# VRM: Knowledge Distillation via Virtual Relation Matching

Weijia Zhang[1]　　　　Fei Xie[1]　　　Weidong Cai[2]　　　Chao Ma[1,*]

[1] Shanghai Jiao Tong University　　[2] The University of Sydney

{weijia.zhang, feixie, chaoma}@sjtu.edu.cn　　tom.cai@sydney.edu.au

## Abstract

*Knowledge distillation (KD) aims to transfer the knowledge of a more capable yet cumbersome teacher model to a lightweight student model. In recent years, relation-based KD methods have fallen behind, as their instance-matching counterparts dominate in performance. In this paper, we revive relational KD by identifying and tackling several key issues in relation-based methods, including their susceptibility to overfitting and spurious responses. Specifically, we transfer novelly constructed affinity graphs that compactly encapsulate a wealth of beneficial inter-sample, inter-class, and inter-view correlations by exploiting virtual views and relations as a new kind of knowledge. As a result, the student has access to richer guidance signals and stronger regularisation throughout the distillation process. To further mitigate the adverse impact of spurious responses, we prune the affinity graphs by dynamically detaching redundant and unreliable edges. Extensive experiments on CIFAR-100, ImageNet, and MS-COCO datasets demonstrate the superior performance of the proposed virtual relation matching (VRM) method, where it consistently sets new state-of-the-art records over a range of models, architectures, tasks, and set-ups. For instance, VRM for the first time hits 74.0% accuracy for ResNet50→MobileNetV2 distillation on ImageNet, and improves DeiT-T by 14.44% on CIFAR-100 with a ResNet56 teacher. The code and models are released at* https://github.com/VISION-SJTU/VRM.

## 1. Introduction

Deep learning is achieving incredible performance at the cost of increasing model complexity and overheads. As a consequence, large and cumbersome neural models struggle to work in resource-constrained environments. Knowledge distillation (KD) [27] has been proposed to address this issue by transferring the knowledge of larger and more capable models to smaller and lightweight ones that are resource-friendly. KD works by minimising the dis-
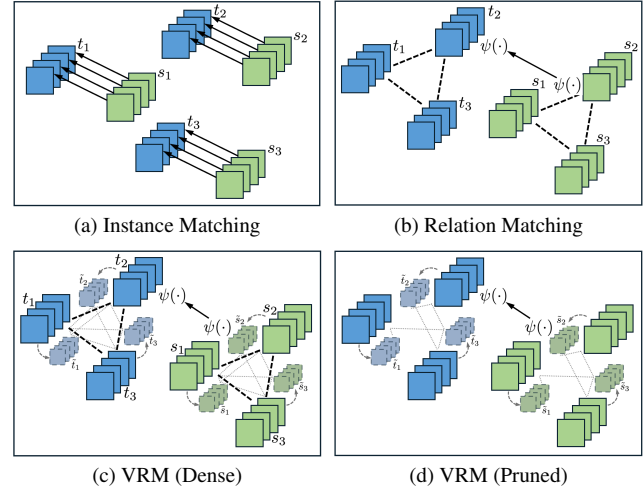


Figure 1. Conceptual illustration of VRM compared to existing KD methods based on instance matching and relation matching.

tance between compact representations of knowledge extracted from the teacher and student models. It has found widespread application across a spectrum of downstream tasks [18, 22, 30, 34, 62, 64, 67, 70, 75, 77, 83].

According to the type of knowledge representations to be transferred, KD methods can be broadly categorised into feature-based [56], logit-based [27], and relation-based [52] approaches. The former two directly match the feature maps or logit vectors produced by the teacher and student models for each training sample, which is essentially *instance matching* (IM). By contrast, *relation matching* (RM) methods construct and match structured relations extracted within a batch of model responses. A conceptual illustration is presented in Figs. 1a and 1b.

Instance matching has been the prevailing distillation approach in recent years. Popular KD benchmarks see a dominance by IM methods such as FCFD [45], NORM [47], and CRLD [76], with different downstream tasks successfully tackled by directly adopting IM-based distillation [6, 14, 62, 69]. Yet, recent studies discovered that relational knowledge is more robust to variations in neural architectures, data modalities, and tasks [52, 61]. Meanwhile, methods transferring relations have also achieved promising

---

*Corresponding author.

performance for a range of tasks, including but not limited to segmentation [67] and detection [15, 32].

Despite growing interest, relation-based methods still fall significantly short compared to their instance matching counterparts. Even the strongest RM method has been outperformed easily by recent IM solutions [29] (see Tabs. 1 and 2). RM-based methods also struggle with more challenging tasks such as object detection [29]. Moreover, previous RM solutions are primarily limited to matching inter-sample [29, 53, 61], inter-class [29], or inter-channel [46, 73] relations via simple Gram matrices. To our best knowledge, no different forms of relations other than these have been proposed since the work of DIST [29].

This paper fills this gap with a new kind of relations for KD – *inter-view* relations (Fig. 1c), which seamlessly and compactly integrate with previous inter-sample and inter-class relations. Our designs are motivated by two important observations made about RM methods in a set of pilot experiments: 1) RM methods are more susceptible to overfitting than IM methods; 2) RM methods are subject to an adverse gradient propagation effect. We empirically find that incorporating richer and more diverse relations into the matching objective helps mitigate both issues.

To this end, we generate *virtual views* of samples through simple transformations, followed by constructing *virtual affinity graphs* and transferring the *virtual relations* between real and virtual samples along the edges. In lieu of Gram matrices that suffer from significant knowledge loss [29, 53, 61], we preserve the raw relations along the secondary dimension as auxiliary knowledge which adds to the types and density of relational knowledge transferred. Moreover, we also prune our affinity graphs by striping away both redundant and unreliable edges to further alleviate the propagating gradients of spurious samples (Fig. 1d).

The above insights and remedies altogether lead to a novel *Virtual Relation Matching* (VRM) framework for knowledge distillation. VRM is conceptually simple, easy to implement, and devoid of complicated training procedures. It is capable of transferring rich, sophisticated knowledge robust to overfitting and spurious signals. VRM sets new state-of-the-art performance for different datasets, tasks, and settings. Perhaps more significant is that VRM makes relation-based methods regain competitiveness and back in the lead over instance matching approaches in different scenarios. To summarise, the contributions of this work include:

- We make an early effort to present comparative analyses of existing KD methods through the lens of training dynamics and sample-wise gradients, and identify overfitting and spurious gradient diffusion as two main cruxes in relational KD methods.
- We distill richer, more diverse relations by generating virtual views, constructing virtual affinity graphs, and matching virtual relations. We also for the first time tackle relational KD with considerations of spurious samples and gradients by pruning redundant and unreliable edges, alongside designs to relax the matching criterion.
- We present the streamlined VRM framework for knowledge distillation, with extensive experimental results on a diversity of models and tasks to highlight its superior performance, alongside rigorous analyses on the soundness and efficiency of our designs.

## 2. Related Work

**KD via instance matching.** KD was first proposed in Hinton et al. [27], where a student is trained to mimic the prediction of a pre-trained teacher model for each sample. Follow-up works have mostly followed such *instance matching* (IM) paradigm, and can be categorised into logit (or prediction)-based and feature-based methods according to what is matched. Logit-based KD has evolved from using adaptive-softened logits [33, 41] to decoupling target- and non-target logits [24, 72, 81] and applying logit transformation [59, 82]. Others [50, 58] set up auxiliary ad-hoc networks between teacher and student to facilitate logit transfer. Feature-based methods minimise the distance between the feature maps [1, 25, 56] or salient regions within in features [23, 74] from specified layers in both teacher and student networks. Some also design sophisticated distillation paths [7, 42]. All these methods are based on instance-wise transfer of knowledge, and are herein referred to as the "IM" methods, as illustrated by Fig. 1a.

**KD via relation matching.** Some works transfer instead mutual relations mined amongst the network outputs extracted from a batch of training instances. These *relation matching* (RM) methods usually involve constructing network outputs into compact relation representations that encode rich higher-order information, as depicted in Fig. 1b. Different relation encoding functions mare, including inter-sample [29, 48, 52–54, 61], inter-class [29], inter-channel [46, 73, 78], inter-layer [48], and contrastive [60] relations. To date, IM solutions have dominated KD with their superior performance, leading top-performing relation-based methods by considerable margins. While many new IM methods are found within the last two years, frustratingly few RM solutions are being proposed. In this work, we strive to close this gap with a new kind of relations for KD — inter-view virtual relations, and revive relation-based KD by making it overtake its IM counterparts.

**Learning with virtual knowledge.** While not a standalone research topic, learning with virtual knowledge finds relevance in a variety of learning-based problems. For instance, a commonly used paradigm in 3D vision tasks is to learn (or construct) from the raw data a virtual view or representation as auxiliary knowledge in solving the main task, rang-

(a) Increasing performance on training set.    (b) Plateaued performance on validation set.    (c) Diffusion of gradient perturbations within a batch.
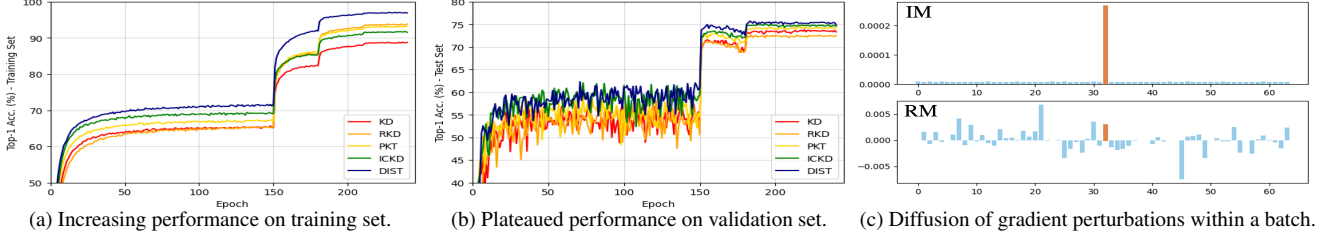
Figure 2. Pilot studies that reveal the overfitting and spurious gradient diffusion issues with RM-based KD.

ing from object reconstruction [5] and optical flow [2] to 3D semantic segmentation [36], monocular 3D object detection [13], and 3D GAN inversion [64]. More broadly, many data-efficient learning methods also share the spirit of utilising virtual knowledge. For instance, self-supervised learning methods generate virtual views of the unlabelled data to enable the learning of pretext tasks [11, 20]. Another popular paradigm is transformation-invariant representation learning [51, 57] in semi-supervised learning and domain adaptation. It enforces consistency between representations learnt for a raw sample and a virtual view of it. The virtual view is often obtained by applying semantic-preserving transformations to the raw sample [16, 17]. This work is more related to this later paradigm, but involves learning with virtual knowledge in a different context, via different approaches, and for a different problem.

## 3. Preliminaries

KD methods generally employ a cross-entropy (CE) loss and a distillation loss to supervise student learning. The CE loss is computed between student logits $\mathbf{z}_i^s$ for each sample and its ground-truth label $\mathbf{y}_i$. The distillation loss matches teacher and student outputs via a distance metric $\phi(\cdot)$. In vanilla KD [27], $\phi(\cdot)$ is the Kullback–Leibler divergence (KLD) between teacher logits $\mathbf{z}^t$ and student logits $\mathbf{z}^s$:

$$\mathcal{L}_i^{\text{KD}} = \phi_{\text{KLD}}(\mathbf{z}_i^s, \mathbf{z}_i^t)$$
$$= \tau^2 \sum_{j=1}^{C} \sigma_j(\mathbf{z}_i^t/\tau) \log \frac{\sigma_j(\mathbf{z}_i^t/\tau)}{\sigma_j(\mathbf{z}_i^s/\tau)}, \qquad (1)$$

where $\sigma(\cdot)$ is the Softmax operation with temperature parameter $\tau$, and $C$ is the number of classes. For feature-based methods, $\phi(\cdot)$ can be the mean squared error (MSE) between teacher and student feature maps for each instance [56], i.e., $\mathcal{L}_i^{\text{KD}} = \phi_{\text{MSE}}(\mathbf{f}_i^s, \mathbf{f}_i^t)$.

For relation-based methods, a relation encoding function $\psi(\cdot)$ first abstracts teacher or student outputs of all instances within a training batch into a relational representation, before applying $\phi(\cdot)$ to match these relational representations between the teacher and the student. For example, the KD objectives of DIST [29] take the general form of:

$$\mathcal{L}^{\text{KD}} = \phi(\psi(\mathbf{z}_1^s, \mathbf{z}_2^s, ..., \mathbf{z}_B^s), \psi(\mathbf{z}_1^t, \mathbf{z}_2^t, ..., \mathbf{z}_B^t)), \quad (2)$$

where $B$ is the batch size, and subscript $i$ of $\mathcal{L}^{\text{KD}}$ is dropped as the loss is computed for a batch of instances. DIST uses both inter-class and inter-sample relation encoders as $\psi(\cdot)$.

## 4. Pilot Studies

**Training dynamics of KD methods.** We examine the training dynamics of different relation-based methods on CIFAR-100 with ResNet32×4→ResNet8×4 as the teacher-student pair. In Figs. 2a and 2b, we immediately notice that relational methods achieve significantly higher training accuracy, but they only marginally lead or even fall short in test accuracy compared to IM-based KD [27]. We hypothesise that relational methods are more prone to overfitting. This is expected given that the optimality of IM matching (cond. $\mathcal{A}$) implies the optimality of relation matching (cond. $\mathcal{B}$), while the converse does not hold. Concretely, $\mathcal{A} \Rightarrow \mathcal{B} \land \neg\mathcal{B} \Rightarrow \neg\mathcal{A}$. In other words, relation matching is a weaker and less constrained objective than IM, which makes the student more readily fit the teaching signals and not generalise well. Thus, we conclude that *C1: relation matching methods are more prone to overfitting.*

**Gradient analysis of KD methods.** We investigate the gradient patterns within a batch when a spurious sample produces a major misguiding signal. To this end, we first generate two random vectors $\mathbf{x}, \mathbf{y} \sim \mathcal{N}(0, 1)$ for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{B \times D}$, where $B$ is the batch size and $D$ is the dimension of per-sample predictions, $\mathbf{x}$ is taken as the sample-wise predictions, and $\mathbf{y}$ the supervision signals. We then add a noise vector $\epsilon = c \cdot \mathbf{z}$ to $\mathbf{x}_t$, where $\mathbf{z} \sim \mathcal{N}(0, 1)$ and $c$ is a scaling factor. In this case, $\mathbf{x}_t + \epsilon$ becomes a spurious prediction within our batch. We compute the loss from $\mathbf{x}$ and $\mathbf{y}$ using either IM or RM objectives, and consider the change in sample-wise gradients $\mathbf{g}$ within the batch upon the injection of the spurious sample. Formally, we visualise:

$$\Delta\mathbf{g} = \left[\|\frac{\partial\mathcal{L}}{\partial(\mathbf{x}'_i)}\|_2 - \|\frac{\partial\mathcal{L}}{\partial(\mathbf{x}_i)}\|_2\right]_{i=1}^{B},$$
$$\text{s. t. } \mathbf{x}'_i = \mathbf{x}_i + \epsilon \cdot \mathbb{I}(i = t), \quad \mathcal{L} \in \{\mathcal{L}_{IM}, \mathcal{L}_{RM}\}. \quad (3)$$

In Fig. 2c ($B = 64$ and $t = 32$), when the IM objective is used, only the spurious sample receives a prominent gradient. Whereas for an RM objective, many other samples receive significant gradients as they are directly connected
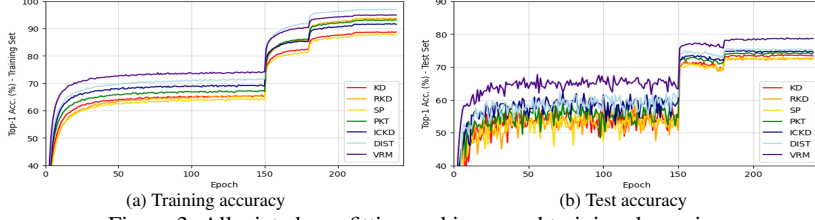
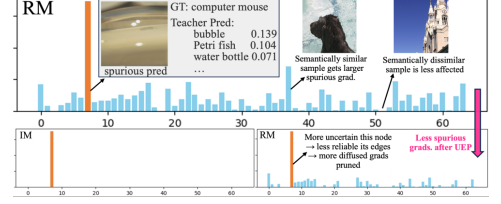Figure 3. Alleviated overfitting and improved training dynamics.



Figure 4. Alleviated spurious gradients.

to $\mathbf{x}_t$ in the computational graph of the RM loss. In other words, the spurious signals produced by one malign prediction will propagate to and affect all samples within a batch (in fact, those closer to $\mathbf{x}_t$ within the prediction manifold are more strongly affected). This means other sample-wise predictions will be significantly updated only to accommodate a malign prediction, even if they are already in relatively good shape. Through this investigation, we discover that *C2: relation matching methods are more prone to the adverse impact of spurious samples*. We also present this pilot analysis on ImageNet training in Fig. 4.

For *C1*, common approaches to combat overfitting include the incorporation of richer learning signals and regularisation, which for RM-based KD methods means richer relations constructed and transferred. For *C2*, an intuitive solution is to identify and suppress the effect of spurious predictions or relations or to slacken the matching criterion. Guided by these principles, we now proceed to formally build up our method step by step.

## 5. Method

### 5.1. Constructing Inter-Sample Relations

We first construct relation graph $\mathcal{G}^{IS}$ that encodes inter-sample affinity within a batch of sample predictions $\{\mathbf{z}_i\}_{i=1}^B$. Different from [52, 61], our relations are constructed from predicted logits which embed more compact categorical knowledge. We use the pairwise distance between instance-wise predictions within a batch as our measure of affinity. Existing methods leverage the Gram matrices [29, 53, 54, 61] to encode inter-sample relations, but we find that this leads to collapsed inter-class knowledge via the inner product operation. Instead, our pairwise distance preserves the inter-class knowledge along the secondary dimension (*i.e.*, the class dimension), which enables such information to be explicitly transferred as auxiliary knowledge alongside the matching of inter-sample relations.

Thus, we have constructed a dense relation graph $\mathcal{G}^{IS}$, which comprises $B$ vertices and $B \times B$ edges: $\mathcal{G}^{IS} = (\mathcal{V}^{IS}, \mathcal{E}^{IS})$. Each vertex in $\mathcal{G}^{IS}$ represents the prediction vector $\mathbf{z} \in \mathbb{R}^C$ for one instance within a batch, and is connected to all instances within the batch including itself. The attribute of edge $\mathcal{E}_{i,j}^{IS}$ connecting vertices $i$ and $j$ describes the class-wise relations between the predictions of instances $i$ and $j$. In practice, we can organise all edges into matrix

$\mathcal{E}^{IS} \in \mathbb{R}^{B \times B \times C}$, in which:

$$\mathcal{E}_{i,j}^{IS} = \frac{\mathbf{z}_i - \mathbf{z}_j}{\|\mathbf{z}_i - \mathbf{z}_j\|_2} \in \mathbb{R}^C \quad \text{for} \quad i, j \in [1, B], \quad (4)$$

where we empirically find that normalisation along the secondary dimension helps regularise relations and improves performance. Concretely, $\mathcal{E}$ encodes inter-sample class-wise relations within a batch of $B$ training samples. This design is different from and empirically more effective than previous relation formulations.

### 5.2. Constructing Inter-Class Relations

Although $\mathcal{G}^{IS}$ encodes rich inter-sample relations within a batch, it fails to explicitly model inter-class correlation patterns that are also beneficial structured knowledge [29]. Therefore, we propose to build a novel inter-class batch-wise relation graph $\mathcal{G}^{IC} = (\mathcal{V}^{IC}, \mathcal{E}^{IC})$, instead of the Gram matrices in the inner product space as in [29]. The construction of $\mathcal{G}^{IC}$ mirrors that of $\mathcal{G}^{IS}$. Vertices in $\mathcal{G}^{IC}$ are the class-wise logit vectors $\mathbf{w} \in \mathbb{R}^B$. Each edge in $\mathcal{E}^{IC} \in \mathbb{R}^{C \times C \times B}$ embeds the pairwise difference between the $i$-th and $j$-th per-class vectors. Concretely:

$$\mathcal{E}_{i,j}^{IC} = \frac{\mathbf{w}_i - \mathbf{w}_j}{\|\mathbf{w}_i - \mathbf{w}_j\|_2} \in \mathbb{R}^B \quad \text{for} \quad i, j \in [1, C]. \quad (5)$$

Our inter-class relations preserve the batch-wise discrepancies by treating them as a dimension of additional knowledge (reciprocal to the case of inter-sample relations), which is unlike any other previous methods [29]. We will demonstrate in Tab. 7 that our novel formulation of inter-sample and inter-class relations by preserving the raw affinity knowledge along the secondary dimension perform significantly better than previous relation encoders $\psi(\cdot)$ via Gram matrices [29, 52, 53, 61] or third-order angular distances [52].

### 5.3. Constructing Virtual Relations

For each prediction $\mathbf{z}_i$ within a batch $\{\mathbf{z}_i\}_{i=1}^B$, we create a virtual view of it, denoted as "$\tilde{\mathbf{z}}_i$", by applying semantic-preserving transformations to original image $\mathbf{x}_i$. While other transformations are applicable, we choose RandAugment [17] that applies stochastic image transformations (see Supplementary Material for details). With our batch of predictions augmented into $\{\mathbf{z}_i, \tilde{\mathbf{z}}_i\}_{i=1}^B$, we can construct a

| Teacher<br>Student | Venue | ResNet56<br>ResNet20 | ResNet32×4<br>ResNet8×4 | WRN-40-2<br>WRN-16-2 | WRN-40-2<br>WRN-40-1 | VGG13<br>VGG8 | ResNet32×4<br>ShuffleNetV2 | VGG13<br>MobileNetV2 | ResNet50<br>MobileNetV2 | WRN-40-2<br>ShuffleNetV1 |
|---|---|---|---|---|---|---|---|---|---|---|
| Teacher | | 72.34 | 79.42 | 75.61 | 75.61 | 74.64 | 79.42 | 74.64 | 79.34 | 75.61 |
| Student | | 69.06 | 72.50 | 73.26 | 71.98 | 70.36 | 71.82 | 64.60 | 64.60 | 70.50 |
| *Feature-based* | | | | | | | | | | |
| FitNets [56] | ICLR'15 | 69.21 | 73.50 | 73.58 | 72.24 | 71.02 | 73.54 | 64.16 | 63.16 | 73.73 |
| AT [74] | ICLR'17 | 70.55 | 73.44 | 74.08 | 72.77 | 71.43 | 72.73 | 59.40 | 58.58 | 73.32 |
| CRD [60] | ICLR'20 | 71.16 | 75.51 | 75.48 | 74.14 | 73.94 | 75.65 | 69.63 | 69.11 | 76.05 |
| SRRL [68] | ICLR'21 | 71.13 | 75.33 | 75.59 | 74.18 | 73.44 | - | - | - | - |
| PEFD [12] | NeurIPS'22 | 70.07 | 76.08 | 76.02 | 74.92 | 74.35 | - | - | - | - |
| TaT [42] | CVPR'22 | 71.59 | 75.89 | 76.06 | 74.97 | 74.39 | - | - | - | - |
| ReviewKD [10] | CVPR'21 | 71.89 | 75.63 | 76.12 | 75.09 | 74.84 | 77.78 | 70.37 | 69.89 | 77.14 |
| NORM [47] | ICLR'23 | 71.35 | 76.49 | 75.65 | 74.82 | 73.95 | 78.32 | 69.38 | 71.17 | 77.63 |
| FCFD [45] | ICLR'23 | 71.96 | 76.62 | 76.43 | 75.46 | 75.22 | 78.18 | 70.65 | 71.00 | 77.99 |
| *Logit-based* | | | | | | | | | | |
| KD [27] | arXiv'15 | 70.66 | 73.33 | 74.92 | 73.54 | 72.98 | 74.45 | 67.37 | 67.35 | 74.83 |
| TAKD [50] | AAAI'20 | 70.83 | 73.81 | 75.12 | 73.78 | 73.23 | 74.82 | 67.91 | 68.02 | 75.34 |
| CTKD [41] | AAAI'23 | 71.19 | 73.79 | 75.45 | 73.93 | 73.52 | 75.31 | 68.46 | 68.47 | 75.78 |
| NKD [72] | ICCV'23 | 70.40 | 76.35 | 75.24 | 74.07 | 74.86 | 76.26 | 70.22 | 70.76 | 75.96 |
| DKD [81] | CVPR'22 | 71.97 | 76.32 | 76.24 | 74.81 | 74.68 | 77.07 | 69.71 | 70.35 | 76.70 |
| LSKD [59] | CVPR'24 | 71.43 | 76.62 | 76.11 | 74.37 | 74.36 | 75.56 | 68.61 | 69.02 | - |
| TTM [82] | ICLR'24 | 71.83 | 76.17 | 76.23 | 74.32 | 74.33 | 76.55 | 69.16 | 69.59 | 75.42 |
| CRLD [76] | MM'24 | 72.10 | 77.60 | 76.45 | 75.58 | 75.27 | 78.27 | 70.39 | 71.36 | - |
| *Relation-based* | | | | | | | | | | |
| RKD [52] | CVPR'19 | 69.61 | 71.90 | 73.35 | 72.22 | 71.48 | 73.21 | 64.52 | 64.43 | 72.21 |
| CC [54] | CVPR'19 | 69.63 | 72.97 | 73.56 | 72.21 | 70.71 | 71.29 | 64.86 | 65.43 | 71.38 |
| SP [61] | ICCV'19 | 69.67 | 72.94 | 73.83 | 72.43 | 72.68 | 74.56 | 66.30 | 68.08 | 74.52 |
| ICKD [46] | ICCV'21 | 71.76 | 75.25 | 75.64 | 74.33 | 73.42 | - | - | - | - |
| DIST [29]† | NeurIPS'22 | 71.75 | 76.31 | - | 74.73 | - | 77.35 | 68.50 | 68.66 | 76.40 |
| **VRM** | - | **72.09** | **78.76** | **77.47** | **76.46** | **76.19** | **79.34** | **71.66** | **72.30** | **78.62** |

Table 1. Results for same- and different-model teacher-student pairs on CIFAR-100. †: using re-trained, stronger teachers.

larger inter-sample edge matrix $\mathcal{E}^{IS} \in \mathbb{R}^{2B \times 2B \times C}$ and a larger inter-class edge matrix $\mathcal{E}^{IC} \in \mathbb{R}^{C \times C \times 2B}$.

From the perspective of sample views, our new $\mathcal{E}^{IS}$ and $\mathcal{E}^{IC}$ constructed from a batch of real and virtual samples essentially capture three types of knowledge, namely relations amongst real views (denoted as "real-real"), relations amongst virtual views ("virtual-virtual"), and relations between pairs of real and virtual views ("real-virtual"). For instance, a real-virtual edge that connects real vertex $m$ and virtual vertex $n$ in $\mathcal{E}^{IS}$ is computed as:

$$\mathcal{E}^{IS}_{m,n} = \frac{\mathbf{z}_m - \tilde{\mathbf{z}}_n}{\|\mathbf{z}_m - \tilde{\mathbf{z}}_n\|_2} \in \mathbb{R}^C. \qquad (6)$$

### 5.4. Pruning into Sparse Graphs

**Pruning redundant edges.** The augmented $\mathcal{E}^{IS}$ in Sec. 5.3 contains $2B \times 2B$ edges, leading to quadrupled overheads. For better efficiency, we prune $\mathcal{G}^{IS}$ into sparse graphs. Noticing that $\mathcal{E}^{IS}$ is symmetric along its diagonals, we first prune its redundant half to save up to $50\%$ edge count. We also remove intra-view edges to get a further $50\%$ reduction, as we empirically find them redundant and harm knowledge transfer. For $\mathcal{G}^{IC}$, we decompose the augmented batch of predictions of size $2B$ into a real-view batch and a virtual-view batch, each of size $B$, and in lieu use the inter-sample batch-wise affinity vectors between them as its vertices. Compared to their original intra-view formulation of size $2B$ in Sec. 5.3, this design encodes purely inter-view

affinity knowledge with halved parameters. With redundant edge pruning (REP), our graphs now become sparse and the remaining edges can again be rearranged into compact matrices: $\mathcal{E}^{ISV} \in \mathbb{R}^{B \times B \times C}$ and $\mathcal{E}^{ICV} \in \mathbb{R}^{C \times C \times B}$.

**Pruning unreliable edges.** To mitigate the diffusive effect of spurious predictions discovered earlier, we further identify and prune the unreliable edges. In previous graph learning works, the absolute certainty of two vertices are often used to determine the reliability of an edge. For instance, REM [9] computes the reliability of an edge as the mean of the maximum predicted probabilities of two samples (*i.e.*, two vertices). However, we argue this could bias the learning towards easy samples. Instead, we measure the discrepancy between two predictions. The larger this discrepancy, the more unreliable the relation constructed. The unreliable edge pruning (UEP) criterion is given by $\mathcal{E}^{ISV}_{i,j} = \emptyset$ if $\mathrm{H}(\mathbf{z}^s_i, \tilde{\mathbf{z}}^s_j) > P_m$ where $\mathrm{H}(\cdot)$ computes the joint entropy (JE) between two predictions and $P_m$ is the $m$-th percentile within the batch. Note that the criterion is enforced on student predictions, resulting in adaptive and dynamic pruning as different edges get pruned in each iteration, which improves learning most of the time.

### 5.5. Full Objective

With $\mathcal{G}^{IS}$ and $\mathcal{G}^{IC}$ constructed for both teacher and student predictions, our VRM objective matches edge matrices $\mathcal{E}^{ISV}$ and $\mathcal{E}^{ICV}$ between teacher and student via distance

| Teacher<br>Student | Venue | ResNet34<br>ResNet18 | ResNet50<br>MobileNetV1 |
|---|---|---|---|
| Teacher | | 73.31/91.42 | 76.16/92.86 |
| Student | | 69.75/89.07 | 68.87/88.76 |
| *Feature-based* | | | |
| AT [74] | ICLR'17 | 70.69/90.01 | 69.56/89.33 |
| OFD [25] | ICCV'19 | 70.81/89.98 | 71.25/90.34 |
| CRD [60] | ICLR'20 | 71.17/90.13 | 71.37/90.41 |
| CAT-KD [23] | CVPR'23 | 71.26/90.45 | 72.24/91.13 |
| SimKD [8] | CVPR'22 | 71.59/90.48 | 72.25/90.86 |
| ReviewKD [10] | CVPR'21 | 71.61/90.51 | 72.56/91.00 |
| SRRL [68] | ICLR'21 | 71.73/90.60 | 72.49/90.92 |
| PEFD [12] | NeurIPS'22 | 71.94/90.68 | 73.16/91.24 |
| RSD [78] | ICCV'25 | 72.18/90.75 | 73.05/91.26 |
| FCFD [45] | ICLR'23 | 72.24/90.74 | 73.37/91.35 |
| *Logit-based* | | | |
| KD [27] | arXiv'15 | 70.66/89.88 | 68.58/88.98 |
| TAKD [50] | AAAI'20 | 70.78/90.16 | 70.82/90.01 |
| DKD [81] | CVPR'22 | 71.70/90.41 | 72.05/91.05 |
| SDD [63] | CVPR'24 | 71.14/90.05 | 72.24/90.71 |
| TTM [82] | ICLR'24 | 72.19/- | 73.09/- |
| LSKD [59] | CVPR'24 | 72.08/90.74 | 73.22/91.59 |
| CRLD [76] | MM'24 | 72.37/90.76 | 73.53/91.43 |
| *Relation-based* | | | |
| RKD [52] | CVPR'19 | 71.34/90.37 | 71.32/90.62 |
| CC [54] | CVPR'19 | 70.74/- | - |
| ICKD [46] | ICCV'21 | 72.19/90.72 | - |
| DIST [29] | NeurIPS'22 | 72.07/90.42 | 73.24/91.12 |
| **VRM** | - | **72.67/90.68** | **74.16/91.78** |

Table 2. Results on ImageNet.

| | ResNet101→ResNet18 | | | ResNet50→MobileNetV2 | | |
|---|---|---|---|---|---|---|
| | AP | $AP_{50}$ | $AP_{75}$ | AP | $AP_{50}$ | $AP_{75}$ |
| Teacher | 42.04 | 62.48 | 45.88 | 40.22 | 61.02 | 43.81 |
| Student | 33.26 | 53.61 | 35.26 | 29.47 | 48.87 | 30.90 |
| *Feature-based* | | | | | | |
| FitNets [56] | 34.43 | 54.16 | 36.71 | 30.20 | 49.80 | 31.69 |
| FGFI [62] | 35.44 | 55.51 | 38.17 | 31.16 | 50.68 | 32.92 |
| TAKD [50] | 34.59 | 55.35 | 37.12 | 31.26 | 51.03 | 33.46 |
| ReviewKD [10] | 36.75 | 56.72 | 34.00 | 33.71 | 53.15 | 36.13 |
| FCFD [45] | 37.37 | 57.60 | 40.34 | 34.97 | 55.04 | 37.51 |
| *Logit-based* | | | | | | |
| KD [27] | 33.97 | 54.66 | 36.62 | 30.13 | 50.28 | 31.35 |
| CTKD [41] | 34.56 | 55.43 | 36.91 | 31.39 | 52.34 | 33.10 |
| LSKD [59] | - | - | - | 31.74 | 52.77 | 33.40 |
| DKD [81] | 35.05 | 56.60 | 37.54 | 32.34 | 53.77 | 34.01 |
| *Relation-based* | | | | | | |
| **VRM** | **35.46** | **56.91** | **37.93** | **32.67** | **53.96** | **34.48** |

Table 3. Results on MS-COCO. All detectors are based on a Faster-RCNN [55] backbone.

| Teacher | Student | T. | S. | KD | AT | SP | LG | AutoKD | LSKD | **VRM** |
|---|---|---|---|---|---|---|---|---|---|---|
| ResNet56 | DeiT-T | 70.43 | 65.08 | 73.25 | 73.51 | 67.36 | 78.15 | <u>78.58</u> | 78.55 | **79.52** |
| ResNet56 | T2T-ViT-7 | 70.43 | 69.37 | 74.15 | 74.01 | 72.26 | 78.35 | <u>78.62</u> | 78.43 | **78.88** |
| ResNet56 | PiT-T | 70.43 | 73.58 | 75.47 | 76.03 | 74.97 | 78.48 | 78.51 | <u>78.76</u> | **79.25** |
| ResNet56 | PVT-T | 70.43 | 69.22 | 74.66 | 77.07 | 70.48 | 77.48 | 73.60 | <u>78.43</u> | **79.42** |

Table 4. Results for cross-architecture distillation on CIFAR-100.

metric $\phi(\cdot)$, for which the Huber loss is used. Formally, $L_{vrm}^{ISV} = \phi(\mathcal{E}_S^{ISV}, \mathcal{E}_T^{ISV})$ and $L_{vrm}^{ICV} = \phi(\mathcal{E}_S^{ICV}, \mathcal{E}_T^{ICV})$. The full objective is a weighted combination of the CE loss and the proposed VRM losses:

$$L_{total} = L_{ce} + \alpha L_{vrm}^{ISV} + \beta L_{vrm}^{ICV} \qquad (7)$$

where $L_{ce}$ is the CE loss applied to student's predictions of both real and virtual views and supervised by GT; $\alpha$ and $\beta$ are balancing scalars. A PyTorch-style pseudo-code for the computing of VRM losses is provided in Algorithm 1 in Supplementary Material.

# 6. Experiments

## 6.1. Experimental Settings

Our method is evaluated on CIFAR-100 [35] and ImageNet [19] datasets for image classification, and MS-COCO [43] for object detection. All experimental configurations follow the standard practice in prior works. More details are provided in Supplementary Material.

## 6.2. Main Results

**Results on CIFAR-100.** The results for different KD methods on CIFAR-100 are shown in Tab. 1. For *same-model KD*, (left of Tab. 1), VRM surpasses all previous relation-based methods across all teacher-student pairs by large margins, including DIST [29]. Noticeably, VRM significantly outperforms the strongest feature-based method, FCFD [45]. On average, it also performs much better than top logit-based methods such as CRLD [76] and TTM [82].

These are significant, given that IM methods are known to be naturally more adapt at distilling between homogeneous model pairs. For *different-model KD* (right of Tab. 1), which RM methods are supposed to be more competent with, VRM readily surpasses all RM and IM methods with notable margins. These results highlight VRM's versatility on both homogeneous and heterogeneous model pairs.

**Results on ImageNet.** VRM surpasses all feature- and relation-based methods on this large-scale dataset, as shown in Tab. 2. The advantage of VRM is again more apparent over the heterogeneous pair (*i.e.*, ResNet50 → MobileNetV2), whereby it for the first time hits 74.0% Top-1 accuracy. Notably, VRM outperforms strong competitors such as FCFD [45] and CRLD [76] with comparable or even less computational overheads as compared in Tab. 10.

**Results on MS-COCO.** We demonstrate that VRM generalises to more challenging tasks by adapting it to object detection. From Tab. 3, VRM performs better than existing logit-based methods and competitively to feature-based methods. VRM is slightly behind top-performing feature-based methods such as FCFD [45] and ReviewKD [10]. This is a consequence of the very nature of the object detection task, where fine-grained contextual features play a vital role, making feature-based methods inherently better-off [62]. Nonetheless, we experimentally demonstrate that VRM is effective on object detection, whereby it improves the baseline by 2–3% AP, surpasses all logit-based methods, and is on par with strong feature-based methods.

**Results for cross-architecture distillation.** We follow the

| Component | RN32×4 RN8×4 | VGG13 VGG8 |
|---|---|---|
| Baseline | 72.50 | 70.36 |
| + Match $\{\mathcal{E}^{IS}\}$ | 75.64 | 74.08 |
| + Match $\{\mathcal{E}^{IS}, \mathcal{E}^{IC}\}$ | 75.94 | 74.47 |
| + Match $\{\mathcal{E}^{ISV}\}$ | 77.63 | 75.30 |
| + Match $\{\mathcal{E}^{ISV}, \mathcal{E}^{ICV}\}$ | 78.38 | 75.56 |
| + Pruning | **78.76** | **76.19** |

Table 5. Ablation of major designs.

| Matching Choice | RN32×4 RN8×4 | VGG13 VGG8 | RN50 MNV2 |
|---|---|---|---|
| Match $\varnothing$ | 72.50 | 70.36 | 68.33 |
| Match $\{\mathcal{E}\}$ | **78.76** | **76.19** | **72.30** |
| Match $\{\mathcal{E}, \mathcal{V}\}$ | 78.63 | 75.60 | 70.97 |
| Match $\{\mathcal{V}\}$ | 78.46 | 75.77 | 68.33 |

Table 6. Effect of matching vertices.

| $\psi(\cdot)$ | Shape of Rel. Matrix | Method | RN32×4 RN8×4 | VGG13 VGG8 | RN50 MNV2 |
|---|---|---|---|---|---|
| Gram matrices | $[B,B]$ & $[C,C]$ | CC, SP, PKT, DIST | 77.31 | 75.12 | 70.28 |
| Angular relations | $[B,B,B]$ | RKD | 77.14 | 75.09 | 69.65 |
| ISV & ICV | $[B,B,C]$ & $[C,C,B]$ | VRM | **78.46** | **75.30** | **72.10** |

Table 7. Comparing different relation functions.

| $\alpha$ | 32 | 64 | 128 | 256 | 512 |
|---|---|---|---|---|---|
| Acc. | 78.03 | 78.13 | **78.76** | 78.32 | 77.84 |

| $\beta$ | 8 | 16 | 32 | 64 | 128 |
|---|---|---|---|---|---|
| Acc. | 78.40 | 78.51 | **78.76** | 78.86 | 78.60 |

| $n$ | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| Acc. | 77.92 | 78.38 | **78.76** | 78.55 | 78.43 | 78.50 |

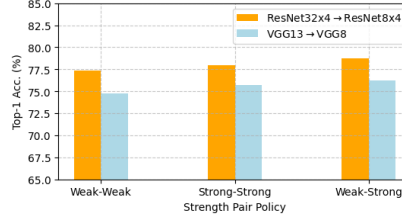Table 8. Effect of different $\alpha$, $\beta$, and $n$.
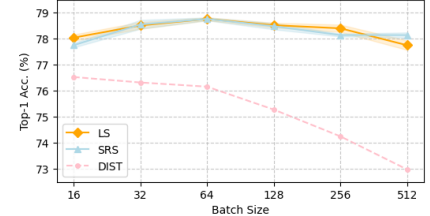


Figure 5. Effect of different difficulty pairs.



Figure 6. Robustness to varying $B$.

settings in [39] and [59] to perform CNN-to-ViT cross-architecture distillation. Tab. 4 shows that VRM sets new record under this set-up for different ViT architectures. Notably, by simply replacing the vanilla KD loss with the proposed VRM objective and making no other modifications, VRM improves DeiT-T's accuracy by *14.44%*, leading the relation-based SP [61] by *12.16%* and even surpassing AutoKD [40] with AutoML search.

## 6.3. Ablation Studies

**Ablations of main design choices.** In Tab. 5, starting from the baseline where only the CE loss is applied, we gradually incorporate each of our designs. As shown, each extra design consistently brings performance gains, which corroborate the validity of individual design choices.

**Effect of different relation encoding functions.** Tab. 7 compares our proposed relation encoding scheme with existing ones. To demonstrate the superiority of our formulation, we substitute it with existing Gram matrices [29, 53, 54, 61] or angle-wise relations [52]. Results show that our inter-sample class-wise and inter-class sample-wise relations consistently yield much better performance than existing relations, which highlights VRM's contribution of a novel and superior type of relation encoding in addition to the introduction of virtual relation.

**Effect of varying hyperparameters.** VRM involves three major hyperparameters: $\alpha$ and $\beta$, and $n$. According to Tab. 8, VRM works generally well with $\alpha$ and $\beta$ in a reasonable range. Larger $\alpha$ and $\beta$ may produce better results for certain model pairs but worse results for others. For $n$, a larger $n$ means more difficult images and larger inter-view prediction discrepancy. From Tab. 8, $n = 2$ works best, whereas other values also report competitive results.

**Effect of different real-virtual difficulty pairs.** We also experiment with different strength combinations for generating the real and virtual samples. The results are presented in Fig. 5, where "Weak" denotes the default transformation used in previous methods, and "Strong" denotes RandAugment with $n = 2$. Overall, we conclude that 1) a moderate discrepancy leads to optimal performance, and 2) the discrepancy is significant to the success of VRM.

**Robustness to varying batch sizes.** Relational methods are known to be sensitive to training batch size $B$. We conduct experiments to examine how robust VRM is against varying $B$. To adjust the learning rate accordingly, we consider two LR scaling rules: linear LR scaling [21] and square root LR scaling [11]. From Fig. 6, VRM yields competitive results across a wide range of $B$ values. In contrast, DIST [29] degrades significantly as $B$ grows.

## 6.4. Further Analysis

**Analysis of training dynamics.** Figs. 3a and 3b plot the per-epoch training and validation set accuracies throughout training. VRM maintains a all-time lead in both training and validation performance, with faster convergence. Besides, while VRM's lead in training performance tapers off towards the end of training, it remains more substantial, if not further enlarging, in validation performance, which highlights its superior generalisation properties. This analysis also shows our designs are effective in mitigating the issues with existing RM methods identified in the pilot studies.

**Analysis of spurious gradients.** We revisit our second pilot study by analysing how spurious gradients are suppressed in real ImageNet training. From Fig. 4, similar gradient diffusion patterns are observed in real distillation. Samples semantically more similar to the sample with spurious predictions are more affected by spurious gradients (*e.g.*, the dog image with a watery background resembles the bubbly texture in another sample that is actually a computer mouse.).
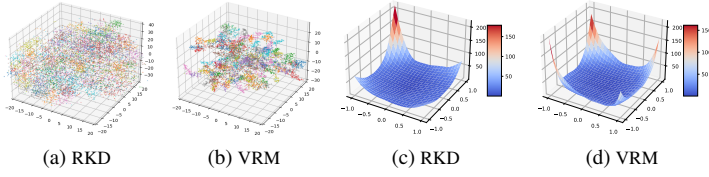
| | | (a) RKD | (b) VRM | (c) RKD | (d) VRM |

Figure 7. t-SNE (a–b) and loss landscape (c–d) visualisations for RN8×4 students distilled with an RN32×4 teacher via RKD and VRM.

Table 9. Effect of different transformation strength pairs on ImageNet.

| Real | Virtual | RN50→MNV2 | RN34→RN18 |
|------|---------|-----------|-----------|
| Default | Default | **74.16**/**91.78** | **72.67**/90.68 |
| Default | RandAug(n=0) | 74.10/91.70 | 72.60/**90.88** |
| Default | RandAug(n=1) | 74.13/91.63 | 72.59/90.82 |
| Default | RandAug(n=2) | 74.13/91.68 | 72.48/90.73 |

| Method | KD | FitNets | RKD | ICKD | DIST | CRD | ReviewKD |
|--------|-----|---------|------|------|------|------|----------|
| Train. Time (ms) | 24.9 | 26.7 | 31.0 | 28.0 | 27.2 | 41.2 | 39.9 |
| Peak GPU Mem. (MB) | 323 | 330 | 330 | 381 | 330 | 1418 | 1042 |
| Method | SDD | MLLD | CRLD | NORM | PEFD | FCFD | **VRM** |
| Train. Time (ms) | 34.2 | 57.2 | 43.2 | 35.1 | 36.2 | 56.4 | **47.2** |
| Peak GPU Mem. (MB) | 690 | 576 | 551 | 1806 | 701 | 953 | **579** |

Table 10. Training efficiency of different distillation methods.

| Pruning | RN32×4→RN8×4 | | | V13→V8 | W40-2→SNV2 |
|---------|------|-------------|-----------|--------|------------|
| | Acc. | $t_{batch}$ (s) | Mem. (MB) | Acc. | Acc. |
| w/o | 78.38 | 61.4 | 34.3 | 75.56 | 78.45 |
| + REP | 78.46 | **46.3** | 13.6 | 75.30 | 78.41 |
| + UEP | **78.76** | 47.2 | **13.1** | **76.19** | **78.62** |

Table 11. Effectiveness and efficiency of pruning operations.

**Visualisation of embedding space.** We conduct t-SNE analysis on student penultimate layer embeddings learnt via different methods. As presented in Fig. 7a and 7b, VRM leads to more compact per-class clusters with clearer inter-class separation and less stray points. These imply better student features distilled by VRM for the downstream task.

**Visualisation of loss landscape.** We further analyse the generalisation and convergence properties of our method through the lens of visualised loss landscape [38]. From Fig. 7c and 7d, compared to RKD, VRM has a wider, flatter, and deeper region of minima – typical hint of better model generalisation and robustness; this wide convexity basin surrounded by salient pikes in various directions indicates excellent convergence properties. More visualisations are provided in Supplementary Material.

**Vertex matching.** Within the constructed graphs $\mathcal{G}^{IS}$ and $\mathcal{G}^{IC}$, VRM matches edges $\mathcal{E}^{ISV}$ and $\mathcal{E}^{ICV}$ which carry relational knowledge. By extension, we can naturally expect vertices $\mathcal{V}$ to be also transferred. From Tab. 6, matching vertices is not as effective. The advantage of matching relations (*i.e.*, edges) is more pronounced for heterogeneous KD pairs such as ResNet50→MobileNetV2. Adding vertex matching to the proposed edge matching does not improve but instead degrade performance. We argue this is because introducing vertex matching objectives make the matching criterion more stringent which is against our motivation of using more slackened matching.

**Role of transformation operations.** We emphasise the strong performance of VRM is not a simple outcome of the transformations involved. This can be verified by the results in Fig. 5 and Tab. 9, where using only the default weak transformation (*i.e.*, random crop and horizontal flip) for both views achieves highly competitive and even superior performance under VRM. In fact, in the case of "Weak-Weak", both views still differ because of the stochastic operations used. Essentially, *the success of VRM lies exactly in the discrepancy of teacher and student predictions*, where regularisation comes into crucial play. On smaller and easier datasets such as CIFAR-100, the gap between teacher and student predictions are relatively small, and manually enlarging this gap via extra transformations leads to promising outcomes. Whereas the more difficult ImageNet data already sees large teacher-student prediction discrepancy, such that it benefits less from further amplification.

**Training efficiency.** From Tab. 10, VRM is reasonably efficient compared to existing algorithms, including some leading feature- [12, 45, 47] and logit-based [33, 63] methods. VRM does not introduce any extra overheads at inference. We also breakdown the proposed pruning designs in Tab. 11 to understand how they individually impact the performance and efficiency of VRM. We see REP boosts efficiency at minimal performance cost, whereas UEP trades minimal efficiency cost for performance.

**More experiments, analyses, and discussions** are provided in Supplementary Material.

# 7. Conclusion

This paper presents VRM, a novel knowledge distillation framework that constructs and transfers virtual relations. Our designs are motivated by a set of pilot experiments, from which we identified two main cruxes with existing relation-based KD methods: their tendency to overfit and susceptibility to adverse gradient propagation. A series of tailored designs are developed and are shown to successfully mitigate these issues. We have conducted extensive experiments on different tasks and multiple datasets and verified VRM's validity and superiority in diverse settings, whereby it consistently achieves state-of-the-art performance. We hope that this work could renew the community's interest in relation-based knowledge distillation, and encourage more systematic reassessment of the design principles of such solutions.

# References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D. Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *CVPR*, 2019. 2, 1, 3

[2] Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia. Learning optical flow from still images. In *CVPR*, 2021. 3

[3] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. *NeurIPS*, 2014. 6

[4] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. *arXiv preprint arXiv:2106.04732*, 2021. 6

[5] Joao Carreira, Abhishek Kar, Shubham Tulsiani, and Jitendra Malik. Virtual view networks for object reconstruction. In *CVPR*, 2015. 3

[6] Jiahao Chang, Shuo Wang, Hai-Ming Xu, Zehui Chen, Chenhongyi Yang, and Feng Zhao. Detrdistill: A universal knowledge distillation framework for detr-families. In *ICCV*, 2023. 1

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

[8] Defang Chen, Jian-Ping Mei, Hailin Zhang, Can Wang, Yan Feng, and Chun Chen. Knowledge distillation with the reused teacher classifier. In *CVPR*, 2022. 6

[9] Peibin Chen, Tao Ma, Xu Qin, Weidi Xu, and Shuchang Zhou. Data-efficient semi-supervised learning by reliable edge mining. In *CVPR*, 2020. 5

[10] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *CVPR*, 2021. 5, 6, 1, 2, 3

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. 2020. 3, 7, 6

[12] Yudong Chen, Sen Wang, Jiajun Liu, Xuwei Xu, Frank de Hoog, and Zi Huang. Improved feature distillation via projector ensemble. In *NeurIPS*, 2022. 5, 6, 8, 1, 2

[13] Yi-Nan Chen, Hang Dai, and Yong Ding. Pseudo-stereo for monocular 3d object detection in autonomous driving. In *CVPR*, 2022. 3

[14] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Bevdistill: Cross-modal bev distillation for multi-view 3d object detection. In *ICLR*, 2023. 1

[15] Zhiyu Chong, Xinzhu Ma, Hong Zhang, Yuxin Yue, Haojie Li, Zhihui Wang, and Wanli Ouyang. Monodistill: Learning spatial features for monocular 3d object detection. *ICLR*, 2022. 2

[16] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019. 3

[17] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *NeurIPS*, 2020. 3, 4

[18] Kaiwen Cui, Yingchen Yu, Fangneng Zhan, Shengcai Liao, Shijian Lu, and Eric P Xing. Kd-dlgan: Data limited image generation via knowledge distillation. In *CVPR*, 2023. 1

[19] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6, 2

[20] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 3

[21] P Goyal. Accurate, large minibatch sgd: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 7

[22] Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *ICLR*, 2024. 1

[23] Ziyao Guo, Haonan Yan, Hui Li, and Xiaodong Lin. Class attention transfer based knowledge distillation. In *CVPR*, 2023. 2, 6, 1, 3

[24] Zhiwei Hao, Jianyuan Guo, Kai Han, Yehui Tang, Han Hu, Yunhe Wang, and Chang Xu. One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation. *NeurIPS*, 2024. 2

[25] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *ICCV*, 2019. 2, 6, 1, 3

[26] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *AAAI*, 2019. 1, 2, 3

[27] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowlegde in a neural network. In *arXiv:1503.02531*, 2015. 1, 2, 3, 5, 6, 4

[28] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. In *arXiv:1704.04861*, 2017. 3

[29] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. In *NeurIPS*, 2022. 2, 3, 4, 5, 6, 7, 1

[30] Yuge Huang, Jiaxiang Wu, Xingkun Xu, and Shouhong Ding. Evaluation-oriented knowledge distillation for deep face recognition. In *CVPR*, 2022. 1

[31] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. In *arXiv:1707.01219*, 2017. 1, 2, 3

[32] Sujin Jang, Dae Ung Jo, Sung Ju Hwang, Dongwook Lee, and Daehyun Ji. Stxd: structural and temporal cross-modal distillation for multi-view 3d object detection. *NeurIPS*, 2024. 2

[33] Ying Jin, Jiaqi Wang, and Dahua Lin. Multi-level logit distillation. In *CVPR*, 2023. 2, 8

[34] Minha Kim, Shahroz Tariq, and Simon S Woo. Fretal: Generalizing deepfake detection using knowledge distillation and representation learning. In *CVPR*, 2021. 1

[35] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 6, 2

[36] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3d semantic segmentation. In *ECCV*, 2020. 3

[37] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 6

[38] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *NeurIPS*, 2018. 8

[39] Kehan Li, Runyi Yu, Zhennan Wang, Li Yuan, Guoli Song, and Jie Chen. Locality guidance for improving vision transformers on tiny datasets. In *ECCV*, 2022. 7, 1, 3

[40] Lujun Li, Peijie Dong, Zimian Wei, and Ya Yang. Automated knowledge distillation via monte carlo tree search. In *ICCV*, 2023. 7, 1, 3

[41] Zheng Li, Xiang Li, Lingfeng Yang, Borui Zhao, Renjie Song, Lei Luo, Jun Li, and Jian Yang. Curriculum temperature for knowledge distillation. In *AAAI*, 2023. 2, 5, 6, 1, 3

[42] Sihao Lin, Hongwei Xie, Bing Wang, Kaicheng Yu, Xiaojun Chang, Xiaodan Liang, and Gang Wang. Knowledge distillation via the target-aware transformer. In *CVPR*, 2022. 2, 5, 1

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 6, 2

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

[45] Dongyang Liu, Meina Kan, Shiguang Shan, and Xilin Chen. Function-consistent feature distillation. In *ICLR*, 2023. 1, 5, 6, 8, 2, 3

[46] Li Liu, Qingle Huang, Sihao Lin, Hongwei Xie, Bing Wang, Xiaojun Chang, and Xiaodan Liang. Exploring inter-channel correlation for diversity-preserved knowledge distillation. In *ICCV*, 2021. 2, 5, 6, 1

[47] Xiaolong Liu, Lujun Li, Chao Li, and Anbang Yao. Norm: Knowledge distillation via n-to-one representation matching. In *ICLR*, 2023. 1, 5, 8, 2, 3

[48] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. Knowledge distillation via instance relationship graph. In *CVPR*, 2019. 2

[49] Yahui Liu, Enver Sangineto, Wei Bi, Nicu Sebe, Bruno Lepri, and Marco Nadai. Efficient training of visual transformers with small datasets. *NeurIPS*, 2021. 3

[50] Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *AAAI*, 2020. 2, 5, 6, 1, 3

[51] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020. 3

[52] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *IEEE TMM*, 2019. 1, 2, 4, 5, 6, 7, 3

[53] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *ECCV*, 2018. 2, 4, 7, 1, 3, 5

[54] Baoyun Peng, Xiao Jin, Jiaheng Liu, Shunfeng Zhou, Yichao Wu, Yu Liu, Dongsheng Li, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *CVPR*, 2019. 2, 4, 5, 6, 7, 1, 3

[55] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE T-PAMI*, 39(6):1137–1149, 2016. 6

[56] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015. 1, 2, 3, 5, 6

[57] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020. 3

[58] Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. Densely guided knowledge distillation using multiple teacher assistants. In *ICCV*, 2021. 2

[59] Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation. In *CVPR*, 2024. 2, 5, 6, 7, 1, 3

[60] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *ICLR*, 2020. 2, 5, 6, 1, 3

[61] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *ICCV*, 2019. 1, 2, 4, 5, 7, 3

[62] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *CVPR*, 2019. 1, 6, 3

[63] Shicai Wei, Chunbo Luo, and Yang Luo. Scale decoupled distillation. In *CVPR*, 2024. 6, 8, 1, 3

[64] Jiaxin Xie, Hao Ouyang, Jingtan Piao, Chenyang Lei, and Qifeng Chen. High-fidelity 3d gan inversion by pseudo-multi-view optimization. In *CVPR*, 2023. 1, 3

[65] Guodong Xu, Ziwei Liu, Xiaoxiao Li, and Chen Change Loy. Knowledge distillation meets self-supervision. In *ECCV*, 2020. 6

[66] Chuanguang Yang, Zhulin An, Linhang Cai, and Yongjun Xu. Hierarchical self-supervised augmented knowledge distillation. In *IJCAI*, 2021. 6

[67] Chuanguang Yang, Helong Zhou, Zhulin An, Xue Jiang, Yongjun Xu, and Qian Zhang. Cross-image relational knowledge distillation for semantic segmentation. In *CVPR*, 2022. 1, 2

[68] Jing Yang, Brais Martinez, Adrian Bulat, and Georgios Tzimiropoulos. Knowledge distillation via softmax regression representation learning. In *ICLR*, 2021. 5, 6, 1, 2

[69] Longrong Yang, Xianpan Zhou, Xuewei Li, Liang Qiao, Zheyang Li, Ziwei Yang, Gaoang Wang, and Xi Li. Bridging cross-task protocol inconsistency for distillation in dense object detection. In *ICCV*, 2023. 1

[70] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *CVPR*, 2022. 1

[71] Zhendong Yang, Zhe Li, Mingqi Shao, Dachuan Shi, Zehuan Yuan, and Chun Yuan. Masked generative distillation. In *ECCV*, 2022. 1, 3

[72] Zhendong Yang, Ailing Zeng, Zhe Li, Tianke Zhang, Chun Yuan, and Yu Li. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. In *ICCV*, 2023. 2, 5, 1, 3

[73] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Km. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *CVPR*, 2017. 2, 1

[74] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017. 2, 5, 6, 1, 3

[75] Bo Zhang, Jiacheng Sui, and Li Niu. Foreground object search by distilling composite image feature. In *ICCV*, 2023. 1

[76] Weijia Zhang, Dongnan Liu, Weidong Cai, and Chao Ma. Cross-view consistency regularisation for knowledge distillation. In *ACM MM*, 2024. 1, 5, 6, 2, 3

[77] Weijia Zhang, Dongnan Liu, Chao Ma, and Weidong Cai. Alleviating foreground sparsity for semi-supervised monocular 3d object detection. In *WACV*, 2024. 1

[78] Weijia Zhang, Yuehao Liu, Wu Ran, and Chao Ma. Cross-architecture distillation made simple with redundancy suppression. *ICCV*, 2025. 2, 6, 1

[79] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *CVPR*, 2018. 3

[80] Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. Deep mutual learning. In *CVPR*, 2018. 1, 2, 3

[81] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *CVPR*, 2022. 2, 5, 6, 1, 3, 4

[82] Kaixiang Zheng and En-Hui Yang. Knowledge distillation based on transformed teaching matching. In *ICLR*, 2024. 2, 5, 6, 1, 3

[83] Shengchao Zhou, Weizhou Liu, Chen Hu, Shuchang Zhou, and Chao Ma. Unidistill: A universal cross-modality knowledge distillation framework for 3d object detection in bird's-eye view. In *CVPR*, 2023. 1