Relational Representation Distillation

Supplementary Material

Contents

6. Algorithm

We present the pseudocode of our method in Algorithm 1.

7. Implementation Details

We implement RRD in PyTorch following the protocol of [36][1]. This protocol has become a common standard and is widely used among many papers to demonstrate their knowledge distillation methods [1, 13, 14, 16, 17, 28–30, 32, 36, 37, 42, 43]. This

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

Algorithm 1 Pseudocode in a PyTorch-like style.

```
# f_s, f_t: student and teacher networks
# queue: memory of K features (CxK)
# t_s, t_t: temperature for student and teacher

for x in loader: # load a minibatch x with N samples
   s = f_s.forward(x) # student embeddings: NxC
   s = normalize(s, dim=1) # L2 normalization

   with torch.no_grad(): # no gradients
      t = f_t.forward(x) # teacher embeddings: NxC
      t = normalize(t, dim=1) # L2 normalization

   # enqueue the current minibatch
   enqueue(queue, t)

   # student similarities
   out_s = mm(s.view(N, C), queue.view(C, K))

   # teacher similarities
   out_t = mm(t.view(N, C), queue.view(C, K))

   # relational loss using softmax and log-softmax
   loss = -sum(
      softmax(out_t / t_t, dim=1) *
      log_softmax(out_s / t_s, dim=1), dim=1
   ).mean()

   # SGD update: student network only
   loss.backward()
   update(f_s.params)

   # dequeue the earliest minibatch
   dequeue(queue)
```

`mm`: matrix multiplication; `softmax`: exponential normalization across dimensions; `log_softmax`: logarithm of softmax; `normalize`: ℓ_2 normalization.

approach is consistent with the standard practice in the field, allowing for fair comparison across methods. For visualization, we follow the implementation of [35][2].

7.1. Baseline Methods

We compare our approach to the following state-of-the-art methods from the literature: (1) Knowledge Distillation (KD) [14]; (2) FitNets: Hints for Thin Deep Nets [32]; (3) Attention Transfer (AT) [43]; (4) Similarity-Preserving Knowledge Distillation (SP) [37]; (5) Correlation Congruence (CC) [30]; (6) Variational Information Distillation for Knowledge Transfer (VID) [1]; (7) Relational Knowledge Distillation (RKD) [28]; (8) Learning Deep Representations with Probabilistic Knowledge Transfer (PKT) [29]; (9) Knowledge Transfer via Distillation of Activation Boundaries Formed by Hidden Neurons (AB) [13]; (10) Paraphrasing Complex Network: Network Compression via Factor Transfer (FT) [17]; (11) A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning (FSP) [42]; (12) Like What You Like: Knowledge Distill via Neuron Selectivity Transfer (NST) [16]; (13) Contrastive Representation Distillation (CRD) [36]; (14) A Comprehensive Overhaul of Feature Distillation (OFD) [12]; (15) Rethinking Soft Labels for Knowledge Distillation: A Bias-Variance Tradeoff Perspective (WSLD) [48]; (16) Respecting Transfer Gap in Knowledge Distillation (IPWD) [27]; (17) Knowledge Distillation via Softmax Regression Representation Learning (SRRL) [41]; (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) [8]; (22) Knowledge Distillation via N-to-One Representation Matching (NORM) [23]; (23) Wasserstein Contrastive Representation Distillation (WCoRD) [4]; (24) Complementary Relation Contrastive Distillation (CRCD) [49]; (25) Feature Kernel Distillation (FKD) [10]; (26) Information Theoretic Representation (ITRD) [26]; (27) Test-Time Memory Distillation (TTM) [47]; (28) Weighted Test-Time Memory Distillation (WTTM) [47]; (29) Decoupled Knowledge Distillation (DKD) [46]; (30) Feature Correlation-based Feature Distillation (FCFD) [22]; (31) Category-aligned Knowledge Distillation (CAT-KD) [9]; (32) Distillation with Instance-wise Selective Teaching (DIST) [15]; (33) Cross-layer Transfer Knowledge Distillation (CTKD) [19]; (34) Relational Representation Distillation (RRD) (ours).

[2]Available at: https://github.com/sunshangquan/logit-standardization-KD.

7.2. Datasets

We take five widely researched datasets: (1) CIFAR-100 [18] is a standard benchmark for knowledge distillation and contains 50,000 training images of size 32×32 with 500 images per class and 10,000 test images. (2) ImageNet ILSVRC-2012 [7], which is more challenging that CIFAR, and includes 1.2 million images from 1,000 classes for training and 50,000 for validation. (3) STL-10 [6] consists of a training set of 5,000 labeled images from 10 classes, and a test set of 8,000 images. (4) Tiny ImageNet [7] has 200 classes, each with 500 training images and 50 validation images. (5) MS-COCO [20] is an 80-category general object detection dataset. The `train2017` split contains 118,000 images, and the `val2017` split contains 5,000 images.

7.3. Network Architectures

We use the following network architectures as described in [36]: (1) Wide Residual Network (WRN) [44], where WRN-d-w represents a wide ResNet with depth d and width factor w; (2) ResNet [11], 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 [11], where ResNet-d represents an ImageNet-style ResNet with Bottleneck blocks and more channels; (4) MobileNet-v2 [33], using a width multiplier of 0.5 in our experiments; (5) VGG [34], where the VGG network used is adapted from its original ImageNet counterpart; and (6) ShuffleNet-v1 [45] and ShuffleNet-v2 [24], which are adapted for efficient training with input sizes of 32×32. To ensure dimensional consistency and preserve relational information during feature alignment, both teacher and student features are passed through lightweight *projection heads*—two-layer MLPs (512 hidden, 128 output)—that nonlinearly project intermediate representations into a shared embedding space for computing relational similarity distributions, while being discarded after training to avoid any additional inference cost. This projection head facilitates knowledge transfer by implicitly encoding relational information from previous samples [25].

7.4. Optimization

We closely follow the protocol of [36]. 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[3], 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 five trials, except for the results on ImageNet, which are reported in a single trial. Our implementation for MS-COCO follows the settings in [46]. We use the two-stage method Faster R-CNN [31] with Feature Pyramid Network (FPN) [21] as the detection framework. We evaluate three teacher–student settings: ResNet-101 to ResNet-18, ResNet-101 to ResNet-50, and ResNet-50 to MobileNet-V2 [33]. All students are trained with the $1\times$ scheduler (schedulers and task-specific loss weights follow Detectron2 [40]). We take one RTX 6000 GPU to train the model on CIFAR-100 and four L40 GPUs on ImageNet and MS-COCO.

8. Theoretical Analysis

We propose a learning objective with dual aims: aligning the student model's encoding $\mathbf{z}_i^S$ with the teacher model's $\mathbf{z}_i^T$, while enabling $\mathbf{z}_i^S$ to softly contrast with memory-stored random samples $\mathcal{M}$. This can be expressed by minimizing both the ℓ_2 distance between $\mathbf{z}_i^T$ and $\mathbf{z}_i^S$, and the cross-entropy using $\mathcal{M}$:

$$\begin{aligned}\mathcal{L} &= \frac{1}{N}\sum_{i}^{N}\left\{\phi \cdot |\mathbf{z}_i^T - \mathbf{z}_i^S|_2 - \xi \cdot \mathbf{p}^T(\mathbf{x}_i;\theta_T,\mathcal{M}) \cdot \log \mathbf{p}^S(\mathbf{x}_i;\theta_S,\mathcal{M})\right\} \\ &= \sum_{i}^{N}\left\{ -\phi \cdot \mathbf{z}_i^T \cdot \mathbf{z}_i^S - \xi \cdot \sum_{j}^{K} \frac{\exp(\mathbf{z}_i^T \cdot \mathbf{m}_j/\tau_t)}{\sum\limits_{\mathbf{m}\sim\mathcal{M}} \exp(\mathbf{z}_i^T \cdot \mathbf{m}/\tau_t)} \cdot \log \frac{\exp(\mathbf{z}_i^S \cdot \mathbf{m}_j/\tau_s)}{\sum\limits_{\mathbf{m}\sim\mathcal{M}} \exp(\mathbf{z}_i^S \cdot \mathbf{m}/\tau_s)}\right\}.\end{aligned} \tag{13}$$

Direct optimization of Equation (13) presents challenges in finding optimal hyperparameters (ϕ, ξ, τ_t and τ_s). Our objective using $\mathcal{M}^+$ serves as a simplified approximated upper-bound:

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

$$\begin{aligned}\mathcal{L}_{\text{RRD}} &= \frac{1}{N}\sum_{i}^{N} -\mathbf{p}^T(\mathbf{x}_i;\theta_T,\mathcal{M}^+)\cdot\log\mathbf{p}^S(\mathbf{x}_i;\theta_S,\mathcal{M}^+) \\ &= \sum_{i}^{N}\sum_{j}^{K+1} -\underbrace{\frac{\exp(\mathbf{z}_i^T\cdot\mathbf{m}_j/\tau_t)}{\sum_{\mathbf{m}\sim\mathcal{M}^+}\exp(\mathbf{z}_i^T\cdot\mathbf{m}/\tau_t)}}_{\mathbf{a}_j^i}\cdot\log\frac{\exp(\mathbf{z}_i^S\cdot\mathbf{m}_j/\tau_s)}{\sum_{\mathbf{m}\sim\mathcal{M}^+}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau_s)}.\end{aligned} \tag{14}$$

Here, α_j^i represents the temperature-controlled weighting term under τ_t. Since $\mathbf{z}_i^T$ is added as the $(K+1)$-th element in $\mathcal{M}^+$, we can expand the objective as:

$$\begin{aligned}\mathcal{L}_{\text{RRD}} =& \frac{1}{N}\sum_{i}^{N}\Big\{\mathbf{a}_{K+1}^i\cdot\big(-\mathbf{z}_i^S\cdot\mathbf{z}_i^T/\tau_s+\log\sum_{\mathbf{m}\sim\mathcal{M}^+}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau_s)\big) \\ &+\sum_{j=1}^{K}\mathbf{a}_j^i\cdot\big(-\mathbf{z}_i^S\cdot\mathbf{m}_j/\tau_s+\log\sum_{\mathbf{m}\sim\mathcal{M}^+}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau_s)\big)\Big\}.\end{aligned} \tag{15}$$

The Log-Sum-Exp (LSE) function is defined as $\text{LSE}(\mathcal{M}^+,\mathbf{z}_i^S)=\log\sum_{m\sim\mathcal{M}^+}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau_s)$. The LSE term in the first line maintains non-negativity since inner products between $\mathbf{z}^S$ and $\mathcal{M}$ fall within $[-1,+1]$:

$$\text{LSE}(\mathcal{M}^+,\mathbf{z}_i^S)\geq\log\big(M\cdot\exp(-1/\tau_s)\big)=\log\big(M\cdot\exp(-10)\big)>0, \tag{16}$$

where M represents the cardinality of the queue $\mathcal{M}^+$ with fixed $\tau_s=0.1$.
The second line's LSE term follows:

$$\text{LSE}(\mathcal{M}^+,\mathbf{z}_i^S)\geq\text{LSE}(\mathcal{M},\mathbf{z}_i^S). \tag{17}$$

This demonstrates that Equation (14) minimizes a relaxed upper-bound of Equation (13):

$$\begin{aligned}\mathcal{L}_{\text{RRD}} &= \frac{1}{N}\sum_{i}^{N} -\mathbf{p}^T(\mathbf{x}_i;\theta_T,\mathcal{M}^+)\cdot\log\mathbf{p}^S(\mathbf{x}_i;\theta_S,\mathcal{M}^+) \\ &\geq \frac{1}{N}\sum_{i}^{N}\Big\{\mathbf{a}_{K+1}^i\cdot(-\mathbf{z}_i^S\cdot\mathbf{z}_i^T/\tau_s)+\sum_{j=1}^{K}\mathbf{a}_j^i\cdot\big(-\mathbf{z}_i^S\cdot\mathbf{m}_j/\tau_s+\log\sum_{\mathbf{m}\sim\mathcal{M}}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau_s)\big)\Big\} \\ &= \frac{1}{N}\sum_{i}^{N}\Big\{-\frac{\mathbf{a}_{K+1}^i}{\tau_s}\cdot\mathbf{z}_i^S\cdot\mathbf{z}_i^T-\mathbf{p}^T(\mathbf{x}_i;\theta_T,\mathcal{M})\cdot\log\mathbf{p}^S(\mathbf{x}_i;\theta_S,\mathcal{M})\Big\}.\end{aligned} \tag{18}$$

This establishes that $\mathcal{L}_{\text{RRD}}$ connects to the intuitive distillation form in Equation (13) (ℓ_2 + cross entropy loss), inherently incorporating alignment and contrast objectives. Our approach simplifies the training process, and empirical evidence shows that both loss formulations yield comparable results when properly tuning τ_t.

8.1. Relation to InfoNCE

Our distillation approach extends InfoNCE [38] with probabilistic negative sample handling. InfoNCE treats negatives equally, while we apply probability-based weights. The standard InfoNCE formulation consists of pulling teacher encodings close while pushing away from random samples:

$$\hat{\theta}_S=\arg\min_{\theta_S}\mathcal{L}_{NCE}=\arg\min_{\theta_S}\sum_{i}^{N}-\log\frac{\exp(\mathbf{z}_i^T\cdot\mathbf{z}_i^S/\tau)}{\sum_{m\sim\mathcal{M}}\exp(\mathbf{z}_i^S\cdot\mathbf{m}/\tau)}. \tag{19}$$

This decomposes into *"pull"* and *"push"* forces - pull acts as positive sample alignment by drawing similar encodings together, while push provides contrast by moving away from negative instances:

$$\mathcal{L}_{NCE} = \sum_{i}^{N} \Big\{ \underbrace{-\mathbf{z}_i^S \cdot \mathbf{z}_i^T / \tau}_{\text{pull}} + \underbrace{\log \sum_{m \sim \mathcal{M}} \exp(\mathbf{z}_i^S \cdot \mathbf{m}/\tau)}_{\text{push}} \Big\}. \tag{20}$$

Similarly, our objective follows this structure with added weighting:

$$\begin{aligned} \mathcal{L}_{\text{RRD}} &= \frac{1}{N \cdot M} \sum_{i}^{N} \sum_{j}^{K+1} - \frac{\exp(\mathbf{z}_i^T \cdot \mathbf{m}_j / \tau_t)}{\sum_{\mathbf{m} \sim \mathcal{M}^+} \exp(\mathbf{z}_i^T \cdot \mathbf{m}/\tau_t)} \cdot \log \frac{\exp(\mathbf{z}_i^S \cdot \mathbf{m}_j / \tau_s)}{\sum_{\mathbf{m} \sim \mathcal{M}^+} \exp(\mathbf{z}_i^S \cdot \mathbf{m})/\tau_s} \\ &= \frac{1}{N \cdot M} \sum_{i}^{N} \sum_{j}^{K+1} \underbrace{\frac{\exp(\mathbf{z}_i^T \cdot \mathbf{m}_j / \tau_t)}{\sum_{\mathbf{m} \sim \mathcal{M}^+} \exp(\mathbf{z}_i^T \cdot \mathbf{m}/\tau_t)}}_{\mathbf{a}_j^i} \cdot (\underbrace{-\mathbf{z}_i^S \cdot \mathbf{z}_i^T / \tau_s}_{\text{pull}} + \underbrace{\log \sum_{\mathbf{m} \sim \mathcal{M}} \exp(\mathbf{z}_i^S \cdot \mathbf{m}/\tau^S)}_{\text{push}})). \end{aligned} \tag{21}$$

The soft labels $\mathbf{A}^i = [\mathbf{a}_1^i, ..., \mathbf{a}_{K+1}^i]$ give us:

$$\mathcal{L}_{\text{RRD}} = \frac{1}{N \cdot M} \sum_{i}^{N} \sum_{j}^{K+1} \mathbf{a}_j^i \cdot \Big\{ -\mathbf{z}_i^S \cdot \mathbf{z}_i^T / \tau_s + \log \sum_{m \sim \mathcal{M}} \exp(\mathbf{z}_i^S \cdot \mathbf{m}/\tau^S)) \Big\}. \tag{22}$$

At low temperature τ_t, $\mathbf{A}^i$ becomes one-hot ($\mathbf{a}_{K+1}^i = 1$), reducing our objective to standard InfoNCE pull-push dynamics (Equation (20)). However, higher τ_t values improve performance - soft learning shows better results on CIFAR and similar datasets [18].

9. Extended Ablation Results

Here we present comprehensive tabular results for all ablation studies discussed in the main paper. The following tables provide detailed accuracy values (mean ± std over three or five runs) for each experimental configuration, supporting and expanding upon the analysis presented in the main text. Section 9.1 examines the critical role of memory bank configurations and update strategies in our framework. Section 9.2 provides detailed ablations on temperature parameters (τ_t, τ_s), memory size (K), and loss coefficients (β, λ). Section 9.3 presents a comprehensive comparison between our RRD method and CRD under various configurations. Finally, Section 9.4 offers complete results for different loss coefficient combinations (β, λ) across all teacher-student pairs.

9.1. Memory Structure

This section investigates the fundamental role of the memory bank in enabling effective relational knowledge distillation through three key aspects. We first analyze why the memory bank is essential for computing rich similarity distributions (Section 9.1.1), then compare different update strategies including FIFO and momentum-based approaches (Section 9.1.2), and finally present empirical results demonstrating the superiority of FIFO memory management (Section 9.1.3).

9.1.1. Role and Importance of the Memory Bank

The memory bank $\mathcal{M}$ serves as a fundamental component in our relational representation distillation framework, enabling the computation of rich similarity distributions that capture relational structures between feature representations. Without this stored repository of reference features, similarity computations would be severely constrained to the current batch, limiting the model's capacity to learn comprehensive relational knowledge. The memory bank maintains features from previous iterations, allowing the model to contrast current representations against a broader, more diverse set of stored embeddings. This extended comparison space is crucial for several reasons: (1) it effectively simulates large-batch training without the associated computational overhead, (2) it provides temporal consistency by preserving historical feature representations, and (3) it enables the model to capture long-range dependencies and relationships that span beyond immediate batch boundaries. From a theoretical perspective, the memory bank extends the support of the similarity distribution $\mathbf{p}^T$, allowing the KL divergence to model higher-order relationships among instances. This prevents overfitting to batch-local patterns and ensures that the learned representations encode global structural information. Without stored references, the diversity of negative samples would be drastically reduced, potentially leading to representation collapse and poor generalization.

Table 7. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. We ablate different memory bank configurations against CRD [36], which we reproduce. ↑ denotes outperformance over CRD and ↓ denotes underperformance. Each experiment is run five times.

Method	CRD [36]	RRD	CRD [36]	RRD	CRD+KD [36]	RRD+KD	CRD+KD [36]	RRD+KD
Momentum	✗	✗	✓	✓	✗	✗	✓	✓
Top-1	74.46	75.44 (↑)	73.55	73.56 (↑)	75.40	75.84 (↑)	75.21	75.45 (↑)

9.1.2. Memory Update Strategies

The memory bank $\mathcal{M}$ can be maintained using two distinct update strategies:

1. First-In-First-Out (FIFO): In this approach, we add the teacher's features from the current batch while removing the oldest stored features at each iteration. This ensures continuous refreshing of the memory bank with the latest feature embeddings, maintaining an up-to-date representation of the teacher's evolving feature space. The FIFO mechanism preserves the instantaneous relational geometry of the teacher's feature space, providing sharp and adaptive relational targets that accurately reflect the current teacher distribution.
2. Momentum update: Instead of replacing old features entirely, stored features in $\mathcal{M}$ are updated gradually using an exponential moving average (EMA):

$$\mathbf{m}_k \leftarrow \alpha \cdot \mathbf{m}_k + (1-\alpha) \cdot \mathbf{z}_i^T, \tag{23}$$

where $\alpha \in [0, 1)$ is a momentum coefficient controlling the decay rate of historical information. This approach maintains smoothed representations of past embeddings, ensuring temporal consistency across training iterations. The momentum-based memory approximates a moving integration over past teacher states:

$$\mathcal{M}_t = (1-\alpha) \sum_{j=0}^{\infty} \alpha^j f_{\theta_{t-j}}^T(\mathbf{x}), \tag{24}$$

which implicitly regularizes representation drift and reduces gradient variance across iterations.

Critically, regardless of the update strategy, appending the current teacher embedding as $\mathbf{m}_{K+1}$ ensures that the teacher's similarity distribution maintains a clear peak rather than becoming uniform over memory features. This prevents the KL divergence from degenerating into weak contrastive learning and maintains the discriminative power of the relational signal.

9.1.3. Ablation Results

We evaluated the impact of different memory bank configurations by comparing our method against CRD [36] using both update strategies: (1) a momentum queue with $\alpha = 0.999$, similar to [36], and (2) a non-momentum FIFO queue. As shown in Table 7, RRD demonstrates superior performance across both configurations, with particularly strong results using the FIFO queue. This indicates that continuously refreshing the teacher memory with up-to-date embeddings yields more accurate relational cues, despite the smoother dynamics provided by the EMA-based queue. The superior performance of the FIFO strategy can be attributed to its preservation of fine-grained relational distinctions. While the EMA memory produces smoother training dynamics by maintaining low-pass filtered versions of recent teacher features, it may blur subtle relational patterns as training progresses. The FIFO queue, conversely, maintains stronger alignment between teacher and student similarity distributions by preserving the current relational structure. We deliberately chose to use a shared teacher memory bank rather than separate banks for teacher and student. Separate memory banks would prevent direct feature alignment, as the KL divergence would operate between mismatched distributions from different feature spaces, breaking the one-to-one correspondence between teacher and student similarities. Similarly, storing student features in the memory would produce unstable targets due to the continuous evolution of student representations during training. The shared teacher memory bank provides stationary and semantically consistent targets, enabling the student to learn from a well-defined, high-quality relational prior while maintaining feature-space correspondence and training stability. The quality of the target similarity distribution $\mathbf{p}^T(\mathbf{x}_i; \theta_T, \mathcal{M}^+)$ ultimately depends on maintaining this large memory buffer to store feature embeddings from teacher batches, ensuring that structural relationships between teacher and student models are preserved through aligned similarity distributions optimized via KL divergence.

Table 8. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. We ablate temperature parameters (τ_t, τ_s) while fixing memory size $K = 16384$, $\beta = 1$, and $\lambda = 0$. Results are averaged over three runs.

τ_t	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.1	0.2	1.0
					$\tau_s = 0.1$					
RRD (ours)	75.37 (±0.14)	75.99 (±0.30)	75.48 (±0.17)	75.47 (±0.27)	75.40 (±0.10)	75.37 (±0.06)	75.10 (±0.32)	74.64 (±0.26)	73.80 (±0.18)	73.32 (±0.11)
					$\tau_s = 0.2$					
RRD (ours)	75.33 (±0.33)	75.57 (±0.06)	75.24 (±0.08)	74.95 (±0.12)	74.93 (±0.08)	74.64 (±0.36)	74.50 (±0.23)	74.31 (±0.31)	73.89 (±0.06)	73.62 (±0.27)
					$\tau_s = 1.0$					
RRD (ours)	73.87 (±0.25)	73.95 (±0.22)	73.91 (±0.23)	73.84 (±0.25)	73.76 (±0.23)	73.62 (±0.27)	73.48 (±0.29)	73.21 (±0.32)	72.84 (±0.34)	72.43 (±0.38)

Table 9. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. We ablate memory size K while fixing $\tau_t = 0.02$, $\tau_s = 0.1$, $\beta = 1$, and $\lambda = 0$. Results are averaged over three runs.

K	64	256	1024	2048	4096	8192	16384	32768	65536
CRD [36]	75.10 (±0.10)	75.36 (±0.02)	75.41 (±0.16)	75.51 (±0.12)	75.56 (±0.25)	75.61 (±0.22)	75.73 (±0.11)	75.85 (±0.32)	75.82 (±0.32)
RRD (ours)	75.15 (±0.45)	75.19 (±0.14)	75.59 (±0.09)	75.64 (±0.35)	75.73 (±0.11)	75.80 (±0.14)	75.99 (±0.30)	76.00 (±0.18)	75.96 (±0.06)

Table 10. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. We ablate loss coefficient β while fixing $K = 16384$, $\tau_t = 0.02$, $\tau_s = 0.1$, and $\lambda = 0$. Results are averaged over three runs.

β	0	0.1	0.3	0.5	0.7	0.9	1.0	1.1	1.3	1.5	2.0	5.0	10
RRD (ours)	73.54 (±0.52)	74.06 (±0.21)	74.89 (±0.11)	74.92 (±0.52)	75.60 (±0.18)	75.67 (±0.24)	75.99 (±0.30)	75.92 (±0.30)	75.91 (±0.33)	75.90 (±0.18)	75.51 (±0.28)	75.50 (±0.18)	75.32 (±0.26)

Table 11. Ablation study results on CIFAR-100 using WRN-40-2 as the teacher and WRN-16-2 as the student. We ablate loss coefficient λ while fixing $K = 16384$, $\tau_t = 0.02$, $\tau_s = 0.1$, and $\beta = 1$. Results are averaged over three runs.

λ	0	0.1	0.3	0.5	0.7	0.9	1.0	2.0	5.0	10
RRD (ours)	75.99 (±0.30)	75.45 (±0.09)	75.73 (±0.08)	75.73 (±0.19)	75.75 (±0.22)	75.81 (±0.24)	75.86 (±0.25)	75.59 (±0.11)	75.59 (±0.24)	75.04 (±0.11)

9.2. Temperature, Memory Size, and Loss Coefficients

To complement the analysis in the main paper, we present the detailed ablation results performed on CIFAR-100 using WRN-40-2 as teacher and WRN-16-2 as student, with all experiments averaged over three runs. First, we investigate temperature parameters by fixing τ_s to either 0.1, 0.2, or 1.0 and varying $\tau_t \in \{0.01, \ldots, 0.07\} \cup \{0.1, 0.2, 1.0\}$, using a FIFO memory of size $K = 16384$ with $\beta = 1$ and $\lambda = 0$. As shown in Table 8, we observe optimal performance when $\tau_t < \tau_s$, with significant performance degradation when $\tau_t > \tau_s$. Next, we study memory size by varying $K \in \{64, 256, 1024, 2048, 4096, 8192, 16384, 32768, 65536\}$ with fixed $\tau_t = 0.02$, $\tau_s = 0.1$, $\beta = 1$ and $\lambda = 0$. From Table 9, our method demonstrates robustness to memory size, maintaining strong performance even with smaller memory sizes. In contrast, CRD with $\tau = 0.07$ and $\lambda = 0$ requires a large memory buffer of $K = 16384$ to achieve competitive performance (Table 9), yet still underperforms our method. Finally, we evaluate loss coefficients β and λ with $K = 16384$, $\tau_t = 0.02$, $\tau_s = 0.1$, testing values $\{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.1, 1.3, 1.5, 2.0, 5.0, 10.0\}$. Results in Tables 10 and 11 show our method performs well with $\beta \in [0.5, 1.5]$ and $\lambda = 1.0$. We set default values of $\beta = 1$ when $\lambda = 0$ and $\beta = 1.5$ when $\lambda > 0$.

Table 12. Comparison of RRD vs CRD under different momentum structures. We show test top-1 accuracy (%) on CIFAR-100 of student networks trained with CRD and RRD for same architecture teacher-student pairs. Results are averaged over five runs. Δ represents the absolute difference between RRD and CRD in percentage points. $\uparrow$ denotes outperformance over CRD ($\Delta \geq 0$) and $\downarrow$ denotes underperformance ($\Delta < 0$).

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
			Momentum queue				
CRD	74.46 (±0.21)	73.17 (±0.36)	70.26 (±0.18)	70.31 (±0.19)	72.36 (±0.16)	74.46 (±0.15)	72.93 (±0.24)
RRD (ours)	75.44 (±0.23)	74.21 (±0.22)	71.70 (±0.09)	71.57 (±0.21)	73.57 (±0.26)	75.54 (±0.15)	73.81 (±0.12)
Δ	+0.98 ($\uparrow$)	+1.04 ($\uparrow$)	+1.44 ($\uparrow$)	+1.26 ($\uparrow$)	+1.21 ($\uparrow$)	+1.08 ($\uparrow$)	+0.88 ($\uparrow$)
			FIFO queue				
CRD	73.55 (±0.29)	71.14 (±0.35)	68.10 (±0.27)	68.05 (±0.28)	70.38 (±0.34)	73.47 (±0.16)	70.76 (±0.18)
RRD (ours)	73.56 (±0.32)	71.86 (±0.17)	69.82 (±0.26)	69.68 (±0.31)	71.55 (±0.25)	72.94 (±0.34)	70.75 (±0.15)
Δ	+0.01 ($\uparrow$)	+0.72 ($\uparrow$)	+1.72 ($\uparrow$)	+1.63 ($\uparrow$)	+1.17 ($\uparrow$)	-0.53 ($\downarrow$)	-0.01 ($\downarrow$)
			Momentum queue				
CRD+KD	75.40 (±0.24)	74.41 (±0.32)	71.48 (±0.18)	71.23 (±0.22)	73.58 (±0.27)	74.93 (±0.25)	73.90 (±0.29)
RRD+KD (ours)	75.84 (±0.34)	74.72 (±0.20)	71.83 (±0.27)	71.44 (±0.45)	73.59 (±0.15)	74.78 (±0.23)	74.08 (±0.13)
Δ	+0.44 ($\uparrow$)	+0.31 ($\uparrow$)	+0.35 ($\uparrow$)	+0.21 ($\uparrow$)	+0.01 ($\uparrow$)	-0.15 ($\downarrow$)	+0.18 ($\uparrow$)
			FIFO queue				
CRD+KD	75.21 (±0.33)	73.23 (±0.36)	71.33 (±0.33)	70.79 (±0.18)	73.22 (±0.24)	74.78 (±0.16)	72.82 (±0.24)
RRD+KD (ours)	75.45 (±0.16)	73.98 (±0.20)	71.39 (±0.15)	71.06 (±0.45)	73.54 (±0.25)	74.39 (±0.18)	73.45 (±0.24)
Δ	+0.24 ($\uparrow$)	+0.75 ($\uparrow$)	+0.06 ($\uparrow$)	+0.27 ($\uparrow$)	+0.32 ($\uparrow$)	-0.39 ($\downarrow$)	+0.63 ($\uparrow$)

9.3. Comparison of CRD and RRD

Next, we validate our relational objective against the contrastive objective of [36]. For a fair comparison, we evaluate both methods using three different implementations: (1) a FIFO memory bank, which stores and retrieves past samples in a first-in, first-out manner; (2) a momentum memory bank. For a fair comparison between RRD and CRD, we use $K = 16384$ and $\beta = 1$. When combined with KL divergence, we set $\lambda = 0.9$. For RRD, we use $\tau_t = 0.02$ and $\tau_s = 0.1$, while for CRD, we use $\tau = 0.07$. The results, averaged over five runs, are shown in Tables 12 and 13. We observe the following trends:

- RRD performs better with a FIFO queue compared to a momentum queue, achieving an average improvement of 2.81%. This suggests that RRD benefits more from retrieving recent samples rather than accumulated historical representations. When KD is added, the difference between FIFO and momentum queues decreases, with the improvement dropping to 0.76%. This indicates that KD helps mitigate the reliance on a specific memory structure.
- When both methods use a FIFO queue, RRD outperforms CRD by an average of 1.19%. This result shows that the relational objective provides a stronger training signal than the contrastive objective under the same memory setup. However, when KD is applied, the difference reduces to 0.17%, suggesting that KD enables CRD to recover some of the lost performance.
- Unlike FIFO, while CRD benefits from a momentum queue, RRD still maintains competitive in this configuration. When both methods use a momentum queue, RRD demonstrates its robustness by achieving a +0.13% improvement over CRD, showing that our relational approach remains effective even in memory structures that favor CRD's contrastive learning. This advantage is further reinforced when KD is introduced, where RRD extends its lead to +0.28%. These results highlight that RRD's relational learning framework provides consistent improvements across different memory mechanisms, making

Table 13. Comparison of RRD vs CRD under different momentum structures. We show test top-1 accuracy (%) on CIFAR-100 of student networks trained with CRD and RRD for different architecture teacher-student pairs. Results are averaged over five runs. Δ represents the absolute difference between RRD and CRD in percentage points. $\uparrow$ denotes outperformance over CRD ($\Delta \geq 0$) and $\downarrow$ denotes underperformance ($\Delta < 0$).

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
			Momentum queue			
CRD	67.44 (±0.44)	67.11 (±0.54)	72.55 (±0.27)	74.75 (±0.25)	75.91 (±0.31)	74.83 (±0.25)
RRD (ours)	69.34 (±0.38)	69.66 (±0.45)	74.15 (±0.25)	75.17 (±0.52)	76.08 (±0.45)	75.79 (±0.29)
Δ	+1.90 ($\uparrow$)	+2.55 ($\uparrow$)	+1.60 ($\uparrow$)	+0.42 ($\uparrow$)	+0.17 ($\uparrow$)	+0.96 ($\uparrow$)
			FIFO queue			
CRD	65.79 (±0.40)	66.04 (±0.26)	70.69 (±0.18)	72.88 (±0.13)	73.68 (±0.18)	73.13 (±0.20)
RRD (ours)	65.57 (±0.33)	65.47 (±0.32)	71.16 (±0.51)	71.95 (±0.16)	73.13 (±0.39)	71.94 (±0.30)
Δ	-0.22 ($\downarrow$)	-0.57 ($\downarrow$)	+0.47 ($\uparrow$)	-0.93 ($\downarrow$)	-0.55 ($\downarrow$)	-1.19 ($\downarrow$)
			FIFO queue			
CRD+KD	69.36 (±0.18)	69.38 (±0.80)	74.05 (±0.15)	75.51 (±0.23)	76.56 (±0.13)	76.30 (±0.28)
RRD+KD (ours)	69.75 (±0.34)	69.92 (±0.29)	74.20 (±0.27)	75.37 (±0.29)	76.51 (±0.33)	76.30 (±0.44)
Δ	+0.39 ($\uparrow$)	+0.54 ($\uparrow$)	+0.15 ($\uparrow$)	-0.14 ($\downarrow$)	-0.05 ($\downarrow$)	0.00 ($\uparrow$)
			Momentum queue			
CRD+KD	67.86 (±0.16)	68.26 (±0.41)	73.03 (±0.24)	74.16 (±0.11)	74.92 (±0.23)	75.01 (±0.20)
RRD+KD (ours)	68.16 (±0.25)	68.14 (±0.80)	73.34 (±0.27)	74.29 (±0.10)	75.57 (±0.25)	75.57 (±0.33)
Δ	+0.30 ($\uparrow$)	-0.12 ($\downarrow$)	+0.31 ($\uparrow$)	+0.13 ($\uparrow$)	+0.65 ($\uparrow$)	+0.56 ($\uparrow$)

it a more versatile and reliable choice for knowledge distillation tasks.

- When using the best configuration for each method (CRD with a momentum queue and RRD with a FIFO queue), standalone RRD achieves a 2.95% improvement over CRD. This suggests that RRD benefits significantly from direct retrieval in a FIFO queue, whereas CRD performs better when leveraging past representations in a momentum queue. When KD is applied, the difference is reduced to 1.05%, showing that while KD benefits both methods, it does not entirely remove the advantage of RRD in this setup.

9.4. Full Ablations of RRD on Loss Coefficients

Tables 14 and 15 presents a comprehensive evaluation of different loss coefficient configurations. Using $\beta = 1$ establishes a strong baseline, particularly excelling in same-architecture scenarios. Increasing β to 1.5 shows improved performance in cross-architecture transfer scenarios. When combining with KL divergence [14], both $\lambda = 0.9$ and $\lambda = 1.0$ configurations with $\beta = 1$ perform similarly. However, the combination of $\beta = 1.5$ and $\lambda = 0.9$ demonstrates superior performance on cross-architecture scenarios, achieving the best results on several challenging pairs.

10. Extended Results

This section presents additional experiments complementing the main results. We report *extended* comparisons on ImageNet (Section 10.1) and CIFAR-100 (Section 10.2), transferability (Section 10.3), correlation alignment (Section 10.4) and semantic preservation (Section 10.5).

Table 14. Comparison of different RRD configurations. We show test top-1 accuracy (%) on CIFAR-100 of student networks trained with various distillation methods for same architecture teacher-student pairs. Results adapted from [36]. We report mean accuracy, standard deviation (std), and maximum accuracy (max) over five runs.

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
KD [14]	74.92	73.54	70.66	70.67	73.08	73.33	72.98
CRD [36]	75.48	74.14	71.16	71.46	73.48	75.51	73.94
CRD+KD [36]	75.64	74.38	71.63	71.56	73.75	75.46	74.29
			$\beta = 1$				
RRD (ours)	75.85	74.61	71.72	71.92	73.73	75.68	73.99
std	(±0.15)	(±0.21)	(±0.34)	(±0.08)	(±0.13)	(±0.18)	(±0.13)
max	76.10	74.96	72.41	72.03	73.89	76.10	74.18
			$\beta = 1.5$				
RRD (ours)	75.74	74.47	71.48	71.60	73.56	75.97	73.94
std	(±0.17)	(±0.35)	(±0.35)	(±0.26)	(±0.12)	(±0.15)	(±0.21)
max	75.99	74.95	72.07	71.90	73.66	76.15	74.30
			$\beta = 1, \lambda = 0.9$				
RRD+KD (ours)	75.60	74.70	71.73	71.37	73.80	75.25	73.99
std	(±0.19)	(±0.20)	(±0.16)	(±0.24)	(±0.28)	(±0.19)	(±0.28)
max	75.79	74.89	71.97	71.68	74.19	75.46	74.32
			$\beta = 1, \lambda = 1$				
RRD+KD (ours)	75.53	74.51	71.57	71.34	73.76	75.43	73.89
std	(±0.13)	(±0.14)	(±0.30)	(±0.13)	(±0.11)	(±0.17)	(±0.15)
max	75.70	74.63	71.91	71.54	73.87	75.66	74.06
			$\beta = 1.5, \lambda = 0.9$				
RRD+KD (ours)	75.67	74.68	72.03	71.75	73.96	75.53	74.37
std	(±0.36)	(±0.22)	(±0.31)	(±0.18)	(±0.09)	(±0.29)	(±0.38)
max	76.22	74.83	71.98	71.83	74.09	75.72	74.67

10.1. Results on ImageNet

Table 16 reports the top-1 accuracy (%) of various distillation methods on the ImageNet validation set across multiple teacher–student pairs. The results show that RRD achieves consistently strong performance, surpassing most relational and logit-based baselines such as CRD and KD.

10.2. Results on CIFAR-100

For CIFAR-100, we evaluate a broad set of distillation approaches on both same-architecture and cross-architecture teacher–student settings. Tables 20 and 21 list the top-1 accuracies (%) with respect to the KD baseline, using arrows to denote relative improvements or degradations. RRD performs competitively in both setups and provides consistent gains when combined with KD.

10.3. Results on Transferability

To assess the transferability of learned representations, we evaluate students distilled on CIFAR-100 when transferred to two downstream datasets: STL-10 and Tiny ImageNet. Using WRN-40-2 as the teacher and WRN-16-2 as the student, all convolutional layers are frozen, and only the final classification layer is retrained with cross-entropy loss for 120 epochs using SGD with momentum. This protocol isolates the representational quality of pretrained features without further adaptation. Table 17 reports top-1 accuracies, showing that RRD yields strong feature generalization, outperforming CRD on STL-10 and maintaining comparable results on Tiny ImageNet. These results indicate that RRD transfers effectively even without explicit KD supervision.

Table 15. Comparison of different RRD configurations. We show test top-1 accuracy (%) on CIFAR-100 of student networks trained with various distillation methods for different architecture teacher-student pairs. Results adapted from [36]. We report mean accuracy, standard deviation (std), and maximum accuracy (max) over five runs.

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
KD [14]	67.37	67.35	73.81	74.07	74.45	74.83
CRD [36]	69.73	69.11	74.30	75.11	75.65	76.05
CRD+KD [36]	69.94	69.54	74.97	75.12	76.05	76.27
			$\beta = 1$			
RRD (ours)	69.59	70.03	74.29	75.60	76.25	75.98
std	(±0.24)	(±0.22)	(±0.06)	(±0.24)	(±0.13)	(±0.07)
max	69.92	70.41	74.39	75.85	76.43	76.10
			$\beta = 1.5$			
RRD (ours)	69.78	70.34	74.61	75.68	76.48	75.95
std	(±0.43)	(±0.41)	(±0.31)	(±0.16)	(±0.22)	(±0.21)
max	70.43	70.90	75.11	75.87	76.70	76.17
			$\beta = 1, \lambda = 0.9$			
RRD+KD (ours)	69.80	69.73	74.18	75.49	76.31	76.39
std	(±0.26)	(±0.14)	(±0.10)	(±0.50)	(±0.23)	(±0.36)
max	70.25	69.90	74.30	75.93	76.70	76.65
			$\beta = 1, \lambda = 1$			
RRD+KD (ours)	69.80	69.44	74.32	75.57	76.44	76.44
std	(±0.20)	(±0.87)	(±0.22)	(±0.30)	(±0.16)	(±0.10)
max	70.08	70.22	74.55	75.82	76.58	76.56
			$\beta = 1.5, \lambda = 0.9$			
RRD+KD (ours)	69.99	69.65	74.53	76.68	76.87	76.64
std	(±0.85)	(±1.08)	(±0.12)	(±0.25)	(±0.08)	(±0.29)
max	70.16	70.62	74.66	76.04	76.95	76.97

Table 16. Test top-1 accuracy (%) on ImageNet validation set for student networks trained with various distillation methods across different teacher-student architectures. Results for our method are based on a *single* run. The abbreviations are listed in Section 7.1.

Teacher Student	ResNet-34 ResNet-18	ResNet-50 ResNet-18	ResNet-50 MobileNet-v2
Teacher	73.31	76.16	76.16
Student	69.75	69.75	69.63
KD [14]	70.67	71.29	70.49
AT [43]	71.03	71.18	70.18
SP [37]	70.62	71.08	n/a
CC [30]	69.96	n/a	n/a
RKD [28]	70.40	n/a	68.50
FT [17]	n/a	n/a	69.88
FSP [42]	70.58	n/a	n/a
CRD [36]	71.17	71.25	69.07
OFD [12]	71.03	n/a	71.33
RRD (ours)	72.03	71.97	71.54
RRD+KD (ours)	71.99	71.88	71.56

10.4. Correlation Evaluation

To quantify relational consistency between teacher and student embeddings, we compute three complementary metrics on CIFAR-100: Frobenius norm difference, Pearson correlation, and structural similarity index (SSIM). Evaluations are conducted

Table 17. Test top-1 accuracy (%) comparison for transfer learning, where WRN-16-2 (student) is distilled from WRN-40-2 (teacher) and transferred from CIFAR-100 to STL-10 and Tiny ImageNet. Results are averaged over five checkpoints. The abbreviations are listed in Section 7.1.

	λ	β	CIFAR-100 → STL-10	CIFAR-100 → Tiny ImageNet
Teacher	n/a	n/a	68.6	31.5
Student	n/a	n/a	69.7	33.7
without KL divergence				
CRD [36]	n/a	n/a	71.6	35.6
RRD (ours)	0	1	72.0 ± 0.65	35.5 ± 0.36
RRD (ours)	0	1.5	72.3 ± 0.31	35.5 ± 0.42
with KL divergence				
CRD+KD [36]	n/a	n/a	72.2	35.5
RRD+KD (ours)	0.9	1	71.9 ± 0.46	35.0 ± 0.13
RRD+KD (ours)	0.9	1.5	72.0 ± 0.28	35.2 ± 0.38

Table 18. Correlation matrix alignment metrics on CIFAR-100. Quantitative comparison of relational alignment between teacher (WRN-40-2) and student (WRN-40-1) embeddings. Lower Frobenius distance and higher Pearson/SSIM indicate stronger relational consistency. The abbreviations are listed in Section 7.1.

Method	Frobenius ↓	Pearson ↑	SSIM ↑
Vanilla	10.491	0.951	0.935
KD [14]	2.868	0.994	0.992
FitNet [32]	10.491	0.950	0.934
AT [43]	9.450	0.953	0.941
SP [37]	8.932	0.955	0.945
CC [30]	11.148	0.944	0.928
VID [1]	9.180	0.960	0.950
RKD [28]	8.994	0.963	0.953
PKT [29]	6.285	0.982	0.973
ABOUND [13]	9.523	0.956	0.943
FT [17]	12.584	0.932	0.913
NST [16]	10.574	0.938	0.923
CRD [36]	8.862	0.946	0.938
CRD+KD [36]	3.265	0.992	0.989
RRD (ours)	6.969	0.970	0.964
RRD+KD (ours)	2.720	0.995	0.993

using WRN-40-2 as the teacher and WRN-40-1 as the student, measuring the degree to which each method preserves the teacher's inter-class correlation structure among logits. Table 18 summarizes the results across baselines. RRD shows superior correlation alignment, and its combination with KD further improves consistency, reflecting the complementary effects of relational and probabilistic supervision.

10.5. Semantic Evaluation

We further analyze the semantic organization of student embeddings on CIFAR-100, distilled from WRN-40-2 (teacher) to WRN-40-1 (student). Penultimate-layer features are extracted and ℓ_2-normalized before computing intra-class compactness, inter-class separation, normalized mutual information (NMI) with superclass taxonomy, and retrieval precision via mean average precision at 5 (mAP@5). Table 19 presents the results over multiple baselines. RRD produces compact and well-separated clusters, while RRD+KD achieves the highest overall NMI and retrieval accuracy, confirming improved semantic structure preservation in the distilled representation.

Table 19. Semantic similarity preservation on CIFAR-100. Evaluation of intra-/inter-class structure, NMI, and retrieval mAP@5 of student embeddings under different distillation methods. The abbreviations are listed in Section 7.1.

Method	Intra ↓	Inter ↑	NMI ↑	mAP@5 ↑
Vanilla	0.945	0.866	0.551	81.1
KD [14]	0.914	1.012	0.546	84.2
FitNet [32]	0.945	0.873	0.556	82.0
AT [43]	0.882	0.891	0.565	82.8
SP [37]	0.768	0.905	0.568	84.2
CC [30]	0.948	0.872	0.562	81.2
VID [1]	0.905	0.892	0.562	82.6
RKD [28]	0.985	0.897	0.549	82.5
PKT [29]	0.882	0.942	0.581	84.9
ABOUND [13]	0.944	0.878	0.557	82.1
FT [17]	0.882	0.887	0.550	81.9
NST [16]	0.831	0.879	0.564	80.8
CRD [36]	0.910	0.915	0.509	84.6
CRD+KD [36]	0.973	0.926	0.463	85.2
RRD (ours)	0.928	0.909	0.544	83.9
RRD+KD (ours)	0.902	1.023	0.562	85.9

11. Visualization

This section provides analytical visualizations to better interpret the structural and semantic properties of the representations learned through distillation. We first analyze inter-class correlation structures (Section 11.1), followed by t-SNE projections of feature embeddings (Section 11.2), and finally qualitative nearest-neighbor retrievals (Section 11.3) to assess semantic consistency in feature space.

11.1. Visualisation of Inter-class Correlations

To examine inter-class relational structure, we visualize the correlation matrices between class-wise mean logits of teacher and student networks. Using WRN-40-2 as the teacher and WRN-40-1 as the student, each model's logits are normalized to zero mean and unit variance, averaged per class on CIFAR-100, and used to form 100×100 correlation matrices. We then compute and visualize the absolute difference between teacher and student matrices, focusing on off-diagonal elements to capture inter-class relationships. Figure 4 compares these differences across methods. RRD produces smoother and more consistent correlation structures than other relational methods, and when combined with KD, further reduces structural deviations.

11.2. Visualization of t-SNE Embeddings

We employ t-SNE [39] to visualize the embedding distributions of student and teacher models and to analyze how well the student captures the teacher's relational topology. Features are extracted from the CIFAR-100 validation set using WRN-40-2 (teacher) and WRN-40-1 (student), ℓ_2-normalized, and projected into two dimensions using a perplexity of 40 and 300 optimization iterations. Two visualization sets are provided: one focusing on the first 20 classes for clarity and another on all 100 classes for completeness. As shown in Figures 5 and 6, RRD's embeddings exhibit tighter class clusters and smoother global organization, preserving both local and global relationships between samples.

11.3. Nearest-neighbor Retrieval

For qualitative evaluation of semantic consistency, we perform nearest-neighbor (NN) retrieval using CIFAR-100 validation features. Features from WRN-40-2 (teacher) and WRN-40-1 (student) are extracted from the penultimate layer, ℓ_2-normalized, and compared via cosine similarity. For each query image, we retrieve the top-5 most similar samples within the validation set, excluding the query itself. Figure 7 presents qualitative examples of 5-NN retrievals. RRD retrieves semantically coherent neighbors consistent with the teacher's feature space, while RRD+KD exhibits the strongest semantic precision, with nearly all retrieved samples belonging to the correct class.

12. Discussion

This section discusses key aspects of our experimental methodology and situates our approach relative to existing distillation paradigms. We first justify our experimental setup and dataset selection (Section 12.1), then provide a deeper comparison with related relational methods (Section 12.2), highlighting both conceptual and empirical distinctions.

12.1. On the Choice of Experimental Setup and Dataset

Given computational resource constraints, we select CIFAR-100 as our primary dataset for demonstrating and analyzing our method, while also validating scalability on ImageNet. Our choice to focus ablation studies on CIFAR-100 rather than ImageNet is well-justified, as our approach builds upon and shares fundamental similarities with [36]. Specifically, we demonstrate in Section 8.1 how our method can be viewed as a relaxation of the contrastive learning framework used in CRD, allowing us to adopt similar hyperparameter settings. Our extensive ablation studies on CIFAR-100 yield an important insight: while we thoroughly investigate different hyperparameter configurations, we find that performance remains relatively stable across a range of settings, provided the core principles of our approach are maintained. This robustness to hyperparameter variation further supports our experimental design choice of comprehensive CIFAR-100 evaluation with additional ImageNet validation.

12.2. Comparison with Related Methods

While space constraints prevent us from presenting exhaustive comparisons in the main paper, we provide comprehensive experimental results against all cited methods for CIFAR-100 in Section 7.1. Of particular note is our comparison with other *relational*-based approaches like RKD [28], PKT [29], and CC [30], where our method demonstrates consistent improvements. These improvements are most pronounced in same-architecture pairs using WRN and ResNet variants, while maintaining robust performance in more challenging cross-architecture scenarios (e.g., ResNet to MobileNet transfers) where competing methods often struggle. The effectiveness of our approach is particularly evident in cases with substantial architectural differences between teacher and student networks, showing better adaptation to varying network capacities than existing methods. For detailed performance metrics and analysis across different architectures and datasets, we encourage readers to refer to our supplementary material and the original works cited throughout this paper.

13. Broader Impact

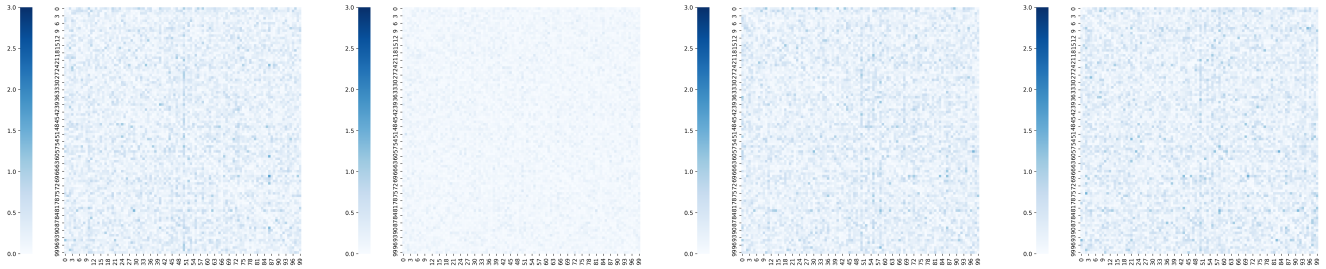
The presented research should be categorized as research in the field of knowledge distillation. The primary goal is to reduce computational demands, thereby lowering the energy requirements of AI systems and contributing to more sustainable technology deployment. However, this technique also harbors 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 and adverse impacts.

Table 20. 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. Results of our method are averaged over five runs. For RRD we set $\beta = 1$ and $\lambda = 0$. For RRD+KD we set $\beta = 1.5$ and $\lambda = 0.9$. The abbreviations are listed in Section 7.1.

Teacher	WRN-40-2	WRN-40-2	resnet-56	resnet-110	resnet-110	resnet-32x4	VGG-13
Student	WRN-16-2	WRN-40-1	resnet-20	resnet-20	resnet-32	resnet-8x4	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 [14]	74.92	73.54	70.66	70.67	73.08	73.33	72.98
FitNet [32]	73.58 (↓)	72.24 (↓)	69.21 (↓)	68.99 (↓)	71.06 (↓)	73.50 (↑)	71.02 (↓)
AT [43]	74.08 (↓)	72.77 (↓)	70.55 (↓)	70.22 (↓)	72.31 (↓)	73.44 (↑)	71.43 (↓)
SP [37]	73.83 (↓)	72.43 (↓)	69.67 (↓)	70.04 (↓)	72.69 (↓)	72.94 (↓)	72.68 (↓)
CC [30]	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 [28]	73.35 (↓)	72.22 (↓)	69.61 (↓)	69.25 (↓)	71.82 (↓)	71.90 (↓)	71.48 (↓)
PKT [29]	74.54 (↓)	73.45 (↓)	70.34 (↓)	70.25 (↓)	72.61 (↓)	73.64 (↑)	72.88 (↓)
AB [13]	72.50 (↓)	72.38 (↓)	69.47 (↓)	69.53 (↓)	70.98 (↓)	73.17 (↓)	70.94 (↓)
FT [17]	73.25 (↓)	71.59 (↓)	69.84 (↓)	70.22 (↓)	72.37 (↓)	72.86 (↓)	70.58 (↓)
FSP [42]	72.91 (↓)	n/a	69.95 (↓)	70.11 (↓)	71.89 (↓)	72.62 (↓)	70.33 (↓)
NST [16]	73.68 (↓)	72.24 (↓)	69.60 (↓)	69.53 (↓)	71.96 (↓)	73.30 (↓)	71.53 (↓)
CRD [36]	75.48 (↑)	74.14 (↑)	71.16 (↑)	71.46 (↑)	73.48 (↑)	75.51 (↑)	73.94 (↑)
CRD+KD [36]	75.64 (↑)	74.38 (↑)	71.63 (↑)	71.56 (↑)	73.75 (↑)	75.46 (↑)	74.29 (↑)
OFD [12]	75.24 (↑)	74.33 (↑)	70.38 (↓)	n/a	73.23 (↑)	74.95 (↑)	73.95 (↑)
WSLD [48]	n/a	73.74 (↑)	71.53 (↑)	n/a	73.36 (↑)	74.79 (↑)	n/a
IPWD [27]	n/a	74.64 (↑)	71.32 (↑)	n/a	73.91 (↑)	76.03 (↑)	n/a
SRRL [41]	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 [8]	76.36 (↑)	n/a	72.03 (↑)	n/a	73.74 (↑)	n/a	n/a
NORM [23]	75.65 (↑)	74.82 (↑)	71.35 (↑)	71.55 (↑)	73.67 (↑)	76.49 (↑)	73.95 (↑)
NORM+KD [23]	76.26 (↑)	75.42 (↑)	71.61 (↑)	72.00 (↑)	74.95 (↑)	76.98 (↑)	74.46 (↑)
NORM+CRD [23]	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 [49]	76.67 (↑)	75.95 (↑)	73.21 (↑)	72.33 (↑)	74.98 (↑)	76.42 (↑)	74.97 (↑)
FKD [10]	n/a	n/a	n/a	n/a	n/a	75.57 (↑)	73.78 (↑)
ITRD (corr) [26]	75.85 (↑)	74.90 (↑)	71.45 (↑)	71.77 (↑)	74.02 (↑)	75.63 (↑)	74.70 (↑)
ITRD (corr+mi) [26]	76.12 (↑)	75.18 (↑)	71.47 (↑)	71.99 (↑)	74.26 (↑)	76.19 (↑)	74.93 (↑)
TTM [47]	76.23 (↑)	74.32 (↑)	71.83 (↑)	71.46 (↑)	73.97 (↑)	76.17 (↑)	74.33 (↑)
WTTM [47]	76.37 (↑)	74.58 (↑)	71.92 (↑)	71.67 (↑)	74.13 (↑)	76.06 (↑)	74.44 (↑)
WTTM+CRD [47]	76.61 (↑)	74.94 (↑)	72.20 (↑)	72.13 (↑)	74.52 (↑)	76.65 (↑)	74.71 (↑)
WTTM+ITRD [47]	76.65 (↑)	75.34 (↑)	72.16 (↑)	72.20 (↑)	74.36 (↑)	77.36 (↑)	75.13 (↑)
DKD [46]	76.24 (↑)	74.81 (↑)	71.97 (↑)	n/a	74.11 (↑)	76.32 (↑)	74.68 (↑)
FCFD [22]	76.34 (↑)	75.43 (↑)	71.68 (↑)	n/a	n/a	76.80 (↑)	74.86 (↑)
FCFD+KD [22]	76.43 (↑)	75.46 (↑)	71.96 (↑)	n/a	n/a	76.62 (↑)	75.22 (↑)
CAT-KD [9]	75.60 (↑)	74.82 (↑)	71.62 (↑)	n/a	73.62 (↑)	76.91 (↑)	74.65 (↑)
DIST [15]	n/a	74.73 (↑)	71.75 (↑)	n/a	n/a	76.31 (↑)	n/a
CTKD [19]	75.45 (↑)	73.93 (↑)	71.19 (↑)	70.99 (↑)	73.52 (↑)	n/a	73.52 (↑)
RRD (ours)	75.85 (↑)	74.61 (↑)	71.89 (↑)	71.92 (↑)	73.73 (↑)	75.77 (↑)	74.01 (↑)
RRD+KD (ours)	75.67 (↑)	74.68 (↑)	72.03 (↑)	71.75 (↑)	73.96 (↑)	75.53 (↑)	74.37 (↑)

Table 21. 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. Results of our method are averaged over five runs. For RRD we set $\beta = 1$ and $\lambda = 0$. For RRD+KD we set $\beta = 1.5$ and $\lambda = 0.9$. The abbreviations are listed in Section 7.1.

Teacher	VGG-13	ResNet-50	ResNet-50	ResNet-32x4	ResNet-32x4	WRN-40-2
Student	MobileNet-v2	MobileNet-v2	VGG-8	ShuffleNet-v1	ShuffleNet-v2	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 [14]	67.37	67.35	73.81	74.07	74.45	74.83
FitNet [32]	64.14 (↓)	63.16 (↓)	70.69 (↓)	73.59 (↓)	73.54 (↓)	73.73 (↓)
AT [43]	59.40 (↓)	58.58 (↓)	71.84 (↓)	71.73 (↓)	72.73 (↓)	73.32 (↓)
SP [37]	66.30 (↓)	68.08 (↑)	73.34 (↓)	73.48 (↓)	74.56 (↑)	74.52 (↓)
CC [30]	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 [28]	64.52 (↓)	64.43 (↓)	71.50 (↓)	72.28 (↓)	73.21 (↓)	72.21 (↓)
PKT [29]	67.13 (↓)	66.52 (↓)	73.01 (↓)	74.10 (↑)	74.69 (↑)	73.89 (↓)
AB [13]	66.06 (↓)	67.20 (↓)	70.65 (↓)	73.55 (↓)	74.31 (↓)	73.34 (↓)
FT [17]	61.78 (↓)	60.99 (↓)	70.29 (↓)	71.75 (↓)	72.50 (↓)	72.03 (↓)
NST [16]	58.16 (↓)	64.96 (↓)	71.28 (↓)	74.12 (↑)	74.68 (↑)	76.09 (↑)
CRD [36]	69.73 (↑)	69.11 (↑)	74.3 (↑)	75.11 (↑)	75.65 (↑)	76.05 (↑)
CRD+KD [36]	69.94 (↑)	69.54 (↑)	74.58 (↑)	75.12 (↑)	76.05 (↑)	76.27 (↑)
OFD [12]	69.48 (↑)	69.04 (↑)	n/a	75.98 (↑)	76.82 (↑)	75.85 (↑)
WSLD [48]	n/a	68.79 (↑)	73.80 (↓)	75.09 (↑)	n/a	75.23 (↑)
IPWD [27]	n/a	70.25 (↑)	74.97 (↑)	76.03 (↑)	n/a	76.44 (↑)
SRRL [41]	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 [8]	n/a	n/a	n/a	77.18 (↑)	77.54 (↑)	77.24 (↑)
NORM [23]	68.94 (↑)	70.56 (↑)	75.17 (↑)	77.42 (↑)	78.07 (↑)	77.06 (↑)
NORM+KD [23]	69.38 (↑)	71.17 (↑)	75.67 (↑)	77.79 (↑)	78.32 (↑)	77.63 (↑)
NORM+CRD [23]	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 [49]	n/a	n/a	n/a	n/a	n/a	n/a
FKD [10]	n/a	n/a	74.61 (↑)	75 (↑)	n/a	n/a
ITRD (corr) [26]	69.97 (↑)	71.41 (↑)	75.71 (↑)	76.8 (↑)	77.27 (↑)	77.35 (↑)
ITRD (corr+mi) [26]	70.39 (↑)	71.34 (↑)	75.49 (↑)	76.91 (↑)	77.40 (↑)	77.09 (↑)
TTM [47]	68.98 (↑)	69.24 (↑)	74.87 (↑)	74.18 (↑)	76.57 (↑)	75.39 (↑)
WTTM [47]	69.16 (↑)	69.59 (↑)	74.82 (↑)	74.37 (↑)	76.55 (↑)	75.42 (↑)
WTTM+CRD [47]	70.30 (↑)	70.84 (↑)	75.30 (↑)	75.82 (↑)	77.04 (↑)	76.86 (↑)
WTTM+ITRD [47]	70.70 (↑)	71.56 (↑)	76.00 (↑)	77.03 (↑)	77.68 (↑)	77.44 (↑)
DKD [46]	69.71 (↑)	70.35 (↑)	n/a	76.45 (↑)	77.07 (↑)	76.70 (↑)
FCFD [22]	70.67 (↑)	71.07 (↑)	n/a	78.12 (↑)	78.20 (↑)	77.81 (↑)
FCFD+KD [22]	70.65 (↑)	71.00 (↑)	n/a	78.12 (↑)	78.18 (↑)	77.99 (↑)
CAT-KD [9]	69.13 (↑)	71.36 (↑)	n/a	78.26 (↑)	78.41 (↑)	77.35 (↑)
DIST [15]	n/a	68.66 (↑)	n/a	76.34 (↑)	77.35 (↑)	n/a
CTKD [19]	68.46 (↑)	68.47 (↑)	n/a	74.78 (↑)	75.31 (↑)	75.78 (↑)
RRD (ours)	69.61 (↑)	70.11 (↑)	74.30 (↑)	75.60 (↑)	76.31 (↑)	75.98 (↑)
RRD+KD (ours)	69.99 (↑)	69.65 (↑)	74.53 (↑)	76.68 (↑)	76.87 (↑)	76.64 (↑)

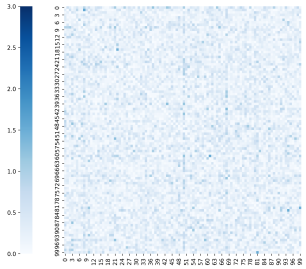


(a) Vanilla
Mean: 0.24, Max: 1.66

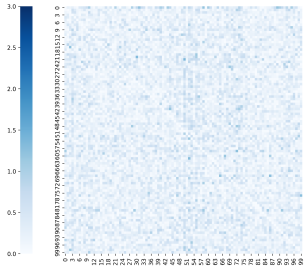
(b) KD [14]
Mean: 0.09, Max: 0.49

(c) FitNets [32]
Mean: 0.25, Max: 1.38

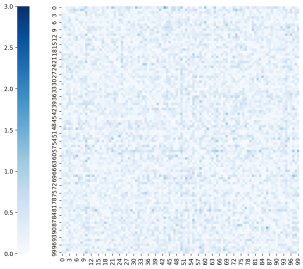
(d) AT [43]
Mean: 0.26, Max: 1.43



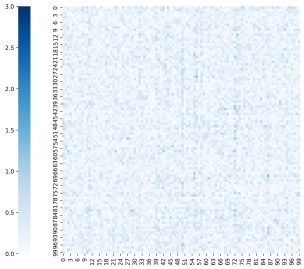
(e) SP [37]
Mean: 0.25, Max: 1.57



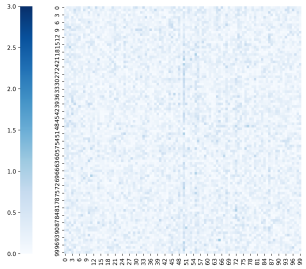
(f) CC [30]
Mean: 0.234, Max: 1.37



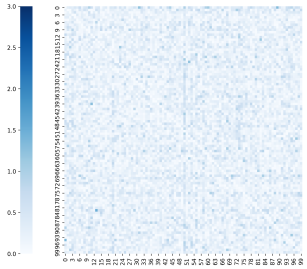
(g) VID [1]
Mean: 0.23, Max: 1.46



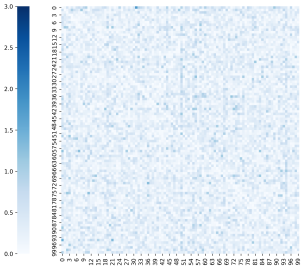
(h) RKD [28]
Mean: 0.21, Max: 1.24



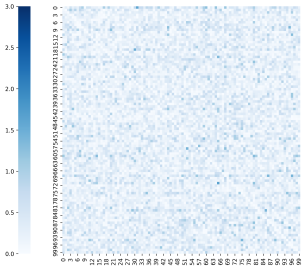
(i) PKT [29]
Mean: 0.19, Max: 1.14



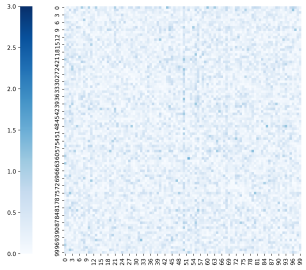
(j) AB [13]
Mean: 0.24, Max: 1.49



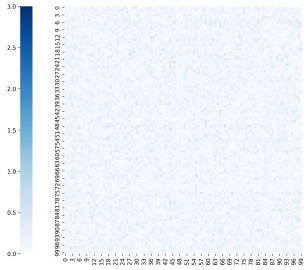
(k) FT [17]
Mean: 0.26, Max: 1.76



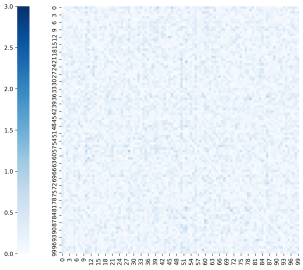
(l) NST [16]
Mean: 0.28, Max: 1.65



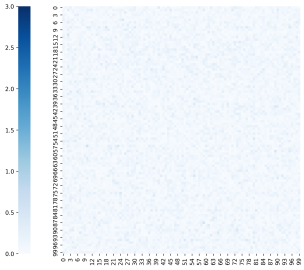
(m) CRD [36]
Mean: 0.23, Max: 1.56



(n) CRD+KD [36]
Mean: 0.10, Max: 0.57

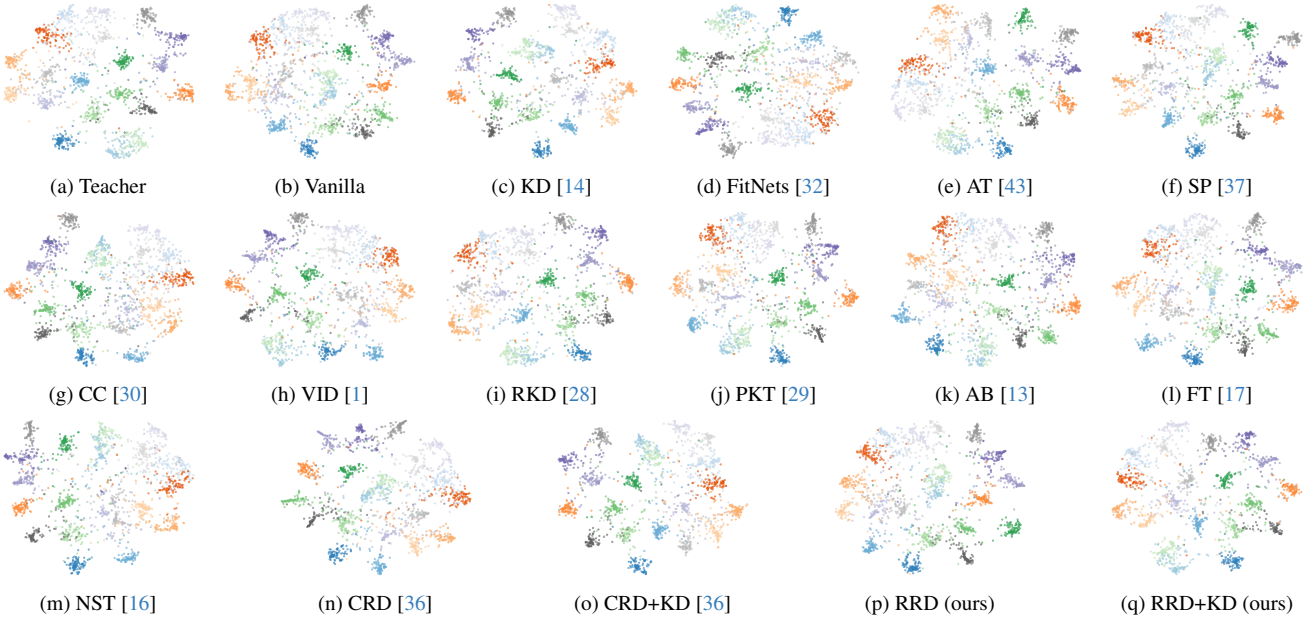


(o) RRD (ours)
Mean: 0.18, Max: 0.99



(p) RRD+KD (ours)
Mean: 0.07, Max: 0.55

Figure 4. Correlation matrix of the average logit difference between teacher and student logits on CIFAR-100. We use WRN-40-2 as the teacher and WRN-40-1 as the student. Methods have been re-implemented accordingly. The abbreviations are listed in Section 7.1.

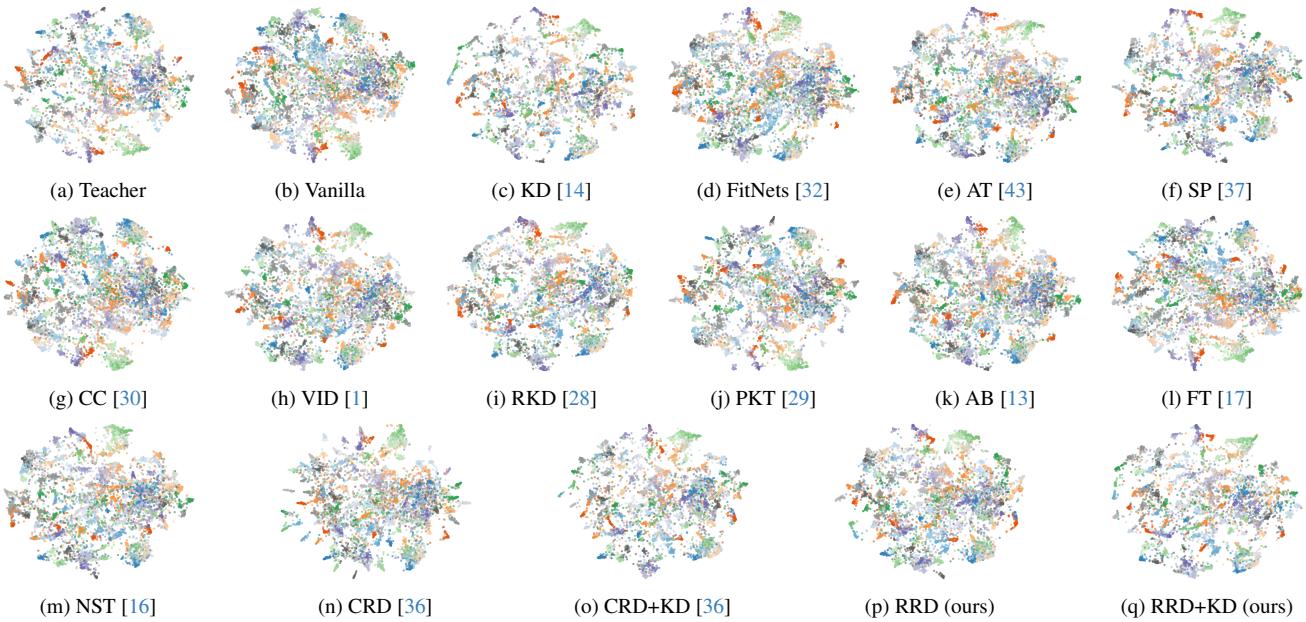


(a) Teacher (b) Vanilla (c) KD [14] (d) FitNets [32] (e) AT [43] (f) SP [37]

(g) CC [30] (h) VID [1] (i) RKD [28] (j) PKT [29] (k) AB [13] (l) FT [17]

(m) NST [16] (n) CRD [36] (o) CRD+KD [36] (p) RRD (ours) (q) RRD+KD (ours)

Figure 5. t-SNE visualizations of embeddings from teacher and student networks on CIFAR-100 (first 20 classes). We use WRN-40-2 as the teacher and WRN-40-1 as the student. Methods have been re-implemented accordingly. The abbreviations are listed in Section 7.1.



(a) Teacher (b) Vanilla (c) KD [14] (d) FitNets [32] (e) AT [43] (f) SP [37]

(g) CC [30] (h) VID [1] (i) RKD [28] (j) PKT [29] (k) AB [13] (l) FT [17]

(m) NST [16] (n) CRD [36] (o) CRD+KD [36] (p) RRD (ours) (q) RRD+KD (ours)

Figure 6. t-SNE visualization of embeddings from teacher and student networks on the full CIFAR-100 dataset. WRN-40-2 is used as the teacher and WRN-40-1 as the student. All methods are re-implemented accordingly. The abbreviations are listed in Section 7.1.

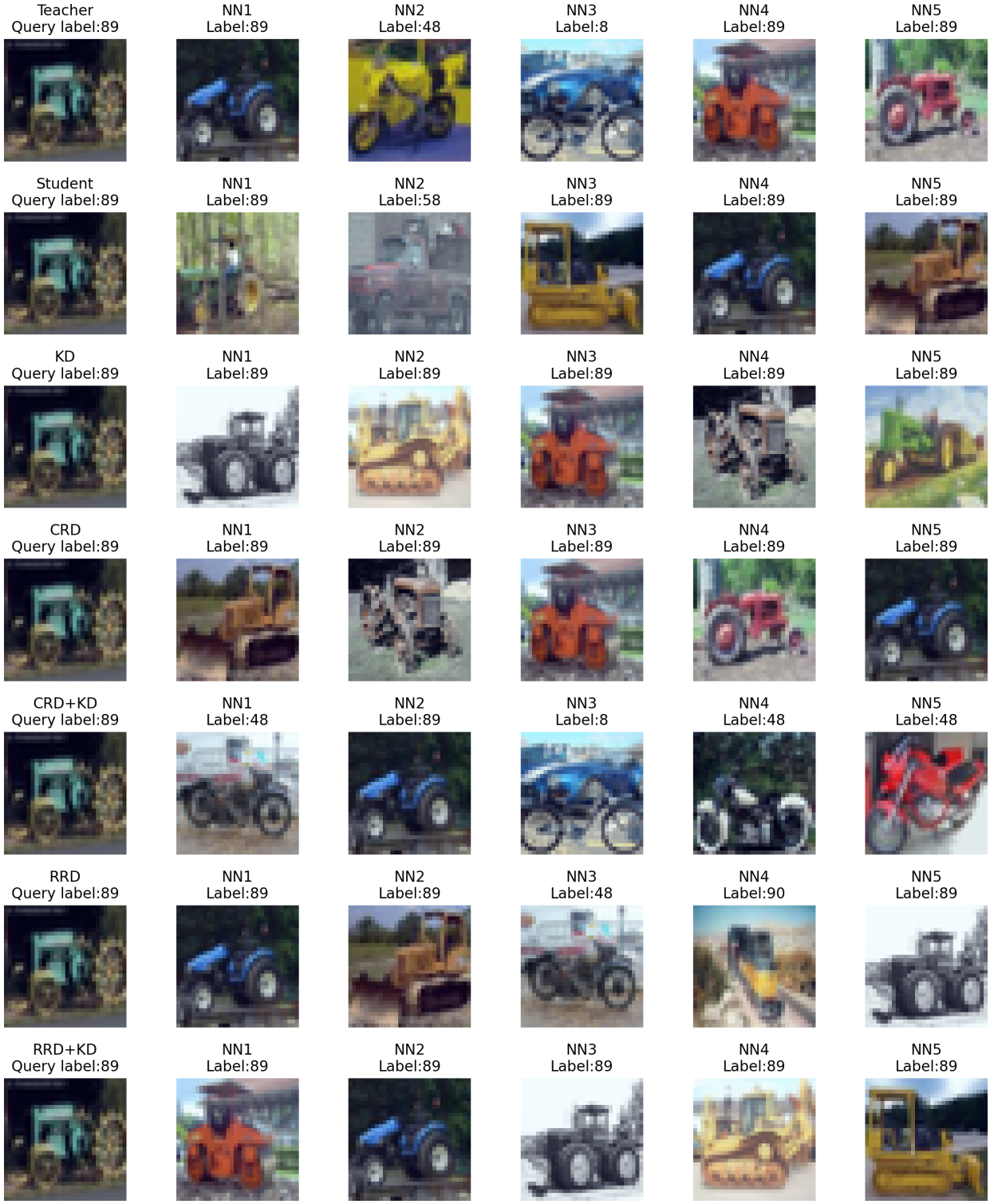


Figure 7. Nearest-neighbor retrieval on CIFAR-100. WRN-40-2 is used as the teacher and WRN-40-1 as the student. Qualitative 5-NN retrieval results showing semantic consistency of student representations distilled with different methods. The abbreviations are listed in Section 7.1.

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, 2, 12, 13, 15, 16, 17, 18

[2] Defang Chen, Jian-Ping Mei, Yuan Zhang, Can Wang, Yan Feng, and Chun Chen. Cross-layer distillation with semantic calibration, 2021. 2, 15, 16

[3] Defang Chen, Jian-Ping Mei, Hailin Zhang, Can Wang, Yan Feng, and Chun Chen. Knowledge distillation with the reused teacher classifier, 2022. 2, 15, 16

[4] Liqun Chen, Dong Wang, Zhe Gan, Jingjing Liu, Ricardo Henao, and Lawrence Carin. Wasserstein contrastive representation distillation, 2021. 2, 15, 16

[5] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review, 2021. 2, 15, 16

[6] Adam Coates and Andrew Y. Ng. The importance of encoding versus training with sparse coding and vector quantization. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, page 921–928, Madison, WI, USA, 2011. Omnipress. 3

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 3

[8] Xueqing Deng, Dawei Sun, Shawn Newsam, and Peng Wang. Distpro: Searching a fast knowledge distillation process via meta optimization, 2022. 2, 15, 16

[9] Ziyao Guo, Haonan Yan, Hui Li, and Xiaodong Lin. Class attention transfer based knowledge distillation, 2023. 2, 15, 16

[10] Bobby He and Mete Ozay. Feature kernel distillation. In *International Conference on Learning Representations*, 2022. 2, 15, 16

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[12] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation, 2019. 2, 11, 15, 16

[13] 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, 2, 12, 13, 15, 16, 17, 18

[14] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 1, 2, 9, 10, 11, 12, 13, 15, 16, 17, 18

[15] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher, 2022. 2, 15, 16

[16] 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, 2, 12, 13, 15, 16, 17, 18

[17] 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, 2, 11, 12, 13, 15, 16, 17, 18

[18] Alex Krizhevsky. Learning multiple layers of features from tiny images. pages 32–33, 2009. 3, 5

[19] Zheng Li, Xiang Li, Lingfeng Yang, Borui Zhao, Renjie Song, Lei Luo, Jun Li, and Jian Yang. Curriculum temperature for knowledge distillation, 2022. 2, 15, 16

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 3

[21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2017. 3

[22] Dongyang Liu, Meina Kan, Shiguang Shan, and Xilin Chen. Function-consistent feature distillation, 2023. 2, 15, 16

[23] Xiaolong Liu, Lujun Li, Chao Li, and Anbang Yao. Norm: Knowledge distillation via n-to-one representation matching, 2023. 2, 15, 16

[24] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 116–131, 2018. 3

[25] Roy Miles and Krystian Mikolajczyk. Understanding the role of the projector in knowledge distillation, 2024. 3

[26] Roy Miles, Adrian Lopez Rodriguez, and Krystian Mikolajczyk. Information theoretic representation distillation, 2022. 2, 15, 16

[27] Yulei Niu, Long Chen, Chang Zhou, and Hanwang Zhang. Respecting transfer gap in knowledge distillation, 2022. 2, 15, 16

[28] 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, 2, 11, 12, 13, 14, 15, 16, 17, 18

[29] 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. 2, 12, 13, 14, 15, 16, 17, 18

[30] 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, 2, 11, 12, 13, 14, 15, 16, 17, 18

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. 3

[32] 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, 2, 12, 13, 15, 16, 17, 18

[33] 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. 3

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 3

[35] Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation, 2024. 2

[36] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation, 2022. 1, 2, 3, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18

[37] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1365–1374, 2019. 1, 2, 11, 12, 13, 15, 16, 17, 18

[38] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. 4

[39] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. 13

[40] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019. 3

[41] Jing Yang, Brais Martinez, Adrian Bulat, and Georgios Tzimiropoulos. Knowledge distillation via softmax regression representation learning. In *International Conference on Learning Representations*, 2021. 2, 15, 16

[42] 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, 2, 11, 15

[43] 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, 2, 11, 12, 13, 15, 16, 17, 18

[44] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks, 2017. 3

[45] 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. 3

[46] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation, 2022. 2, 3, 15, 16

[47] Kaixiang Zheng and En-Hui Yang. Knowledge distillation based on transformed teacher matching, 2024. 2, 15, 16

[48] 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. 2, 15, 16

[49] Jinguo Zhu, Shixiang Tang, Dapeng Chen, Shijie Yu, Yakun Liu, Aijun Yang, Mingzhe Rong, and Xiaohua Wang. Complementary relation contrastive distillation, 2021. 2, 15, 16