Slimmed Asymmetrical Contrastive Learning and Cross Distillation for Lightweight Model Training

1 Supplementary Material

1.1 Algorithm

Algorithm 1: PyTorch-style pseudocode for the proposed algorithm

```
# f:  encoder model
# h:  projector head
# s:  slim ratio of SACL
# slicer:  SACL slicer
# alpha:  weight between CL loss and CD loss
# lambda:  weight on the off-diagonal terms
def normalize(z):
   z_norm = (z - z.mean(dim=0)) / z.std(dim=0)
   return z_norm

for batch in trainloader:
   x_a, x_b = batch

   # SACL forward pass
   slicer.remove_mask()
   z1 = h(f(x_a))
   slicer.activate_mask()
   z2 = h(f(x_b))

   # reverse the order of input
   with torch.no_grad():
      slicer.remove_mask()
      z1t = h(f(x_b))
      slicer.activate_mask()
      z2t = h(f(x_a))

   # cross correlation
   cab = mm(normalize(z1).T, normalize(z2)) / N
   caat = mm(normalize(z1).T, normalize(z1t)) / N
   cbbt = mm(normalize(z2).T, normalize(z2t)) / N

   # Contrastive leanring loss
   cl_loss = bt_loss(cab)

   # CD loss
   dcorr_a = off_diagonal(caat).mul_(lambda).sum()
   dcorr_b = off_diagonal(caat).mul_(lambda).sum()
   cd_loss = (dcorr_a + dcorr_b) / 2

   loss = cl_loss.mul(alpha) + cd_loss.mul(1-alpha)
   loss.backward()
   optimizer.step()
```

Preprint. Under review.

1.2 Compared to the Log-based distillation loss

From the perspective of knowledge distillation, the negative `logarithm`-based distillation loss has been widely incorporated into the "teacher-student" learning. In Section 3.2, we proposed the cross-distillation (XD) learning scheme. The distillation objective in Eq (10) is the inner decorrelation minimization between embeddings z and $[\tilde{z}]$. In addition to the correlation-based distillation loss, we also investigate the `negative logarithm` (e.g, $-a \log b$) distillation loss that is employed in both supervised knowledge distillation [3] and contrastive learning [1].

To avoid the unbalanced loss magnitude, the distillation loss is introduced as the regularization term controlled by the penalty level γ:

$$\mathcal{L} = \mathcal{L}_{\texttt{SACL}}(z_A, z_B) + \gamma \mathcal{L}_{CD} \tag{1}$$

$$\mathcal{L}_{CD} = (-[\tilde{z}_A] \log z_A + -[\tilde{z}_B] \log z_B)/2 \tag{2}$$

We empirically observe that the `negative logarithm`-based distillation loss failed to outperform the proposed cross-distillation loss $\mathcal{L}_{CD}$ with inner-decorrelation minimization. As shown in the ImageNet-100 results below:

Method	Encoder	# of Params (M)	Linear Eval Acc. (%)
XD	MobileNet-V1 (1×)	3.2	**80.30**
XD (w/ negative log)	MobileNet-V1 (1×)	3.2	79.63*
Barlow Twins [5]	MobileNet-V1 (1×)	3.2	78.40

*: Best accuracy we found with γ =1e-3.

Although the `negative-logarithm` distillation loss is suboptimal compared to the inner decorrelation minimization, the proposed cross-distillation learning scheme is beneficial to lightweight contrastive learning, compared to the baseline [5].

1.3 Detailed Experimental Setup of Pre-training

ImageNet-1K The encoders (MobileNet, EfficientNet, ResNet-50) are trained on ImageNet-1K with 100/200/300 epochs from scratch with the proposed method. We set the batch to 256 with a learning rate = 0.8. We employ the LARS optimizer with weight decay set to 1.5e-6. We set the correlation weights λ to 0.005. The hidden layer dimension of the projector is 4096. The detailed data augmentation is summarized in Table 1

Table 1: Detailed image augmentation settings on ImageNet-1K.

Parameter	X_A	X_B
Random crop size	224 × 224	224 × 224
Horizontal flip probability	0.5	0.5
Color jitter probability	0.8	0.8
Brightness adjustment probability	0.4	0.4
Contrast adjustment probability	0.4	0.4
Saturation adjustment probability	0.2	0.2
Hue adjustment probability	0.1	0.1
Gaussian blurring probability	1.0	0.1
Solarization probability	0.0	0.2

ImageNet-100 With the proposed cross-distillation method, we train the lightweight ViT model on the ImageNet-100 dataset for 400 epochs. The batch size is set to 256 with AdamW optimizer. The learning rate and weight decay are set to 0.005 and 1e-4. The detailed data augmentation is summarized in Table 2:

Table 2: Detailed image augmentation settings on ImageNet-100.

Parameter	X_A	X_B
Random crop size	224×224	224×224
Horizontal flip probability	0.5	0.5
Color jitter probability	0.8	0.8
Brightness adjustment probability	0.4	0.4
Contrast adjustment probability	0.4	0.4
Saturation adjustment probability	0.0	0.2
Hue adjustment probability	0.1	0.1
Gaussian blurring probability	1.0	0.1
Solarization probability	0.0	0.2

CIFAR-10 The proposed method is trained from scratch by 1,000 epochs with LARS-SGD optimizer [4]. We use 256 batch size along with 0.3 learning rate and $1e-4$ weight decay. The Cosine learning rate scheduler is used with 10 epochs of warmup training. The detailed data augmentation is summarized in Table 3.

Table 3: Detailed image augmentation settings on CIFAR-10.

Parameter	X_A	X_B
Random crop size	32×32	32×32
Horizontal flip probability	0.5	0.5
Color jitter probability	0.8	0.8
Brightness adjustment probability	0.4	0.4
Contrast adjustment probability	0.4	0.4
Saturation adjustment probability	0.2	0.2
Hue adjustment probability	0.1	0.1
Gaussian blurring probability	0.0	0.0
Solarization probability	0.0	0.2

1.4 Detailed Experimental Setup of Downstream Fine-tuning

We evaluate the transferability of the pre-trained lightweight model on downstream tasks, including CIFAR-10, CIFAR-100, and VOC2007. Following the settings in [2], we fine-tuned the models for 10,000 steps with SGD and batch size of 64. The learning rate is set to 0.1 with no weight decay. The input samples are resized to 224×224 to maintain the dimensionality as the pre-trained model. The checkpoint of the pre-trained lightweight model will be released soon.

References

[1] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, 2021.

[2] Linus Ericsson, Henry Gouk, and Timothy M Hospedales. How well do self-supervised models transfer? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[3] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[4] Yang You, Igor Gitman, and Boris Ginsburg. Large Batch Training of Convolutional Networks. *arXiv preprint arXiv:1708.03888*, 2017.

[5] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow Twins: Self-supervised Learning via Redundancy Reduction. In *International Conference on Machine Learning (ICML)*, pages 12310–12320, 2021.