DCD: Discriminative and Consistent Representation Distillation

Supplementary Material

6. Algorithm

Algorithm 1 provides the pseudo-code of DCD.

7. Implementation Details

We implement DCD in PyTorch following the implementation of CRD [25][1].

7.1. Baseline Methods

We compare our approach to the following state-of-the-art methods from the literature: (1) Knowledge Distillation (KD) [11]; (2) FitNets: Hints for Thin Deep Nets [22]; (3) Attention Transfer (AT) [29]; (4) Similarity-Preserving Knowledge Distillation (SP) [26]; (5) Correlation Congruence (CC) [21]; (6) Variational Information Distillation for Knowledge Transfer (VID) [1]; (7) Relational Knowledge Distillation (RKD) [19]; (8) Learning Deep Representations with Probabilistic Knowledge Transfer (PKT) [20]; (9) Knowledge Transfer via Distillation of Activation Boundaries Formed by Hidden Neurons (AB) [10]; (10) Paraphrasing Complex Network: Network Compression via Factor Transfer (FT) [13]; (11) A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning (FSP) [28]; (12) Like What You Like: Knowledge Distill via Neuron Selectivity Transfer (NST) [12]; (13) Contrastive Representation Distillation (CRD) [25]; (14) A Comprehensive Overhaul of Feature Distillation (OFD); (15) Rethinking Soft Labels for Knowledge Distillation: A Bias-Variance Tradeoff Perspective (WSLD) [32]; (16) Respecting Transfer Gap in Knowledge Distillation (IPWD) [18]; (17) Knowledge Distillation via Softmax Regression Representation Learning (SRRL) [27]; (18) Cross-Layer Distillation with Semantic Calibration (SemCKD) [2]; (19) Distilling Knowledge via Knowledge Review (ReviewKD) [5]; (20) Knowledge Distillation with the Reused Teacher Classifier (SimKD) [3]; (21) Searching A Fast Knowledge Distillation Process via Meta Optimization (DistPro) [6]; (22) Knowledge Distillation via N-to-One Representation Matching (NORM) [14]; (23) Information Theoretic Representation (ITRD) [17]; (24) Feature Kernel Distillation (FKD) [7]; (25) Complementary Relation Contrastive Distillation (CRCD) [33]; (26) Distilling Knowledge from Self-Supervised Teacher by Embedding Graph Alignment (EGA) [16]; (27) Wasserstein Contrastive Representation Distillation (WCoRD) [4].

[1] Available at: https://github.com/HobbitLong/RepDistiller.

7.2. Network Architectures

We use the following network architectures as described in [25]: (1) Wide Residual Network (WRN) [30], where WRN-d-w represents a wide ResNet with depth d and width factor w; (2) ResNet [8], where resnet-d represents a CIFAR-style ResNet with 3 groups of basic blocks having 16, 32, and 64 channels, respectively, and resnet-8 $\times 4$ and resnet-32 $\times 4$ indicate a 4-times wider network with 64, 128, and 256 channels; (3) ResNet [8], where ResNet-d represents an ImageNet-style ResNet with Bottleneck blocks and more channels; (4) MobileNet-v2 [23], using a width multiplier of 0.5 in our experiments; (5) VGG [24], where the VGG network used is adapted from its original ImageNet counterpart; and (6) ShuffleNet-v1 [31] and ShuffleNet-v2 [15], which are adapted for efficient training with input sizes of 32×32.

7.3. Optimization

All methods evaluated in our experiments use SGD with 0.9 Nesterov momentum. For CIFAR-100, we initialize the learning rate as 0.05, and decay it by 0.1 every 30 epochs after the first 150 epochs until the last 240 epoch. For MobileNet-v2, ShuffleNet-v1, and ShuffleNet-v2, we use a learning rate of 0.01 as this learning rate is optimal for these models in a grid search, while 0.05 is optimal for other models. The batch size is set to 64 for CIFAR-100, and the weight decay is set to 5×10^{-4}. For ImageNet[2] , the initial learning rate is set to 0.1 and then divided by 10 at the 30th, 60th, and 90th epochs of the total 120 training epochs. The mini-batch size is set to 256, and the weight decay is set to 1×10^{-4}. All results are reported as means over three trials, except for the results on ImageNet, which are reported in a single trial.

8. Results

8.1. Extended Comparison with State-of-the-Art

Table 7 and Table 8 provide a comprehensive overview of the top-1 accuracies of student networks trained with various state-of-the-art distillation techniques across a wide range of teacher-student architectural combinations. Our method benefits from its simplicity, as it has no trainable parameters, and the only hyperparameters involved are the loss coefficients.

8.2. Additional Inter-class Correlations

We present supplementary figures shown in Figure 4, which demonstrate the effectiveness of the DCD method across various student-teacher model architectures.

[2] Available at: https://www.image-net.org/.

Algorithm 1 Pseudocode of DCD in a PyTorch-like style.

```
# f_T, f_S: teacher and student networks
# t_dim: The input feature dimension for the teacher
# s_dim: The input feature dimension for the student
# feat_dim: The projection feature space dimension
# N: batch size

class DCDLoss(nn.Module):
   def __init__(self, s_dim, t_dim, feat_dim, init_tau=1.0, max_tau=10.0, init_b=0.0):
      super(DCDLoss, self).__init__()
      # learnable params
      self.tau = nn.Parameter(torch.tensor(init_tau))
      self.b = nn.Parameter(torch.tensor(init_b))

      # embedding layer
      self.embed_s = nn.Linear(s_dim, feat_dim)
      self.embed_t = nn.Linear(t_dim, feat_dim)

   def forward(self, f_s, f_t):
      f_s = self.embed_s(f_s)
      f_t = self.embed_t(f_t)

      f_s = F.normalize(f_s, dim=1)
      f_t = F.normalize(f_t, dim=1)

      tau = self.tau.exp().clamp(0, self.max_tau)

      # contrastive loss
      logits = torch.mm(f_s, f_t.t()) * tau + self.b
      labels = torch.arange(N)
      contrastive_loss = F.cross_entropy(logits, labels)

      # consistent loss
      p1 = F.log_softmax(logits, dim=1)
      p2 = F.softmax(logits, dim=0)
      consistent_loss = F.kl_div(p1, p2)

      return contrastive_loss + 0.5 * consistent_loss
```

9. Ablation Study

There are three main hyperparameters in our objective: the internal DCD coefficient α, which balances the contrastive and invariance losses within the DCD loss; the DCD loss coefficient β, which balances the DCD loss with other loss terms; and the loss coefficient λ, which is typically set to 1.0 but can be adjusted to affect the weighting of certain components. We conduct an ablation study to analyze the impact of these hyperparameters. For this study, we adopt WRN-40-2 as the teacher and WRN-16-2 as the student. Experiments are conducted on CIFAR-100, and the results are shown in Figure 5.

Ablation on loss coefficient α. We tested different values for α: 0.01, 0.1, 0.3, 0.5, 0.7, 1, 2, and 5. As shown in Figure 5a, our method remains robust across changes in α, with no significant difference in performance at low or high values. This robustness can be attributed to the adaptive temperature scaling mechanism, which enables automatic tuning of contrast level and logit scaling during training, providing more stable knowledge transfer.

Ablation on loss coefficient β. We varied β from 0.1 to 100, considering values of 0.1, 0.5, 1, 2, 5, 10, 50, and 100. As illustrated in Figure 5b, extremely high β values cause significant degradation in performance due to the overwhelming contribution of the DCD loss relative to other loss terms. Very low values of β also lead to a slight decrease in performance. The optimal range for β is between 0.5 and 10, suggesting that the DCD loss should be weighted similarly to other loss terms for the best results.

Ablation on loss coefficient λ. While λ is typically set to 1.0 [2, 3, 25], we tested values from 0.1 to 100, considering values of 0.1, 0.5, 1, 2, 5, 10, 50, and 100. As shown in Figure 5c, high values (i.e., $\lambda = 50$ and 100) lead to collapsing training. Lower values, such as 0.5 to 1.0, have similar performance.

10. Broader Impact

The presented research should be categorized as research in the field of knowledge distillation. The primary goal is to reduce computational demands, thereby lowering the energy requirements of AI systems and contributing to more sustainable technology deployment. However, this technique also harbors risks, notably the potential to perpetuate existing biases present in teacher models. Such biases could have profound ethical implications, as in sensitive applications. Furthermore, the versatility of the algorithms developed here enables their application across a broad spectrum of vision-related tasks, but this versatility also introduces the dual-use dilemma, where the technology might yield both beneficial

Table 7. Test top-1 accuracy (%) of student networks on CIFAR-100, comparing students and teachers of the same architecture using various distillation methods. ↑ denotes outperformance over KD and ↓ denotes underperformance.

Teacher Student	WRN-40-2 WRN-16-2	WRN-40-2 WRN-40-1	resnet-56 resnet-20	resnet-110 resnet-20	resnet-110 resnet-32	resnet-32x4 resnet-8x4	VGG-13 VGG-8
Teacher	75.61	75.61	72.34	74.31	74.31	79.42	74.64
Student	73.26	71.98	69.06	69.06	71.14	72.50	70.36
KD [11]	74.92	73.54	70.66	70.67	73.08	73.33	72.98
FitNet [22]	73.58 (↓)	72.24 (↓)	69.21 (↓)	68.99 (↓)	71.06 (↓)	73.50 (↑)	71.02 (↓)
AT [29]	74.08 (↓)	72.77 (↓)	70.55 (↓)	70.22 (↓)	72.31 (↓)	73.44 (↑)	71.43 (↓)
SP [26]	73.83 (↓)	72.43 (↓)	69.67 (↓)	70.04 (↓)	72.69 (↓)	72.94 (↓)	72.68 (↓)
CC [21]	73.56 (↓)	72.21 (↓)	69.63 (↓)	69.48 (↓)	71.48 (↓)	72.97 (↓)	70.81 (↓)
VID [1]	74.11 (↓)	73.30 (↓)	70.38 (↓)	70.16 (↓)	72.61 (↓)	73.09 (↓)	71.23 (↓)
RKD [19]	73.35 (↓)	72.22 (↓)	69.61 (↓)	69.25 (↓)	71.82 (↓)	71.90 (↓)	71.48 (↓)
PKT [20]	74.54 (↓)	73.45 (↓)	70.34 (↓)	70.25 (↓)	72.61 (↓)	73.64 (↑)	72.88 (↓)
AB [10]	72.50 (↓)	72.38 (↓)	69.47 (↓)	69.53 (↓)	70.98 (↓)	73.17 (↓)	70.94 (↓)
FT [13]	73.25 (↓)	71.59 (↓)	69.84 (↓)	70.22 (↓)	72.37 (↓)	72.86 (↓)	70.58 (↓)
FSP [28]	72.91 (↓)	n/a	69.95 (↓)	70.11 (↓)	71.89 (↓)	72.62 (↓)	70.33 (↓)
NST [12]	73.68 (↓)	72.24 (↓)	69.60 (↓)	69.53 (↓)	71.96 (↓)	73.30 (↓)	71.53 (↓)
CRD [25]	75.48 (↑)	74.14 (↑)	71.16 (↑)	71.46 (↑)	73.48 (↑)	75.51 (↑)	73.94 (↑)
CRD+KD [25]	75.64 (↑)	74.38 (↑)	71.63 (↑)	71.56 (↑)	73.75 (↑)	75.46 (↑)	74.29 (↑)
OFD [9]	75.24 (↑)	74.33 (↑)	70.38 (↓)	n/a	73.23 (↑)	74.95 (↑)	73.95 (↑)
WSLD [32]	n/a	73.74 (↑)	71.53 (↑)	n/a	73.36 (↑)	74.79 (↑)	n/a
IPWD [18]	n/a	74.64 (↑)	71.32 (↑)	n/a	73.91 (↑)	76.03 (↑)	n/a
SRRL [27]	n/a	74.64 (↑)	n/a	n/a	n/a	75.39 (↑)	n/a
SemCKD [2]	n/a	74.41 (↑)	n/a	n/a	n/a	76.23 (↑)	n/a
ReviewKD [5]	76.12 (↑)	75.09 (↑)	71.89 (↑)	n/a	73.89 (↑)	75.63 (↑)	74.84 (↑)
SimKD [3]	n/a	75.56 (↑)	n/a	n/a	n/a	78.08 (↑)	n/a
DistPro [6]	76.36 (↑)	n/a	72.03 (↑)	n/a	73.74 (↑)	n/a	n/a
NORM [14]	75.65 (↑)	74.82 (↑)	71.35 (↑)	71.55 (↑)	73.67 (↑)	76.49 (↑)	73.95 (↑)
NORM+KD [14]	76.26 (↑)	75.42 (↑)	71.61 (↑)	72.00 (↑)	74.95 (↑)	76.98 (↑)	74.46 (↑)
NORM+CRD [14]	76.02 (↑)	75.37 (↑)	71.51 (↑)	71.90 (↑)	73.81 (↑)	76.49 (↑)	73.58 (↑)
WCoRD [4]	75.88 (↑)	74.73 (↑)	71.56 (↑)	71.57 (↑)	73.81 (↑)	75.95 (↑)	74.55 (↑)
WCoRD+KD [4]	76.11 (↑)	74.72 (↑)	71.92 (↑)	71.88 (↑)	74.20 (↑)	76.15 (↑)	74.72 (↑)
CRCD [33]	76.67 (↑)	75.95 (↑)	73.21 (↑)	72.33 (↑)	74.98 (↑)	76.42 (↑)	74.97 (↑)
FKD [7]	n/a	n/a	n/a	n/a	n/a	75.57 (↑)	73.78 (↑)
ITRD (corr) [17]	75.85 (↑)	74.90 (↑)	71.45 (↑)	71.77 (↑)	74.02 (↑)	75.63 (↑)	74.70 (↑)
ITRD (corr+mi) [17]	76.12 (↑)	75.18 (↑)	71.47 (↑)	71.99 (↑)	74.26 (↑)	76.19 (↑)	74.93 (↑)
DCD (ours)	74.99 (↑)	73.69 (↑)	71.18 (↑)	71.00 (↑)	73.12 (↑)	74.23 (↑)	73.22 (↑)
DCD+KD (ours)	76.06 (↑)	74.76 (↑)	71.81 (↑)	72.03 (↑)	73.62 (↑)	75.09 (↑)	73.95 (↑)

and adverse impacts. Thus, careful consideration is needed when deploying these methods to ensure they align with ethical guidelines and promote fairness in AI applications.

References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9163–9171, 2019. 1, 3, 4

[2] Defang Chen, Jian-Ping Mei, Yuan Zhang, Can Wang, Yan Feng, and Chun Chen. Cross-layer distillation with semantic calibration, 2021. 1, 2, 3, 4

[3] Defang Chen, Jian-Ping Mei, Hailin Zhang, Can Wang, Yan Feng, and Chun Chen. Knowledge distillation with the reused teacher classifier, 2022. 1, 2, 3, 4

[4] Liqun Chen, Dong Wang, Zhe Gan, Jingjing Liu, Ricardo Henao, and Lawrence Carin. Wasserstein contrastive representation distillation, 2021. 1, 3, 4

[5] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review, 2021. 1, 3, 4

[6] Xueqing Deng, Dawei Sun, Shawn Newsam, and Peng Wang. Distpro: Searching a fast knowledge distillation process via meta optimization, 2022. 1, 3, 4

Table 8. Test top-1 accuracy (%) of student networks on CIFAR-100 involving students and teachers from different architectures, using various distillation methods. ↑ denotes outperformance over KD and ↓ denotes underperformance.

Teacher Student	VGG-13 MobileNet-v2	ResNet-50 MobileNet-v2	ResNet-50 VGG-8	ResNet-32x4 ShuffleNet-v1	ResNet-32x4 ShuffleNet-v2	WRN-40-2 ShuffleNet-v1
Teacher	74.64	79.34	79.34	79.42	79.42	75.61
Student	64.60	64.60	70.36	70.5	71.82	70.5
KD [11]	67.37	67.35	73.81	74.07	74.45	74.83
FitNet [22]	64.14 (↓)	63.16 (↓)	70.69 (↓)	73.59 (↓)	73.54 (↓)	73.73 (↓)
AT [29]	59.40 (↓)	58.58 (↓)	71.84 (↓)	71.73 (↓)	72.73 (↓)	73.32 (↓)
SP [26]	66.30 (↓)	68.08 (↑)	73.34 (↓)	73.48 (↓)	74.56 (↑)	74.52 (↓)
CC [21]	64.86 (↓)	65.43 (↓)	70.25 (↓)	71.14 (↓)	71.29 (↓)	71.38 (↓)
VID [1]	65.56 (↓)	67.57 (↑)	70.30 (↓)	73.38 (↓)	73.40 (↓)	73.61 (↓)
RKD [19]	64.52 (↓)	64.43 (↓)	71.50 (↓)	72.28 (↓)	73.21 (↓)	72.21 (↓)
PKT [20]	67.13 (↓)	66.52 (↓)	73.01 (↓)	74.10 (↑)	74.69 (↑)	73.89 (↓)
AB [10]	66.06 (↓)	67.20 (↓)	70.65 (↓)	73.55 (↓)	74.31 (↓)	73.34 (↓)
FT [13]	61.78 (↓)	60.99 (↓)	70.29 (↓)	71.75 (↓)	72.50 (↓)	72.03 (↓)
NST [12]	58.16 (↓)	64.96 (↓)	71.28 (↓)	74.12 (↑)	74.68 (↑)	76.09 (↑)
CRD [25]	69.73 (↑)	69.11 (↑)	74.3 (↑)	75.11 (↑)	75.65 (↑)	76.05 (↑)
CRD+KD [25]	69.94 (↑)	69.54 (↑)	74.58 (↑)	75.12 (↑)	76.05 (↑)	76.27 (↑)
OFD [9]	69.48 (↑)	69.04 (↑)	n/a	75.98 (↑)	76.82 (↑)	75.85 (↑)
WSLD [32]	n/a	68.79 (↑)	73.80 (↓)	75.09 (↑)	n/a	75.23 (↑)
IPWD [18]	n/a	70.25 (↑)	74.97 (↑)	76.03 (↑)	n/a	76.44 (↑)
SRRL [27]	n/a	n/a	n/a	75.18 (↑)	n/a	n/a
SemCKD [2]	n/a	n/a	n/a	n/a	77.62 (↑)	n/a
ReviewKD [5]	70.37 (↑)	69.89 (↑)	n/a	77.45 (↑)	77.78 (↑)	77.14 (↑)
SimKD [3]	n/a	n/a	n/a	77.18 (↑)	n/a	n/a
DistPro [6]	n/a	n/a	n/a	77.18 (↑)	77.54 (↑)	77.24 (↑)
NORM [14]	68.94 (↑)	70.56 (↑)	75.17 (↑)	77.42 (↑)	78.07 (↑)	77.06 (↑)
NORM+KD [14]	69.38 (↑)	71.17 (↑)	75.67 (↑)	77.79 (↑)	78.32 (↑)	77.63 (↑)
NORM+CRD [14]	69.17 (↑)	71.08 (↑)	75.51 (↑)	77.50 (↑)	77.96 (↑)	77.09 (↑)
WCoRD [4]	69.47 (↑)	70.45 (↑)	74.86 (↑)	75.40 (↑)	75.96 (↑)	76.32 (↑)
WCoRD+KD [4]	70.02 (↑)	70.12 (↑)	74.68 (↑)	75.77 (↑)	76.48 (↑)	76.68 (↑)
CRCD [33]	n/a	n/a	n/a	n/a	n/a	n/a
FKD [7]	n/a	n/a	74.61 (↑)	75 (↑)	n/a	n/a
ITRD (corr) [17]	69.97 (↑)	71.41 (↑)	75.71 (↑)	76.8 (↑)	77.27 (↑)	77.35 (↑)
ITRD (corr+mi) [17]	70.39 (↑)	71.34 (↑)	75.49 (↑)	76.91 (↑)	77.40 (↑)	77.09 (↑)
DCD (ours)	68.35 (↑)	67.39 (↑)	73.85 (↑)	74.26 (↑)	75.26 (↑)	74.98 (↑)
DCD+KD (ours)	69.77 (↑)	70.03 (↑)	74.08 (↑)	76.01 (↑)	76.95 (↑)	76.51 (↑)

[7] Bobby He and Mete Ozay. Feature kernel distillation. In *International Conference on Learning Representations*, 2022. 1, 3, 4

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 1

[9] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation, 2019. 3, 4

[10] Byeongho Heo, Minsik Lee, Seong Joon Yun, Jin Young Choi, and In So Kweon. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3779–3787, 2019. 1, 3, 4

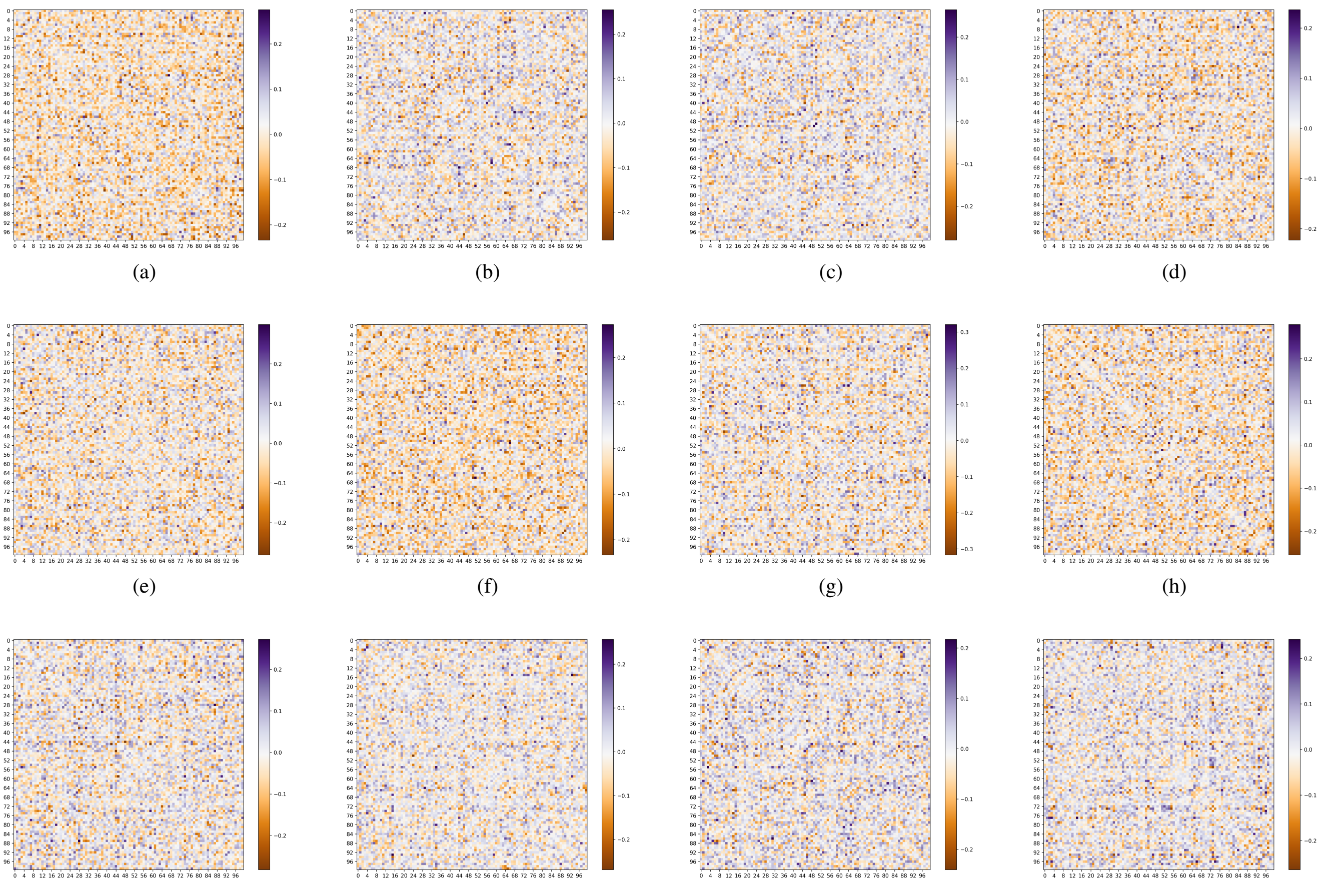
[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 1, 3, 4

[12] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. In *Advances in Neural Information Processing Systems*, pages 185–195, 2017. 1, 3, 4

[13] Jangho Kim, Seongwon Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. In *Advances in Neural Information Processing Systems*, pages 2760–2769, 2018. 1, 3, 4

[14] Xiaolong Liu, Lujun Li, Chao Li, and Anbang Yao. Norm: Knowledge distillation via n-to-one representation matching, 2023. 1, 3, 4

[15] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architec-

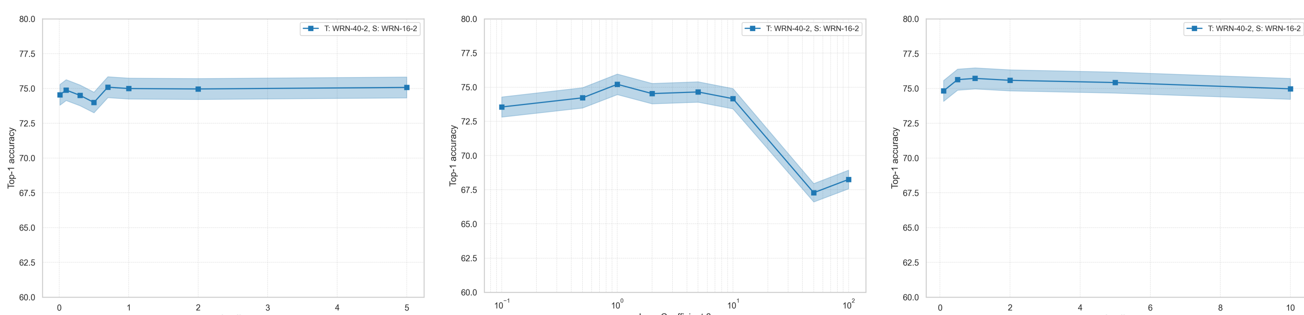


(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Figure 4. Comparison of correlation matrix differences between teacher and student logits across varied student-teacher architectures on the CIFAR-100 task: (a) S: WRN-16-2, T: WRN-40-2; (b) S: resnet-20, T: resnet-56; (c) S: resnet-20, T: resnet-110; (d) S: resnet-32, T: resnet-110; (e) S: resnet-8x4, T: resnet-32x4; (f) S: VGG-8, T: VGG-13; (g) S: MobileNet-v2, T: VGG-13; (h) S: MobileNet-v2, T: ResNet-50; (i) S: VGG-8, T: ResNet-50; (j) S: ShuffleNet-v1, T: ResNet-32x4; (k) S: ShuffleNet-v2, T: ResNet-32x4; (l) S: ShuffleNet-v1, T: WRN-40-2.



(a) Effect of varying α (b) Effect of varying β (c) Effect of varying λ

Figure 5. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. (a) Effect of the internal DCD coefficient α on performance. (b) Effect of DCD loss coefficient β on performance (logarithmic scale). (c) Effect of loss coefficient λ on performance.

ture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 116–131, 2018. 1

[16] Yuchen Ma, Yanbei Chen, and Zeynep Akata. Distilling knowledge from self-supervised teacher by embedding graph

alignment, 2022. 1

[17] Roy Miles, Adrian Lopez Rodriguez, and Krystian Mikolajczyk. Information theoretic representation distillation, 2022. 1, 3, 4

[18] Yulei Niu, Long Chen, Chang Zhou, and Hanwang Zhang. Respecting transfer gap in knowledge distillation, 2022. 1, 3, 4

[19] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019. 1, 3, 4

[20] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 268–284, 2018. 1, 3, 4

[21] Baoyun Peng, Xi Li, Yifan Wu, Yizhou Fan, Bo Wang, Qi Tian, and Jun Liang. Correlation congruence for knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5007–5016, 2019. 1, 3, 4

[22] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *Proceedings of the 4th International Conference on Learning Representations*, 2014. 1, 3, 4

[23] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 1

[24] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 1

[25] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation, 2022. 1, 2, 3, 4

[26] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1365–1374, 2019. 1, 3, 4

[27] Jing Yang, Brais Martinez, Adrian Bulat, and Georgios Tzimiropoulos. Knowledge distillation via softmax regression representation learning. In *International Conference on Learning Representations*, 2021. 1, 3, 4

[28] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4133–4141, 2017. 1, 3

[29] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *Proceedings of the 5th International Conference on Learning Representations*, 2016. 1, 3, 4

[30] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks, 2017. 1

[31] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018. 1

[32] Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective, 2021. 1, 3, 4

[33] Jinguo Zhu, Shixiang Tang, Dapeng Chen, Shijie Yu, Yakun Liu, Aijun Yang, Mingzhe Rong, and Xiaohua Wang. Complementary relation contrastive distillation, 2021. 1, 3, 4