chao, and Anbang Yao. Norm: Knowledge distillation via n-to-one representation matching. 2020.