Appendix of MixSKD: Self-Knowledge Distillation from Mixup for Image Recognition

Chuanguang Yang[1,2], Zhulin An[1(✉)], Helong Zhou[3], Linhang Cai[1,2], Xiang Zhi[1,2], Jiwen Wu[1], Yongjun Xu[1], and Qian Zhang[3]

[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Horizon Robotics
{yangchuanguang, anzhulin, cailinhang19g, zhixiang20g, xyj}@ict.ac.cn,
{helong.zhou, qian01.zhang}@horizon.ai

A Architectural Design of Auxiliary Branches

As discussed in the main paper, we attach one auxiliary branch b_k after each convolutional stage $\phi_k, k = 1, 2, \cdots, K-1$. Each auxiliary branch b_k includes a feature alignment module ζ_k and a linear classifier g_k. The feature alignment module contains several convolutional blocks following the original backbone network, *i.e.* residual block in ResNets [7]. To enable the fine-to-coarse feature transformation, we make the path from the input to the end of each auxiliary branch b_k have the same number of down-sampling as the backbone network f. We illustrate the overall architectures of various networks with auxiliary branches involved in the main paper, including ResNet [7], WRN [18], DenseNet [8] and HCGNet [16]. For better readability, the style of the illustration of architectural details is followed by the original paper.

Table 1. Architectural details of the backbone ResNet-18 [7] with auxiliary branches for CIFAR-100 classification.

Layer name	Output size	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$	$b_3(\cdot)$
conv1	32×32	$3\times3, 64$	-	-	-
conv2_x	32×32	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times2$	-	-	-
conv3_x	16×16	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	-	-
conv4_x	8×8	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	-
conv5_x	4×4	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$
Classifier	1×1	global average pool	global average pool	global average pool	global average pool
		100D fully-connected	100D fully-connected	100D fully-connected	100D fully-connected

Table 2. Architectural details of WRN-16-2 [18] with auxiliary classifiers for CIFAR-100 classification.

Layer name	Output size	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$
conv1	32×32	$3\times3, 16$	-	-
conv2_x	32×32	$\begin{bmatrix}3\times3,32\\3\times3,32\end{bmatrix}\times2$	-	-
conv3_x	16×16	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times2$	-
conv4_x	8×8	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$
Classifier	1×1	global average pool	global average pool	global average pool
		100D fully-connected	100D fully-connected	100D fully-connected

Table 3. Architectural details of DenseNet-40-12 [8] with auxiliary classifiers for CIFAR-100 classification.

Layers	Output size	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$
Convolution	32×32	$3\times3, 16$	-	-
Dense Block (1)	32×32	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$	-	-
Transition Layer (1)	32×32	1×1 conv	-	-
	16×16	2×2 average pool, stride 2	-	-
Dense Block (2)	16×16	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$	-
Transition Layer (2)	16×16	1×1 conv	1×1 conv	-
	8×8	2×2 average pool, stride 2	2×2 average pool, stride 2	-
Dense Block (3)	8×8	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$	$\begin{bmatrix}1\times1\text{ conv}\\3\times3\text{ conv}\end{bmatrix}\times12$
Classification Layer	1×1	global average pool	global average pool	global average pool
		100D fully-connected	100D fully-connected	100D fully-connected

Table 4. Architectural details of HCGNet-A1 [16] with auxiliary classifiers for CIFAR-100 classification.

Stage	IR	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$
Stem	32×32	3×3 Conv,24	-	-
Hybrid Block	32×32	SMG×8 ($k=12$)	-	-
Transition	32×32	SMG×1	SMG×1	-
Hybrid Block	16×16	SMG×8 ($k=24$)	SMG×8 ($k=24$)	-
Transition	16×16	SMG×1	SMG×1	SMG×1
Hybrid Block	8×8	SMG×8 ($k=36$)	SMG×8 ($k=36$)	SMG×8 ($k=36$)
Classification	1×1	global average pool	global average pool	global average pool
	-	100D FC, softmax	100D FC, softmax	100D FC, softmax

Table 5. Architectural details of ResNet-18 [7] with auxiliary classifiers for fined-grained classification. Here, N denotes the number of classes.

Layer name	Output size	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$	$b_3(\cdot)$
conv1	112×112	$7\times7, 64$, stride 2	-	-	-
conv2_x	56×56	3×3, max pool, stride 2	-	-	-
		$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times 2$	-	-	-
conv3_x	28×28	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times 2$	-	-
conv4_x	14×14	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times 2$	-
conv5_x	7×7	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times 2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times 2$
Classifier	1×1	global average pool	global average pool	global average pool	global average pool
		N-D fully-connected	N-D fully-connected	N-D fully-connected	N-D fully-connected

Table 6. Architectural details of ResNet-50 [7] with auxiliary classifiers for ImageNet classification.

Layer name	Output size	$f(\cdot)$	$b_1(\cdot)$	$b_2(\cdot)$	$b_3(\cdot)$
conv1	112×112	$7\times7, 64$, stride 2	-	-	-
conv2_x	56×56	3×3, max pool, stride 2	-	-	-
		$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times 3$	-	-	-
conv3_x	28×28	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times 4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times 4$	-	-
conv4_x	14×14	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times 6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times 6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times 6$	-
conv5_x	7×7	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times 3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times 3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times 3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times 3$
Classifier	1×1	global average pool	global average pool	global average pool	global average pool
		1000D fully-connected	1000D fully-connected	1000D fully-connected	1000D fully-connected

B Experimental setup

B.1 Image Classification

Dataset

- **CIFAR-100** [11] is a standard image classification dataset, containing 50k training images and 10k test images in 100 classes.
- **CUB-200-2011** [15] contains 200 species of birds with 5994 training images and 5794 test images.
- **Standford Dogs** [9] contains 120 breeds of dogs with 12000 training images and 8580 test images.
- **MIT67** [14] contains 67 indoor categories with 5356 training images and 1337 test images.
- **Stanford Cars** [10] contains 196 classes of cars with 8144 training images and 8041 testing images.
- **FGVC-Aircraft** [13] contains 100 classes of aircraft variants with 6667 training images and 3333 testing images.
- **ImageNet** [4] is a large-scale image classification dataset, which contains 1.28 million training images and 50k validation images in 1000 classes. ImageNet is also a hierarchical dataset that includes both coarse- and fine-grained class distinction.

Data pre-processing. We utilize the standard data pre-processing pipeline [8], *i.e.* random cropping and flipping. The resolution of each input image is 32×32 in CIFAR-100 and 224×224 in fine-grained datasets and ImageNet.

Training details

- **CIFAR-100:** all network are trained by stochastic gradient descent (SGD) optimizer with a momentum of 0.9, a weight decay of 5×10^{-4}, and a batch size of 128. We start at 5 epochs for linear warm-up from 0 to an initial learning rate of 0.1, which avoids the possible model collapse issue for data augmentation and Self-KD training. Then the learning rate is divided by 10 after the 105-th and 155-th epochs within the total 205 epochs.
- **Fine-grained classification:** all network are trained by stochastic gradient descent (SGD) optimizer with a momentum of 0.9, a weight decay of 1×10^{-4}, and a batch size of 32. We start at 5 epochs for linear warm-up from 0 to an initial learning rate of 0.1, which avoids the possible model collapse issue for data augmentation and Self-KD training. Then the learning rate is divided by 10 after the 105-th and 155-th epochs within the total 205 epochs.
- **ImageNet:** all network are trained by stochastic gradient descent (SGD) optimizer with a momentum of 0.9, a weight decay of 1×10^{-4}, and a batch size of 256. We start at 5 epochs for linear warm-up from 0 to an initial learning rate of 0.1, which avoids the possible model collapse issue for data augmentation and Self-KD training. Then we use a cosine learning rate scheduler from an initial learning rate of 0.1 to 0 throughout the 300 epochs.

Object Detection

- **COCO-2017** [12] contains 120k training images and 5k validation images. In this paper, we adopt 5k validation images for test.

Data pre-processing. We utilize the default data pre-processing of MMDetection [2]. The shorter side of the input image is resized to 800 pixels, the longer side is limited up to 1333 pixels.

Training details. We adopt a 1x training schedule with a momentum of 0.9 and a weight decay of 0.0001. We start at 500 linear warm-up iterations from 0 to an initial learning rate of 0.02. Then the learning rate is divided by 10 after the 8-th and 11-th epochs within the total 12 epochs. Training is conducted on 8 GPUs using synchronized SGD with a batch size of 1 per GPU.

Semantic Segmentation

- **Pascal VOC** [5] contains 10582/1449/1456 images for train/val/test with 21 semantic categories. Some training images are augmented with extra annotations provided by Hariharan *et al.* [6].
- **ADE20K** [19] contains 20k/2k/3k images for train/val/test with 150 semantic categories.
- **COCO-Stuff-164k** [1] covers 172 labels and contains 164k images: 118k for training, 5k for validation, 20k for test-dev and 20k for the test-challenge.

Data pre-processing. In this paper's semantic segmentation experiments, we retain the original training images and use validation images for test. Following the standard data augmentation [17], we employ random flipping and scaling in the range of [0.5, 2]. During the training phase, we use a crop size of 512×512. During the test phase, we utilize the original image size.

Training details. All experiments are optimized by SGD with a momentum of 0.9, a batch size of 16 and an initial learning rate of 0.02. The number of the total training iterations is 40K. The learning rate is decayed by $(1-\frac{iter}{total\_iter})^{0.9}$ following the polynomial annealing policy [3]. Training is conducted on 8 GPUs using synchronized SGD with a batch size of 2 per GPU. The implementation is based on an open codebase[4] released by Yang *et al.* [17].

[4] https://github.com/winycg/CIRKD

References

1. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: CVPR. pp. 1209–1218 (2018)
2. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al.: Mmdetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 (2019)
3. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE conference on computer vision and pattern recognition. pp. 248–255. IEEE (2009)
5. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
6. Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: ICCV. pp. 991–998. IEEE (2011)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
8. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
9. Khosla, A., Jayadevaprakash, N., Yao, B., Li, F.F.: Novel dataset for fine-grained image categorization: Stanford dogs. In: Proc. CVPR Workshop on Fine-Grained Visual Categorization (FGVC). vol. 2. Citeseer (2011)
10. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the IEEE international conference on computer vision workshops. pp. 554–561 (2013)
11. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Technical Report (2009)
12. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
13. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013)
14. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 413–420. IEEE (2009)
15. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. Technical Report (2011)
16. Yang, C., An, Z., Zhu, H., Hu, X., Zhang, K., Xu, K., Li, C., Xu, Y.: Gated convolutional networks with hybrid connectivity for image classification. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 12581–12588 (2020)
17. Yang, C., Zhou, H., An, Z., Jiang, X., Xu, Y., Zhang, Q.: Cross-image relational knowledge distillation for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12319–12328 (2022)
18. Zagoruyko S, K.N.: Wide residual networks. In: Proceedings of the British Machine Vision Conference (2016)
19. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)