Local Dense Logit Relations for Enhanced Knowledge Distillation

Supplementary Material

1. Optimization Objective

The algorithm for implementing the LDRLD is provided in Algorithm 1, detailing the essential steps.

Algorithm 1 Pseudo code for LDRLD

```
Require: $\mathcal{D}$: training dataset; $T_{\text{net}}$: pre-trained teacher network; $S_{\text{net}}$: student network with parameters $\theta$; $\eta$: learning rate
Ensure: Trained parameters $\theta$ of the student network $S_{\text{net}}$
Load pre-trained teacher network $T_{\text{net}}$
repeat
    Randomly select a mini-batch $\mathcal{B}$ from $\mathcal{D}$
    for each $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{B}$ do
        $\mathbf{Z}^t \leftarrow T_{\text{net}}(\mathbf{x}_i)$   ▷ Compute teacher's output for sample $i$ and teacher's parameters are fixed.
        $\mathbf{Z}^s \leftarrow S_{\text{net}}(\mathbf{x}_i)$   ▷ Compute student's output for sample $i$.
        Compute the task loss $\mathcal{L}_{\text{Task}}$ (i.e., cross entropy).
        Compute the local logit relational knowledge $\mathcal{L}^w$ using Equation (19).
        Compute the remaining non-target knowledge $\mathcal{L}_{\text{RNTK}}$ using Equation (20).
        Compute the total loss $\mathcal{L}_{\text{LDRLD}}$ by combining the task loss $\mathcal{L}_{\text{Task}}$, $\mathcal{L}^w$, and $\mathcal{L}_{\text{RNTK}}$, as described in Equation (21).
    end for
    $\mathcal{L} \leftarrow \frac{1}{|\mathcal{B}|} \sum_i \mathcal{L}_{\text{LDRLD}}$
    Update $\theta \leftarrow \theta - \eta \left(\nabla_\theta \mathcal{L}\right)$
until Convergence criterion is maximum iterations reached
return $\theta$
```

2. Experiments

2.1. Experimental Setups

In our experiments, we evaluate our method using the following five classical datasets.

CIFAR-100 [28] comprises 100 classes, each image with a resolution of 32×32 pixels. The dataset contains 50,000 training images and 10,000 validation images.

ImageNet-1K (ILSVRC2012) [5] is a comprehensive dataset comprising 1,000 classes. The dataset contains 1.2 million training images and 50,000 validation images.

Tiny-ImageNet [30] is a streamlined version of the ImageNet-1K dataset. The dataset includes 200 classes, with images of 64×64 pixels resolution. It comprises 100,000 training images and 10,000 validation images.

Market-1501 [88] is a benchmark dataset for person re-identification (Re-ID), containing 1,501 unique identities and 32,668 images. It includes a query set of 3,368 images covering 751 identities (4–6 images each) and a gallery set of 30,368 images from 12 camera viewpoints.

MS-COCO2017 [35] is a widely used large-scale dataset for object detection, consisting of 80 categories, with 118,000 images in the train2017 split and 5,000 images in the val2017 split.

2.2. Implementations: Comparison with state-of-the-art KD Methods.

In this work, we compare the proposed method with classical benchmarks across several datasets, including CIFAR-100, ImageNet-1K, Market-1501, Tiny-ImageNet, and COCO2017. The comparison focuses on two main categories of KD methods. We evaluate the following feature-based KD methods: FitNets [51], SP [63], CC [46], VID [1], AT [27], OFD [17], PKT [45], NST [23], RKD [44], FT [25],CRD [62], SAKD [58], ReviewKD [3], NORM [38],FCFD [37], DiffKD [22], and CAT-KD [11]. We also evaluate the following logit-based KD methods such as: KD [18], DKD [84], NKD [76], CTKD [32], LCKA [91], WTTM [87], IPWD [43], SDD [66], and LSKD [60].

2.3. Implementations: Selection of Model Architectures.

To thoroughly evaluate the effectiveness of our proposed method, we employed several classical network architectures for image classification, including ResNet [15] (with its variants), VGG [56], MobileNetV1 [20], WideResNet [80] (WRN), ShuffleNetV1 [82], ShuffleNetV2 [39], and MobileNetV2 [52]. In our experimental framework, we used a variety of comparative approaches to assess the performance of different teacher-student pairs. We conducted experiments on the CIFAR-100, Tiny-ImageNet, and ImageNet-1K datasets using student networks with identical and different architectures. Additionally, we performed experiments with teacher-student pairs using identical architectures on the Market-1501 dataset. These experiments validated the effectiveness and consistency of our method across various datasets and network architectures.

2.4. Implementations: Training Details.

For both the Tiny-ImageNet and CIFAR-100 datasets: We train the models for 240 epochs using the Stochastic Gradient Descent optimizer with a momentum of 0.9 and a weight decay of 5×10^{-4}. The learning rate is reduced

Teacher Student	VGG13 MobileNetV2	ResNet50 MobileNetV2	ResNet50 VGG8	ResNet32×4 ShuffleNetV1	ResNet32×4 ShuffleNetV2	WRN-40-2 ShuffleNetV1
LDRLD	$\alpha = 7.0, \beta = 4.0$	$\alpha = 11.0, \beta = 7.0$	$\alpha = 9.5, \beta = 3.5$	$\alpha = 8.0, \beta = 8.0$	$\alpha = 9.5, \beta = 7.0$	$\alpha = 11.0, \beta = 7.0$

Table 7. Hyperparameters for heterogeneous architecture distillation on CIFAR-100 dataset.

Teacher Student	WRN-40-2 WRN-16-2	WRN-40-2 WRN-40-1	ResNet56 ResNet20	ResNet110 ResNet20	ResNet110 ResNet32	ResNet32×4 ResNet8×4	VGG13 VGG8
LDRLD	$\alpha = 11.5, \beta = 7.0$	$\alpha = 9.0, \beta = 4.0$	$\alpha = 9.5, \beta = 1.0$	$\alpha = 6.5, \beta = 1.0$	$\alpha = 8.0, \beta = 1.0$	$\alpha = 10.5, \beta = 7.0$	$\alpha = 11.0, \beta = 8.5$

Table 8. Hyperparameters for homogeneous architecture distillation on CIFAR-100 dataset.

Teacher Student	WRN-40-2 WRN-16-2	ResNet56 ResNet20	ResNet110 ResNet20	VGG13 MobileNetV2	VGG13 VGG8
LDRLD	$\alpha = 8.5, \beta = 5.0$	$\alpha = 8.0, \beta = 4.0$	$\alpha = 8.5, \beta = 5.0$	$\alpha = 6.0, \beta = 4.0$	$\alpha = 8.5, \beta = 4.5$

Table 9. Hyperparameters for homogeneous and heterogeneous architectures distillation on Tiny-ImageNet dataset.

Distillation	Teacher Student	ResNet34 ResNet18	ResNet50 MobileNetV1
LDRLD	α	7.0	5.0
	β	0.025	2.0

Table 10. Hyperparameters for homogeneous and heterogeneous architectures distillation on the ImageNet-1K dataset.

by a factor of 10 at the 150th, 180th, and 210th epochs. Data augmentation is performed using random horizontal flipping. We set the recursion depth to the default value of $d = 7$. The temperature coefficient τ is set to 4.0 across all experiments. The specific training details are as follows:

- (1) For the CIFAR-100 dataset, we use a batch size of 64. The learning rate is set to 0.05 for all architectures except MobileNetV2 [52] and ShuffleNet [39, 82], for which it is set to 0.01. The hyperparameters are provided in Table 7 and Table 8, where the recursion depth is set to $d = 9$ for ResNet32×4 vs. ShuffleNetV1 and to $d = 7$ for all other models. All models are trained with a linear warm-up for 20 epochs. See CRD [62] for detailed training details.
- (2) For the Tiny-ImageNet dataset, we use a batch size of 128. The learning rate is set to 0.1 for all models except MobileNetV2 [52], for which it is set to 0.02. The weighting coefficient for the cross-entropy is set to 1.0. All models are trained with a linear warm-up for 20 epochs. For more details on the hyperparameters, refer to Table 9.

For training on the ImageNet-1K dataset: We train the models for 100 epochs using the Stochastic Gradient Descent optimizer with a weight decay of 1×10^{-4}. The batch size of 256 is used for all models, and the initial learning rate is set to 0.1. A linear warm-up is applied during the first 10 epochs. The learning rate is then reduced by a factor of 10 at the 30th, 60th, and 90th epochs. The loss of cross-entropy is weighted by a coefficient of 1.0, and the temperature coefficient τ is set to 2.0. For model pairing, we use ResNet34 as the teacher and ResNet18 as the student when using the same architecture. For different architectures, ResNet50 serves as the teacher and MobileNetV1 as the student. For further details on the hyperparameters, please refer to the Table 10.

For training on the Matket-1501 dataset: We employ ResNet50 as the backbone to extract features for the teacher network and ResNet18 for the student network. To evaluate the performance of person Re-identification (Re-ID), we use two commonly applied metrics: Cumulative Matching Characteristics (CMC) and mean Average Precision (mAP). The CMC-k metric (i.e., Rank-k matching accuracy) measures the probability that a correct match appears in the top-k ranked retrieval results. CMC performs well when there is only one ground truth per query because it focuses solely on the first correct match among the top-k results during the evaluation. For more detailed explanations of these metrics, please refer to [2, 78].

3. Ablation Study

3.1. Hyperparameter Sensitivity

Impact of α and β on CIFAR-100. The results presented in Tables 11 and 12 show the accuracy of student (%) with different values of α and β across various architectures. We evaluated the impact of specific parameter values on student performance while keeping other parameters fixed. As shown in Table 11, for homogeneous pairs of teacher-student, the experimental results show optimal student performance with α = 9.5 and β = 1.0. Similarly, as shown in Table 12, for heterogeneous pairs of teacher-student, the experimental results demonstrate optimal student performance with α = 9.5 and β = 7.0. These results indicate that the optimal parameter settings differ across architectures.

α	6.0	6.5	7.0	7.5	8.0	8.5	9.0	**9.5**	10.0
Acc	71.87	72.03	71.95	71.94	71.96	71.94	71.90	**72.20**	71.85
β	0.75	**1.0**	1.25	1.5	2.0	2.25	2.5	3.0	4.0
Acc	72.01	**72.20**	71.62	72.04	71.81	71.92	72.05	71.86	71.78

Table 11. Impact of α and β on student's performance(%) on CIFAR-100 using ResNet56 as teacher and ResNet20 as student.

α	6.5	7.0	7.5	8.0	8.5	9.0	**9.5**	10.0	10.5
Acc	77.15	77.10	76.99	76.98	77.14	77.30	**77.33**	76.93	76.87
β	5.0	5.5	6.0	6.5	**7.0**	7.5	8.0	8.5	9.0
Acc	77.01	77.03	76.65	76.92	**77.33**	77.04	76.83	77.02	76.75

Table 12. Impact of α and β on student's performance(%) on CIFAR-100 using ResNet32×4 as teacher and ShuffleNetV2 as student.

Impact of α and β on Tiny-ImageNet. The results presented in Tables 13 and 14 show the student accuracies (%) across various architectures for different values of α and β, while holding other parameters fixed. Specifically, for the teacher-student pair WRN-40-2 and WRN-16-2 (Table 13), the best performance is achieved with $\alpha = 8.5$ and $\beta = 5.0$. Similarly, for the teacher-student pair VGG13 and MobileNetV2 (Table 14), the optimal performance occurs at $\alpha = 6.0$ and $\beta = 4.0$. These results demonstrate that the optimal parameter settings vary across different architectures. For more detailed parameter settings, see Table 9.

α	5.0	5.5	6.0	6.5	7.0	7.5	8.0	**8.5**
Acc	60.65	60.36	60.58	60.26	60.62	60.63	60.49	**60.67**
β	4.5	**5.0**	5.5	6.0	6.5	7.0	7.5	8.0
Acc	60.58	**60.67**	60.65	60.62	60.29	60.40	60.48	60.38

Table 13. Effect of α and β on student's performance on Tiny-Imagenet using WRN-40-2 as teacher and WRN-16-2 as student.

α	5.0	5.5	**6.0**	6.5	7.0	7.5	8.0	8.5
Acc	60.24	60.00	**60.31**	60.29	60.27	60.19	60.02	59.98
β	1.0	2.0	3.0	3.5	**4.0**	4.5	5.0	5.5
Acc	59.73	60.05	60.06	60.18	**60.31**	60.05	59.77	60.08

Table 14. Effect of α and β on student's performance on Tiny-Imagenet using VGG13 as teacher and MobileNetV2 as student.

3.2. Ablation and explanation of ADW

(1) Results shown in Table 15 demonstrate that each loss improves performance while combining both losses achieves the best outcome. (2) $\mathcal{L}_{Local}$ still outperforms the baseline without ADW in Table 5. However, without ADW, using uniform weights for logit pairs fails to mitigate class confusion caused by semantic gaps. In contrast, ADW can enhance fine-grained logit discrimination and improve performance in Table 5.

3.3. Training Efficiency

To evaluate the training cost of our method, we compare it with classical KD techniques by reporting the average runtime per epoch, as shown in Table 16. By incorporating local logit decoupling and combination mechanisms, our method maintains high training efficiency without additional storage and with only a slight increase in runtime. Although LDRLD requires additional training time, it requires GPU memory very similar to logit-based KD methods such as KD and DKD without increasing memory consumption. Furthermore, compared to feature-based KD methods such as CRD and ReviewKD, our approach achieves more efficient training.

$\mathcal{L}_{Local}$	$\mathcal{L}_{RNTK}$	ResNet32×4 ResNet8×4	WRN-40-2 WRN-40-1	ResNet32×4 ShuffleNetV2	WRN-40-2 ShuffleNetV1
-	-	72.50	71.98	71.82	70.50
✓		76.55 (+4.05)	74.45 (+2.47)	76.23 (+4.41)	76.11 (+5.61)
	✓	75.78 (+3.28)	74.19 (+2.21)	76.30 (+4.48)	76.51 (+6.01)
✓	✓	**77.20** (+4.70)	**74.98** (+3.00)	**77.33** (+5.51)	**77.09** (+6.59)

Table 15. Ablation of loss on student's performance on CIFAR100.

Distillation Manner	Feature-based		Logit-based		
Method	CRD	ReviewKD KD	DKD	LDRLD	
Time(s)	17.86	23.44	11.89	12.04	13.65
GPU memory (M)	3064	3148	2052	2052	2052
Accuracy(%)	75.51	75.63	73.33	76.32	77.20

Table 16. We assess the average training time (per epoch) and test accuracy (%) using a GeForce RTX 3090 on the CIFAR-100 dataset for ResNet32×4 (teacher) and ResNet8×4 (student).

4. Exploratory Experiments

4.1. Differences between NCKD and RNTK

DKD decoupling results in NCKD [84], which excludes only the target class, whereas RNTK excludes the top-1 to top-d most confident classes. The latter strategy avoids suppressing high-confidence categories, thereby reducing dependence on common categories, improving the ability to recognize unseen data, and promoting the activation of low-confidence classes. The experimental results demonstrate that the student with RNTK achieves the best performance when d=7, as shown in Table 17, outperforming NCKD and supporting this argument.

Model	NCKD	RNTK			
T:VGG13	-	d = 3	d = 5	d = 7	d = 9
S:VGG8	74.15	73.76	74.19	**74.28**	74.26
T:ResNet32×4	-	d = 3	d = 5	d = 7	d = 9
S:ResNet8×4	75.42	75.34	75.73	**75.78**	75.61

Table 17. A comparison of the top-1 performance of students trained with NCKD and RNTK on the CIFAR-100 dataset.

4.2. Influence of incorrect teacher predictions

We consider the case that the teacher model is poorly calibrated, biased, or makes incorrect predictions, and these issues could propagate to the student model during distillation. In order to deal with these problems, we introduce noisy labels with a noise rate of 0.1 [13], and the experimental results in Table 18 show that even if the teacher is biased or inaccurate, the LDRLD can still effectively transfer knowledge and perform robust. This shows, indicating that the recursive combination strategy may correct the influence of incorrect logits,.

Teacher Student	ResNet110 ResNet20	ResNet110 ResNet32	ResNet32×4 ResNet8×4	VGG13 VGG8
T:Accuracy	74.31	74.31	79.42	74.64
S:Accuracy	69.06	71.14	72.50	70.36
DKD	70.91	74.11	76.32	74.64
DKD(0.1)	71.32	73.81	76.12	74.32
Δ	+0.41	-0.30	-0.20	-0.32
LDRLD	71.98	74.16	77.20	75.06
LDRLD(0.1)	72.08	74.13	77.32	74.91
Δ	+0.10	-0.03	+0.12	-0.15

Table 18. The top-1 performance of students with noisy labels on the CIFAR-100 datasets.

4.3. Predictions of different teachers

To address knowledge capacity gap, we designed different teachers for the same student, as shown in Table 19. As the accuracy of the teacher increases, LDRLD generally delivers logit knowledge more accurately than DKD, thereby enabling the student's performance to more closely approximate that of the teacher, thanks to the accurate leverage of inter-class information by logit combinations.

Teacher Student	ResNet32 ResNet20	ResNet44 ResNet20	ResNet56 ResNet20	ResNet110 ResNet20
T:Accuracy	71.49	72.32	72.34	74.21
S:Accuracy	69.06	69.06	69.06	69.06
DKD	71.23	71.74	71.97	70.91
δ	-0.26	-0.58	-0.37	-3.30
LDRLD	71.45	72.00	72.20	74.16
δ	-0.04	-0.32	-0.14	-0.05

Table 19. The top-1 performance of students with different teachers on the CIFAR-100 datasets. δ represents the gap with the teacher's performance.

4.4. Combination with other KD

Table 20 shows LDRLD is also compatible with other KD, with slight gains possibly due to redundant non-target knowledge transfer.

Teacher Student	ResNet32×4 ResNet8×4	WRN-40-2 WRN-40-1	ResNet32×4 ShuffleNetV2	VGG13 VGG8
LDRLD	77.20	74.98	77.33	75.06
LDRLD+DKD	77.42 (+0.22)	75.25 (+0.27)	77.48 (+0.15)	75.17 (+0.11)
LDRLD+IKD [64]	77.52(+0.32)	75.15 (+0.17)	77.62 (+0.29)	75.12 (+0.06)

Table 20. Combining LDRLD with other KD on CIFAR100.

Manner	R-101 & R-18			R-101 & R-50		
Metrics	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Teacher	42.04	62.48	45.88	42.04	62.48	45.88
Student	33.26	53.61	35.26	37.93	58.84	41.05
FitNet[51]	34.13	54.16	36.71	38.76	59.62	41.80
FGFI [65]	35.44	55.51	38.17	39.44	60.27	43.04
ReviewKD[3]	36.75	56.72	34.00	40.36	60.97	44.08
KD[18]	33.97	54.66	36.62	38.35	59.41	41.71
CTKD [32]	34.56	55.43	36.91	-	-	-
DKD [84]	34.88	56.16	37.08	39.01	60.41	42.33
LDRLD	**35.12**	**56.79**	**37.57**	**39.31**	**61.06**	**42.56**

Table 21. Results on the MS-COCO using Faster-RCNN [50]-FPN [36], with AP evaluated on val2017 dataset.

Teacher Accuracy	ResNet32x4 66.17	ResNet32x4 66.17	VGG13 70.19	VGG13 70.19	ResNet50 60.01
Student Accuracy	MobileNetV2 40.23	ShuffleNetV1 37.28	MobileNetV2 40.23	VGG8 46.32	ShuffleNetV1 37.28
SP	48.49	61.83	44.28	54.78	55.31
CRD	57.45	62.28	56.45	66.10	57.45
SemCKD	56.89	63.78	68.23	66.54	57.20
ReviewKD	-	64.12	58.66	67.10	-
MGD	-	-	-	66.89	57.12
KD	56.09	61.68	53.98	64.18	57.21
DKD	59.94(+3.85)	64.51(+2.83)	58.45(+4.47)	67.20(+3.02)	59.21(+2.00)
NKD	59.81(+3.72)	64.00(+2.32)	58.40(+3.42)	67.16(+2.98)	59.11(+1.90)
LDRLD	**60.99**(+4.90)	**65.19**(+3.51)	**59.73**(+5.75)	**68.27**(+4.09)	**60.46**(+3.25)

Table 22. Top-1 accuracy (%) of student on the CUB200 dataset.

5. Comparison with State-of-the-Art Methods

5.1. Object Detection Results on MS-COCO2017

We apply the LDRLD method to the object detection task and validate it on two different architectures: the teacher-student pairs are ResNet-101 (R-101) and ResNet-18 (R-18), as well as ResNet-101 and ResNet-50 (R-50). The experimental results show that our method outperforms DKD on the MS-COCO2017 dataset, demonstrating its good generalization ability.

5.2. Fine-grained Task Results on the CUB200

The fine-grained task hinges on local features (e.g., feather patterns). LDRLD enhances inter-class separability by decoupling logits into combination pairs $[z_1, z_2]$, amplifying subtle differences while normalizing them to reduce interference from other categories. The results in Table 22 show that LDRLD outperforms other KD and validates generalization ability (see the SDD training detail).

5.3. Fine-grained Task Results on the Tiny-ImageNet.

To further validate the effectiveness of our method, we conducted experiments on Tiny-ImageNet, a fine-grained dataset with numerous categories and higher intra-class similarity. These characteristics allow the teacher to ef-

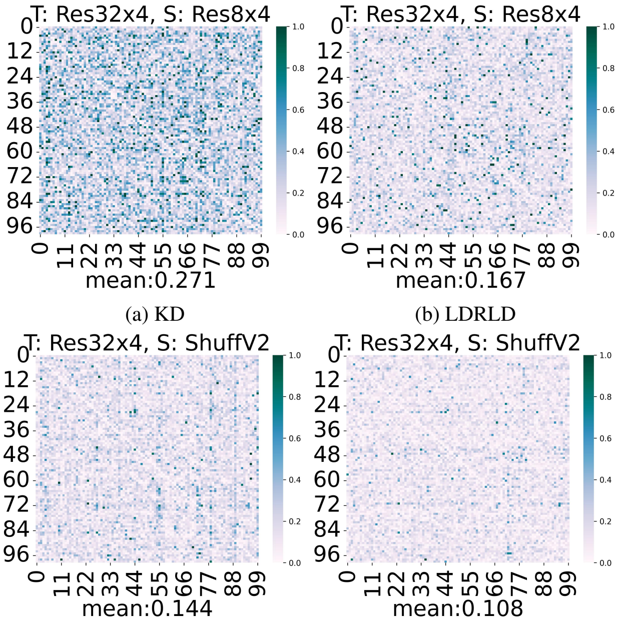
Distillation	Teacher	ResNet56	ResNet110	VGG13	WRN-40-2	VGG13
	Accuracy	56.56	59.01	60.20	60.45	60.20
Manner	Student	ResNet20	ResNet20	VGG8	WRN-16-2	MobileNetV2
	Accuracy	52.66	51.89	56.03	57.17	57.73
Features	AT [27]	54.39	54.57	58.85	59.39	60.84
	SP [63]	54.23	54.38	58.78	57.63	61.90
	CC [46]	54.22	54.26	58.18	58.83	61.32
	VID [1]	53.89	53.94	58.55	58.78	60.84
	PKT [45]	54.29	54.70	58.87	59.19	61.90
	FT [25]	53.90	54.46	58.87	58.85	61.78
	NST [23]	53.66	53.82	58.85	59.07	60.59
	FitNet [51]	54.43	54.04	58.33	58.88	61.37
	RKD [44]	53.95	53.88	58.58	59.31	61.19
	CRD [62]	55.04	54.69	58.88	59.42	61.63
	SAKD [58]	**55.06**	**55.28**	**59.53**	**59.87**	**62.29**
Logits	KD [18]	53.04	53.40	57.33	59.16	60.02
	DKD [84]	54.51	54.89	60.22	59.45	59.00
	WTTM [87]	55.07*	54.39*	**61.30***	59.95*	60.06*
	TeKAP [19]	54.83*	54.58*	60.37*	59.66*	59.08*
	LDRLD	**55.23**	**55.24**	60.91	**60.67**	**60.31**
	Δ	**+2.19**	**+1.84**	**+3.58**	**+1.51**	**+0.29**

Table 23. Evaluate the top-1 accuracy (%) of students on the Tiny-ImageNet validation set.

fectively transfer fine-grained logit knowledge through LDRLD, thereby enhancing the student's performance. As shown in Table 23, our method outperforms most existing classical KD approaches, achieving performance improvements ranging from 0.29% to 3.58% compared to KD. This achievement is primarily due to LDRLD enhancing the student's ability to capture detailed inter-class knowledge, which significantly boosts the student's discriminability in classification.

6. Visualization.

Visualization of Correlation Matrices. By visualizing the difference in correlation matrices between the teacher and the student, we observe that LDRLD produces a smaller difference than vanilla KD, as shown in Figs. 5 (b) and (d) compared to Figs. 5 (a) and (c). This finding indicates that the student's logits are closer to those of the teacher, confirming that our method facilitates more efficient knowledge acquisition. Consequently, it can be concluded that the enhanced knowledge transfer we propose can significantly improve the student's performance.



(a) KD

(b) LDRLD

(c) KD

(d) LDRLD

Figure 5. Visualization of the difference in correlation matrices between student and teacher logits for different teacher-student pairs: ResNet32×4 vs ResNet8×4, and ResNet32×4 vs ShuffleNetV2, on the CIFAR-100 dataset.