DCCD: Reducing Neural Network Redundancy via Distillation

Yuang Liu, Jun Chen, and Yong Liu, *Member, IEEE*

***Abstract*— Deep neural models have achieved remarkable performance on various supervised and unsupervised learning tasks, but it is a challenge to deploy these large-size networks on resource-limited devices. As a representative type of model compression and acceleration methods, knowledge distillation (KD) solves this problem by transferring knowledge from heavy teachers to lightweight students. However, most distillation methods focus on imitating the responses of teacher networks but ignore the information redundancy of student networks. In this article, we propose a novel distillation framework difference-based channel contrastive distillation (DCCD), which introduces channel contrastive knowledge and dynamic difference knowledge into student networks for redundancy reduction. At the feature level, we construct an efficient contrastive objective that broadens student networks' feature expression space and preserves richer information in the feature extraction stage. At the final output level, more detailed knowledge is extracted from teacher networks by making a difference between multiview augmented responses of the same instance. We enhance student networks to be more sensitive to minor dynamic changes. With the improvement of two aspects of DCCD, the student network gains contrastive and difference knowledge and reduces its overfitting and redundancy. Finally, we achieve surprising results that the student approaches and even outperforms the teacher in test accuracy on CIFAR-100. We reduce the top-1 error to 28.16% on ImageNet classification and 24.15% for cross-model transfer with ResNet-18. Empirical experiments and ablation studies on popular datasets show that our proposed method can achieve state-of-the-art accuracy compared with other distillation methods.**

***Index Terms*— Contrastive learning, deep compression, deep learning, knowledge distillation (KD).**

I. Introduction

NOWADAYS, powerful neural networks have become the main driver of development in many fields. In a neural network, more parameters usually result in better performance. With the help of many remarkable techniques, including residual connections and batch normalization, neural networks with thousands of layers can be effectively trained. However, it's difficult to deploy these large-scale deep models on resource-limited embedded systems. Several techniques have been proposed to address this issue, such as low-rank factorization [1], [2], parameter and filters pruning [3], [4], model quantization [5], [6], [7] and knowledge distillation (KD).

Hinton et al. [8] first defined KD and set up a distillation framework with a teacher-student pair as its primary structure. They used ***softmax*** operation and high temperature to effectively extract "dark knowledge" of the high-performance teacher network. After that, many methods [9], [10] focused on improving the response-based distillation from different aspects. Other methods [11], [12] learned richer knowledge from the teacher's intermediate layers. They proposed various knowledge transfers between the hint layers and the guided layers. Besides, relation-based distillation methods [13], [14] further explored the relationships between different layers and image instances. Information flow, correlation congruence, and other deep-seated information were introduced into distillation training. Recently, Tian et al. [15], Xu et al. [16], and Chen et al. [17] combined KD with contrastive learning. They maximize the lower bound of mutual information between the teacher and student representations by contrastive objectives for better performance.

Although previous studies have obtained many excellent results, few methods focus on the redundancy of the student network itself. The student network can gain helpful knowledge from the teacher network, but this may exacerbate its overfitting and parameter redundancy. Some works [18], [19], [20], [21], [22] address this problem by combining distillation and network pruning. Network adjustment [19] uses dropout to measure redundancy and shows that KD assists the performance of the pruned network. InDistill [18] leverages channel pruning properties to reduce the capacity gap between the models and retain the architectural alignment. These pruning with distillation methods reduce the redundancy of the network, but the student network structure needs to be changed.

Therefore, we propose our novel distillation method difference-based channel contrastive distillation (DCCD), which reduces the redundancy of the student network on two levels: At the feature level, we propose channel contrastive distillation (CCD), which constrains the student network channels to imitate the teacher network's corresponding channels and

Manuscript received 14 April 2022; revised 1 November 2022; accepted 16 January 2023. This work was supported by the Key Research and Development Project of Zhejiang Province under Grant 2022C03126 and Grant 2021C01035. *(Yuang Liu and Jun Chen contributed equally to this work.) (Corresponding author: Yong Liu.)*

Yuang Liu and Jun Chen are with the State Key Laboratory of Industrial Control Technology and the Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China (e-mail: yuangliu@zju.edu.cn; junc@zju.edu.cn).

Yong Liu is with the State Key Laboratory of Industrial Control Technology and the Institute of Cyber-systems and Control, Zhejiang University, Hangzhou 310027, China, and also with the Huzhou Institute, Zhejiang University, Huzhou 313000, China (e-mail: yongliu@iipc.zju.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2023.3238337.

Digital Object Identifier 10.1109/TNNLS.2023.3238337

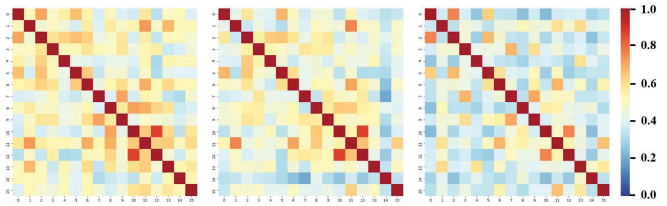


Fig. 1. Heatmaps of cosine similarities between the penultimate layer's outputs in one minibatch data. Cooler color represents a weaker correlation. Our method makes more significant differences between the network responses of different instances. (a) Baseline. (b) KD. (c) Ours.

keep the response distance between non-corresponding channels. The student network obtains a broader expression space and accommodates more knowledge than the baseline model. The representation redundancy between the student network channels is reduced. At the final output level, we supplement the original KD with difference KD (DKD), which extracts the dynamic dark knowledge from the response difference between multiview augmented instances. We add the student network's attention to minor changes in all categories during distillation. Softer distribution loss leads to better regularization for parameter redundancy reduction. With the combination of CCD and DKD, our whole method successfully reduces both representation and parameter redundancy of student networks as Fig. 1 shows. In summary, we make three main contributions as follows.

1) We propose a novel contrastive distillation method channel contrastive distillation, which constructs contrastive objectives between corresponding and non-corresponding channels. We reduce the overlap of feature responses in the student network's channel expression space. Compared with other contrastive-based distillation methods (CCDs), we do not have to maintain a memory buffer for negative samples and use fewer computing resources.
2) We supplement the traditional KD with a difference KD. By focusing on the difference between responses of multiview augmented instances, we extract difference knowledge from the randomness of data augmentation. Minor dynamic changes in all categories are well highlighted, and the student network's expression sensitivity is promoted during difference distillation training.
3) We conduct targeted experiments to investigate the performance of our method in terms of redundancy reduction, data augmentation, hyper-parameter sensitivity, and so on. Extensive empirical experiments and ablation studies show the effectiveness of DCCD on various datasets (CIFAR-100, ImageNet-1K, STL10) for improving the student network's performance.

II. Related Work

Because our framework contains a supplement (DKD) to the original KD and a novel channel-level CCD, we discuss related works in KD and contrastive learning in this section.

A. Knowledge Distillation

After Hinton et al. [8] put forward the concept of KD, many studies have enriched its meaning and methodology. To bridge the gap between the student and the teacher, Mirzadeh et al. [23] proposed teacher assistants and multi-step KD. Malinin et al. [24] distilled the distribution of the predictions from an ensemble of teachers rather than only one teacher. Many researchers have introduced the intermediate representations of teacher networks. Fitnet [11] directly matched the feature activation of the teacher and the student and formed a feature-based distillation method for the first time. Inspired by this, many papers [12], [25], [26] proposed their methods to extract and match the intermediate layers for information transfer. Heo et al. [27] provided a very detailed summary of feature-based distillation methods and proposed their well-designed scheme. Besides, FSP [13] calculated and distilled the information flow of teacher networks based on the Gramian matrices. CCKD [14] focused on the correlation between instances in a minibatch. There are some other approaches [28], [29], [30] that study the deep-level relationships in the network.

Lots of methods [9], [10], [31] have improved the original KD from different aspects. Our method DKD focuses on the response differences caused by the data augmentation randomness. We make a difference between the multiview responses of the same instance and achieve a softer distribution to learn about dark knowledge in all output categories effectively.

B. Contrastive Learning

Instead of learning a signal from individual data samples one at a time, contrastive learning learns by comparing among different samples. Nowadays, contrastive learning methods [32], [33], [34], [35] have become one of the most popular self-supervised approaches. MoCo [36] built a dynamic dictionary with a queue and a moving-averaged encoder for large batches of negative samples. SimCLR [37] set augmented views of other items in a minibatch as negative samples and introduced stronger data augmentation for accuracy improvement. Zbontar et al. [38] allowed the use of very high-dimensional output vectors and didn't require large batches nor asymmetry between the network twins such as a predictor network, gradient stopping, or a moving average on the weight updates.

Some methods have made excellent progress in combining KD and contrastive learning. CRD [15] introduces NCE-based algorithms and a memory buffer for storing negative samples in distillation. Self-supervised for knowledge distillation (SSKD) [16] transfers the hidden information from the teacher to the student by exploiting the similarity between self-supervision signals as an auxiliary task. WCoRD [17] utilizes the dual and primal forms of the Wasserstein distance for global contrastive learning and local feature distribution matching. A novel channel-level contrastive distillation method CCD is proposed in our method. We reduce the response redundancy of the student network channels through the CCD loss, which measures the cross correlation matrix between the student's and teacher's channel features and makes it as close to the identity matrix as possible.

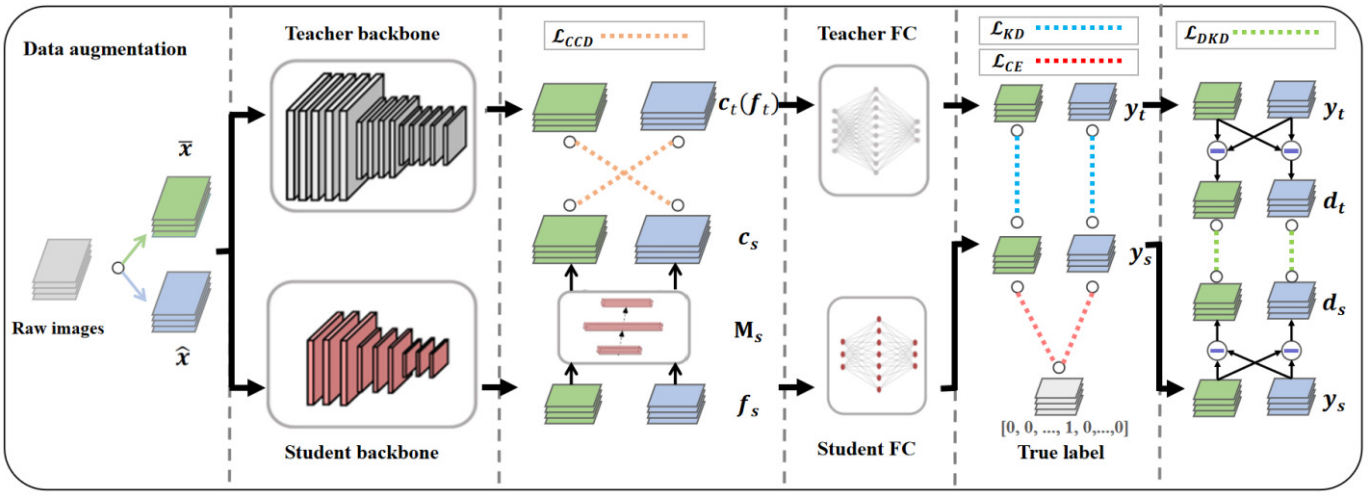


Fig. 2. Overall framework of our proposed method. We introduce CCD loss and DKD loss into the teacher-student distillation structure and take advantage of the randomness of data augmentation. The abbreviations identified in the figure are meticulously described in the proposed method section.

III. Proposed Method

In this section, we introduce the principle of channel contrastive distillation and difference KD. The whole framework of DCCD is shown in Fig. 2.

A. Notations

Our distillation framework consists of a well-performed teacher network **T** and a lightweight student network **S** like the traditional KD, and we use $\{*\}_t$ and $\{*\}_s$ to express the feature maps, outputs, modules, and so on corresponding to the teacher and student network. We note one minibatch (included b pictures) as raw input $\boldsymbol{x}$, and the corresponding ground truth is noted as $\boldsymbol{y}$. The penultimate layer's outputs are extracted for contrastive learning, and we represent them as $\boldsymbol{f}_t$ and $\boldsymbol{f}_s$. The logit representations (before the softmax layer) of teacher and student note as $\boldsymbol{y}_t$ and $\boldsymbol{y}_s$. In particular, we use data augmentation to get twice augmented inputs $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$. They are similar in general but different in detail because of the randomness of data augmentation. We add the same mark symbol to the corresponding penultimate layer outputs ($\hat{\boldsymbol{f}}_t$, $\hat{\boldsymbol{f}}_s$ and $\bar{\boldsymbol{f}}_t$, $\bar{\boldsymbol{f}}_s$) and final logit representations ($\hat{\boldsymbol{y}}_t$, $\hat{\boldsymbol{y}}_s$ and $\bar{\boldsymbol{y}}_t$, $\bar{\boldsymbol{y}}_s$). It should be noted that our method remains consistent with the traditional distillation method on data augmentation settings, which ensures that all improvements come from our well-designed optimization loss rather than additional data processing.

B. Channel Contrastive Distillation

Single imitation of the features of the teacher network has been proven effective in many previous feature-based methods [12], [27]. To match the feature dimension, module $\mathbf{M}_t$ and $\mathbf{M}_s$, respectively are used to transform the feature $\boldsymbol{f}_t$ and $\boldsymbol{f}_s$. A distance function $\boldsymbol{\psi}$ measures the distance between the transformed feature, and the feature-based distillation loss function is generalized as follows:

$$\mathcal{L}_{\text{feature}} = \boldsymbol{\psi}\left(\mathbf{M}_s\left(\boldsymbol{f}_s\right), \mathbf{M}_t\left(\boldsymbol{f}_t\right)\right). \tag{1}$$

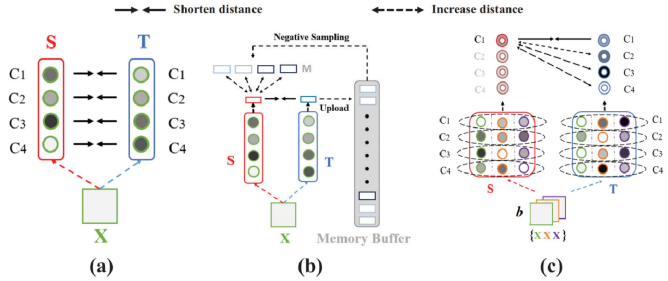


Fig. 3. Graphical illustration of different distillation methods (**X**: input instances; **S**: student network; **T**: teacher network; $\boldsymbol{c}_i$: the ith channel of the teacher or student network). (a) Feature-based distillation. (b) Contrastive response distillation. (c) Channel contrastive distillation.

As shown in Fig. 3(a), traditional feature-based distillation sets channels as basic units and uses only the imitation loss term to reduce the distance between the teacher and the student. However, contrastive response distillation [15] maximizes the mutual information between the student responses and the teacher responses by noise contrastive estimation (NCE). They use a neural network **G** as *critic* to estimate whether a pair comes from the joint distribution ($\eta = 1$) or the marginals ($\eta = 0$). The distribution $\boldsymbol{r}$ conditioned on η that captures whether the pair is congruent ($\boldsymbol{r}((\eta = 1))$) or incongruent ($\boldsymbol{r}((\eta = 0))$). The parameters of student network **S** and critic **G** can be optimized jointly by maximizing by the following equation:

$$\begin{aligned}\mathcal{L}_{\text{CRD}} = \mathbb{E}_{\boldsymbol{r}(\boldsymbol{f}_s, \boldsymbol{f}_t|\eta=1)}\left[\log\left(\mathbf{G}\left(\boldsymbol{f}_s, \boldsymbol{f}_t\right)\right)\right] \\ + k\mathbb{E}_{\boldsymbol{r}(\boldsymbol{f}_s, \boldsymbol{f}_t|\eta=0)}\left[\log\left(1 - \mathbf{G}\left(\boldsymbol{f}_s, \boldsymbol{f}_t\right)\right)\right].\end{aligned} \tag{2}$$

For the stability of the training process, multiple incongruent pairs (larger negative samples number k) are chosen, and a large memory buffer for storing negative samples needs to be maintained and kept up to date during distillation training. As Fig. 3(b) shows, CRD constructs the contrastive loss using instances as the basic units, which shortens the response distance between the teacher-student pair for the corresponding instances and increases it for non-corresponding instances.

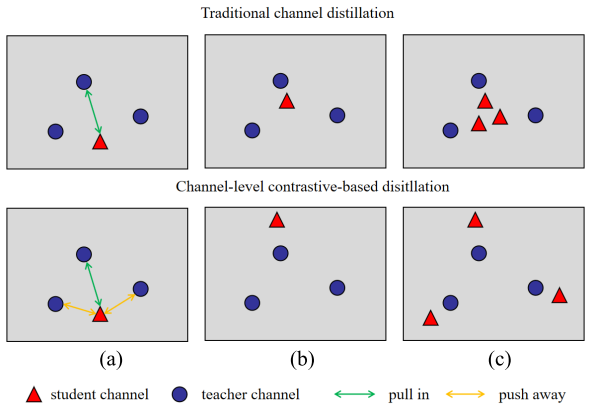


Fig. 4. Effect comparison between traditional channel distillation methods and our channel-level contrastive-based method. (a) Before training. (b) After training. (c) Overall view.

Motivated by both feature-based distillation and CRD, we construct a novel contrastive loss at the channel level as shown in Fig. 3(c). channel contrastive distillation constrains the student network channels (1) mimic the response of the corresponding channels of the teacher network (imitation term) and (2) contrast with other non-corresponding teacher channels to reduce the redundancy information between channel representations (contrastive term). The specific implementation process is as follows.

To match the feature dimension, we also use $\mathbf{M}_s$ to transform the student feature $\boldsymbol{f}_s$ to $\boldsymbol{c}_s$, but simply normalize $\boldsymbol{f}_t$ to $\boldsymbol{c}_t$ to maintain the effectiveness and stability of the teacher network's channel features. $\boldsymbol{c_s}$ and $\boldsymbol{c_t}$ have the same shape $b \times d$ (b represents the number of instances in one minibatch; d represents the number of dimensions for teacher feature expression)

$$\begin{aligned} \mathrm{Norm}(\boldsymbol{x}) &= \frac{\boldsymbol{x} - \mathrm{mean}(\boldsymbol{x})}{\mathrm{std}(\boldsymbol{x})} \\ \boldsymbol{c}_s &= \mathrm{Norm}\big(\mathbf{M}_s\big(\boldsymbol{f}_s\big)\big) \\ \boldsymbol{c}_t &= \mathrm{Norm}\big(\boldsymbol{f}_t\big). \end{aligned} \tag{3}$$

We introduce $\mathcal{L}_{\mathrm{eye}}$ into training as shown in (4), which makes the similarity matrix of the student and teacher channel expressions $\boldsymbol{c}_s$ and $\boldsymbol{c}_t$ as close to the identity matrix as possible [$(\cdot)^t$ represents the transpose operation for input matrix]. As shown in Fig. 4, the imitation loss term shortens the distance between the student channels and the corresponding teacher channels. It makes the feature response of the student network effective enough for the basic target. The contrastive loss term pulls apart the feature expression between the student channels and the other non-corresponding teacher channels, which reduces the overlap of feature meanings between student channels. The student network can contain less redundant information about instances and form a broader feature expression space. The parameter θ controls the learning ratio between the imitation term and the contrastive term

$$\begin{aligned} &\mathcal{L}_{\mathrm{eye}}(\boldsymbol{c}_s, \boldsymbol{c}_t, \theta) \\ &= \underbrace{\sum_i \Big(1 - \big(\boldsymbol{c}_s^i\big)^t \times \boldsymbol{c}_t^i\Big)^2}_{\text{imitation term}} + \underbrace{\frac{\theta}{d-1} \sum_i \sum_{j \neq i} \Big(\big(\boldsymbol{c}_s^i\big)^t \times \boldsymbol{c}_t^j\Big)^2}_{\text{contrastive term}}. \end{aligned} \tag{4}$$

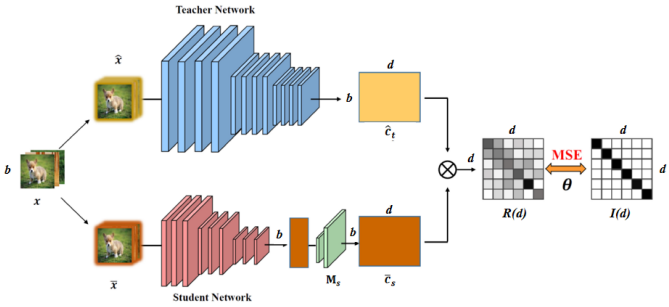


Fig. 5. Diagram of channel contrastive distillation. We process b row image $\boldsymbol{x}$ into two groups augmented data $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$, and calculate the relationship matrix $\boldsymbol{R}(d)$ by the transform module $\mathbf{M}_s$, eventually form our $\mathcal{L}_{\mathrm{eye}}$ compared with the identity matrix $\boldsymbol{I}(d)$. For clarity, only half of the content of CCD is drawn: $\mathcal{L}_{\mathrm{eye}}(\bar{\boldsymbol{c}}_s, \hat{\boldsymbol{c}}_t, \theta)$.

Through the adoption of $\mathcal{L}_{\mathrm{eye}}$ on the student network $\mathbf{S}$ alone: $\mathcal{L}_{\mathrm{eye}}(\boldsymbol{c}_s, \boldsymbol{c}_s, \theta)$, each student network channel may obtain more unique feature by the contrastive term, this will cause the collapse problem at the same time, which is extensively studied in contrastive learning. To address this issue, most contrastive learning methods use larger batch size and richer data augmentation during self-supervised learning. However, our method uses the teacher network $\mathbf{T}$ with fixed parameters as the contrastive template to ensure the stability of contrastive distillation training. Besides, augmented input pair $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ are used as parallel contrastive inputs, representing two views of the same instances. We compose contrastive loss with augmented channel expression ($\hat{\boldsymbol{c}}_s$ and $\bar{\boldsymbol{c}}_t$, $\bar{\boldsymbol{c}}_s$ and $\hat{\boldsymbol{c}}_t$). The student network can obtain a more stable and efficient training process by taking advantage of the randomness of data augmentation. Fig. 5 shows half of the CCD's general process, and the final CCD loss is shown in the following equation:

$$\mathcal{L}_{\mathrm{CCD}} = \mathcal{L}_{\mathrm{eye}}\big(\hat{\boldsymbol{c}}_s, \bar{\boldsymbol{c}}_t, \theta\big) + \mathcal{L}_{\mathrm{eye}}\big(\bar{\boldsymbol{c}}_s, \hat{\boldsymbol{c}}_t, \theta\big). \tag{5}$$

Previous works such as CRD [15] and WCoRD [17] build contrastive loss at the instance level and have achieved excellent results in exploring instance relationships. SSKD [16] also uses multiple data augmentations on the same image to mine richer teacher network knowledge, but it still aligns the expression relationships of different rotation angles between teacher and student networks at the instance level. Compared with these methods, CCD uses channels rather than instances as contrastive objective units. Our method reduces information redundancy inside network channels by distinguishing the feature meanings expressed by different channels. The teacher network provides effective both imitation and contrastive channel targets for the student's channels, so CCD does not need to maintain a large memory buffer for sampling negatives from other batches' responses. Our method optimizes the student network through only one minibatch instance, as same as the traditional KD algorithm. Therefore, it consumes fewer computing resources and enables a simpler training process.

C. Difference KD

Traditional KD loss as shown in (6) introduces temperature τ to form a softer probability distribution $\boldsymbol{p}_s$ and $\boldsymbol{p}_t$ on

both the student network outputs $\boldsymbol{y}_s$ and the teacher network outputs $\boldsymbol{y}_t$, and lets the student mimic the teacher's behavior by adding a strong congruent constraint on predictions using KL divergence

$$\boldsymbol{p}_s^i = \frac{\exp\left(\boldsymbol{y}_s^i/\tau\right)}{\sum_j^n \exp\left(\boldsymbol{y}_s^j/\tau\right)}$$

$$\boldsymbol{p}_t^i = \frac{\exp\left(\boldsymbol{y}_t^i/\tau\right)}{\sum_j^n \exp\left(\boldsymbol{y}_t^j/\tau\right)}$$

$$\mathcal{L}_{\mathrm{KD}} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{KL}\left(\boldsymbol{p}_s^i, \boldsymbol{p}_t^i\right). \tag{6}$$

According to the article [8], temperature τ is used to balance the learning attention ratio between the correct category and other categories. Higher temperature means paying more balanced attention to all categories, which may bring more benefits from "dark knowledge." We propose a new approach to extract more detailed "dark knowledge" from the teacher network $\mathbf{T}$ by making the differences between multiview outputs of the same instance with data augmentation: $\hat{\boldsymbol{d}} = \hat{\boldsymbol{y}} - \bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{d}} = \bar{\boldsymbol{y}} - \hat{\boldsymbol{y}}$.

Our difference KD shortens the difference distributions distance between the teacher and the student, which represents their responses to detailed changes caused by the randomness of data augmentation

$$\hat{\boldsymbol{q}}^i = \frac{\exp\left(\hat{\boldsymbol{d}}^i/\tau\right)}{\sum_j^n \exp\left(\hat{\boldsymbol{d}}^j/\tau\right)}$$

$$\bar{\boldsymbol{q}}^i = \frac{\exp\left(\bar{\boldsymbol{d}}^i/\tau\right)}{\sum_j^n \exp\left(\bar{\boldsymbol{d}}^j/\tau\right)}$$

$$\mathcal{L}_{\mathrm{DKD}} = \frac{1}{2n}\sum_{i=1}^{n}\left(\mathrm{KL}\left(\hat{\boldsymbol{q}}_s^i, \hat{\boldsymbol{q}}_t^i\right) + \mathrm{KL}\left(\bar{\boldsymbol{q}}_s^i, \bar{\boldsymbol{q}}_t^i\right)\right). \tag{7}$$

Because the teacher network and the student network have the essential ability for the target classification task, output $\boldsymbol{y}_s$ and $\boldsymbol{y}_t$ both have a sharp distribution (nearly one for the correct category and zero for other categories) after operation ***softmax*** at the end of distillation training, as Fig. 6(a) and (b) shows. Therefore, the negative categories occupy only a tiny proportion, and their value fluctuations near the small magnitude response have little effect on the distillation loss. $\mathcal{L}_{\mathrm{KD}}$ naturally ignore the relatively minor changes in negative categories' output. At the same time, the network outputs $\hat{\boldsymbol{y}}$ and $\bar{\boldsymbol{y}}$ (for augmented inputs $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ from the same raw instances $\boldsymbol{x}$) roughly keep the overall similarity and the detailed differences. The difference $\hat{\boldsymbol{y}} - \bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{y}} - \hat{\boldsymbol{y}}$ fully represent the response change between the student network and the teacher network in all categories. Their distributions are smoother and easier to display response changes caused by the randomness of data augmentation even after ***softmax*** as Fig. 6(c) shows.

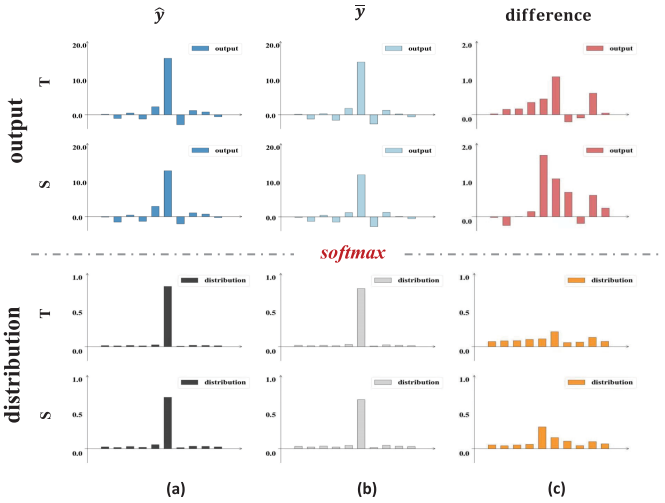


Fig. 6. Output and Distribution of all categories in CIFAR-10 classification for traditional KD versus DKD. (a) Teacher output, student output, teacher distribution, and student distribution in original KD for $\hat{y}$. (b) Same meaning as (a) for $\bar{y}$. (c) DKD calculates the difference output $\hat{y} - \bar{y}$ for the teacher and the student, softer difference distribution is shown.

For traditional KD, Hinton et al. [8] proved that it matches the logits between the teacher model and the student model

$$\frac{\partial \mathcal{L}_{\mathrm{KD}}}{\partial \boldsymbol{y}_s^i} = \frac{1}{\tau}\left(\boldsymbol{p}_s^i - \boldsymbol{p}_t^i\right) = \frac{1}{\tau}\left(\frac{\exp\left(\boldsymbol{y}_s^i/\tau\right)}{\sum_j^n \exp\left(\boldsymbol{y}_s^j/\tau\right)} - \frac{\exp\left(\boldsymbol{y}_t^i/\tau\right)}{\sum_j^n \exp\left(\boldsymbol{y}_t^j/\tau\right)}\right). \tag{8}$$

If the temperature τ is much higher than the magnitude of logits, (8) can be approximated according to its Taylor series

$$\frac{\partial \mathcal{L}_{\mathrm{KD}}}{\partial \boldsymbol{y}_s^i} = \frac{1}{\tau}\left(\frac{1 + \boldsymbol{y}_s^i/\tau}{n + \sum_j^n \boldsymbol{y}_s^j/\tau} - \frac{1 + \boldsymbol{y}_t^i/\tau}{n + \sum_j^n \boldsymbol{y}_t^j/\tau}\right). \tag{9}$$

If it is further assumed that the logits of each instance are zero-mean (i.e., $\sum_j^n y_s^j = \sum_j^n y_t^j = 0$), (9) can be simplified as follows:

$$\frac{\partial \mathcal{L}_{\mathrm{KD}}}{\partial \boldsymbol{y}_s^i} = \frac{1}{n\tau^2}\left(\boldsymbol{y}_s^i - \boldsymbol{y}_t^i\right). \tag{10}$$

Therefore, the traditional KD loss is equal to matching the logits between the student and the teacher under two conditions: (a) High temperature. (b) Zero-mean logits.

For our difference KD loss, it is natural to know that $\hat{\boldsymbol{d}} = -\bar{\boldsymbol{d}}$, so we can write the corresponding function as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\mathrm{DKD}}}{\partial \hat{d}_s^i} &= \frac{1}{2}\left(\left(\hat{\boldsymbol{q}}_s^i - \hat{\boldsymbol{q}}_t^i\right) + \left(\bar{\boldsymbol{q}}_s^i - \bar{\boldsymbol{q}}_t^i\right)\frac{\partial \hat{\boldsymbol{d}}_s^i}{\partial \bar{\boldsymbol{d}}_s^i}\right) \\ &= \frac{1}{2}\left(\left(\hat{\boldsymbol{q}}_s^i - \hat{\boldsymbol{q}}_t^i\right) - \left(\bar{\boldsymbol{q}}_s^i - \bar{\boldsymbol{q}}_t^i\right)\right) \\ &= \frac{1}{2}\left[\left(\frac{\exp\left(\hat{\boldsymbol{d}}_s^i/\tau\right)}{\sum_j^n \exp\left(\hat{\boldsymbol{d}}_s^j/\tau\right)} - \frac{\exp\left(\hat{\boldsymbol{d}}_t^i/\tau\right)}{\sum_j^n \exp\left(\hat{\boldsymbol{d}}_t^j/\tau\right)}\right)\right. \\ &\qquad \left. - \left(\frac{\exp\left(\bar{\boldsymbol{d}}_s^i/\tau\right)}{\sum_j^n \exp\left(\bar{\boldsymbol{d}}_s^j/\tau\right)} - \frac{\exp\left(\bar{\boldsymbol{d}}_t^i/\tau\right)}{\sum_j^n \exp\left(\bar{\boldsymbol{d}}_t^j/\tau\right)}\right)\right].\end{aligned} \tag{11}$$

TABLE I

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART DISTILLATION METHODS ON CIFAR-100. WE EXPERIMENT WITH THE TEACHER-STUDENT PAIRS USING THE SAME AND DIFFERENT NETWORK ARCHITECTURES. WE USE AUTHOR-PROVIDED OR AUTHOR-VERIFIED CODE AND RESULTS FROM CRD, SSKD AND WCORD REPOSITORIES FOR ALL OTHER METHODS. * INDICATES THAT OUR METHOD USE THE SAME ADDITIONAL ROTATION AUGMENTATION AS SSKD. OUR REPORTED RESULTS ARE AVERAGED OVER FIVE RUNS

Teacher **Student**	**WRN-40-2** **WRN-16-2**	**resnet56** **resnet20**	**resnet32x4** **resnet8x4**	**vgg13** **vgg8**	**vgg13** **MobileNetV2**	**ResNet50** **vgg8**	**resnet32x4** **ShuffleNetV2**
Teacher	75.61	72.34	79.42	74.64	74.64	79.34	79.42
Student	73.26	69.06	72.50	70.36	64.60	70.36	71.82
KD	74.92	70.66	73.33	72.98	67.37	73.81	74.45
FitNet	75.12	69.21	74.66	73.22	66.90	73.24	75.15
AT	75.32	70.55	74.53	73.48	65.13	74.01	75.39
SP	74.98	69.67	74.02	73.49	68.41	73.52	74.88
CCKD	75.09	69.63	74.21	73.04	68.02	73.48	74.71
VID	75.14	70.38	74.56	73.19	68.27	73.46	74.85
RKD	74.89	69.61	73.79	72.97	67.87	73.51	74.55
PKT	75.33	70.34	74.23	73.25	68.13	73.61	74.66
AB	70.27	69.47	74.40	73.35	68.23	73.65	74.99
FT	75.15	69.84	74.62	73.44	66.99	72.98	75.06
NST	74.67	69.60	74.28	73.33	63.77	71.74	75.24
CRD	75.64	71.63	75.46	74.29	69.94	74.58	76.05
WCoRD	76.11	71.92	76.15	74.72	70.02	74.68	76.48
SSKD	76.04	71.49	76.20	75.33	**71.53**	75.76	78.61
DCCD	76.56	**72.35**	76.57	74.90	70.01	75.71	77.41
DCCD*	**76.60**	71.72	**76.65**	**75.67**	71.44	**75.88**	**79.29**

Because the logit responses of augmented instances are similar on the whole and different in the details mentioned above, the magnitudes of the response differences are close to zero, and their distributions are approximately zero-mean. Equation (11) can be easy to meet the above two conditions even without high temperature

$$\frac{\partial \mathcal{L}_{\mathrm{DKD}}}{\partial \hat{\boldsymbol{d}}_s^i} = \frac{1}{2n\tau^2}\left(\left(\hat{\boldsymbol{d}}_s^i - \hat{\boldsymbol{d}}_t^i\right) - \left(\bar{\boldsymbol{d}}_s^i - \bar{\boldsymbol{d}}_t^i\right)\right) = \frac{1}{n\tau^2}\left(\hat{\boldsymbol{d}}_s^i - \hat{\boldsymbol{d}}_t^i\right). \tag{12}$$

In the same way as the above proof, we can obtain by the following equation:

$$\frac{\partial \mathcal{L}_{\mathrm{DKD}}}{\partial \bar{\boldsymbol{d}}_s^i} = \frac{1}{n\tau^2}\left(\bar{\boldsymbol{d}}_s^i - \bar{\boldsymbol{d}}_t^i\right). \tag{13}$$

Therefore, difference KD can effectively shorten the distances between the response differences of the student network and the teacher network. We can transfer the dynamic difference knowledge of the teacher network to the student network, which is reflected in the response changes for the randomness of data augmentation. By supplementing traditional $\mathcal{L}_{\mathrm{KD}}$ with $\mathcal{L}_{\mathrm{DKD}}$, we strengthen the sensitivity of the student network and reduce its parameter overfitting.

In the end, the student network is then trained by optimizing the following loss function:

$$\mathcal{L}_{\mathrm{student}} = \mathcal{L}_{\mathrm{CE}} + \alpha(\mathcal{L}_{\mathrm{KD}} + \mathcal{L}_{\mathrm{DKD}}) + \beta\mathcal{L}_{\mathrm{CCD}}. \tag{14}$$

IV. EXPERIMENT

We evaluate the proposed DCCD framework on various KD tasks: model compression on classification tasks and cross-model transfer. Extensive experiments and analyses are conducted to delve into our proposed method.

A. Model Compression

1) Experiments on CIFAR-100: CIFAR-100 [39] is the dataset that most KD methods use to validate their performance. CIFAR-100 contains 50 000 training images and 10 000 test images. We select many excellent KD methods (similarity-preserving knowledge distillation (SP) [40]; variational information distillation (VID) [41]; residual knowledge distillation (RKD) [9]; probabilistic knowledge transfer (PKT) [42]) to evaluate the performances of our method, and consider two scenarios: 1) the student and the teacher share the same network architecture and 2) different network architectures are used.

ResNet [43], visual geometry group (VGG) [44], MobileNet [45], and ShuffleNet [46] are chosen as teacher and student architectures. We run a total of 240 epochs for all methods. Temperature τ is set as 4. For our method, we set α as 1.0 for $\mathcal{L}_{\mathrm{KD}}$ and $\mathcal{L}_{\mathrm{DKD}}$. We keep the $\mathcal{L}_{\mathrm{CCD}}$ roughly equal in value and similar in proportion to other losses in different experiments. The number of teacher network's channels may be 64, 128, 256, and so on, accordingly, we set β as 0.4, 0.2, and 0.1. As for other hyper-parameters of baselines, we follow the setting of CRD [15]. Besides, for a fair comparison of the effectiveness of our method and SSKD, we introduce the same rotation data augmentation as SSKD and mark the corresponding experimental results with *.

Table I compares the top-1 accuracy of different distillation methods on CIFAR-100 dataset. Because the final response

TABLE II

TOP-1 AND TOP-5 ERROR RATES (%) OF DISTILLATION METHODS ON IMAGENET. RESULTS ARE AVERAGED OVER FIVE RUNS WITH TEACHER RESNET-34 AND STUDENT RESNET-18. * INDICATES THAT OUR METHOD USE THE SAME ADDITIONAL ROTATION AUGMENTATION AS SSKD

	Teacher	Student	KD	AT	SP	CCKD	CRD (+KD)	WCoRD (+KD)	SSKD	CCD	CCD+KD	DKD+KD	DCCD	DCCD*
Top-1	26.69	30.25	29.34	29.30	29.38	30.04	28.83 (28.62)	28.51 (28.44)	28.38	28.44	28.37	28.60	28.16	28.05
Top-5	8.58	10.93	10.12	10.00	10.20	10.83	9.87 (9.51)	9.84 (9.45)	9.33	9.70	9.47	9.81	9.44	9.12

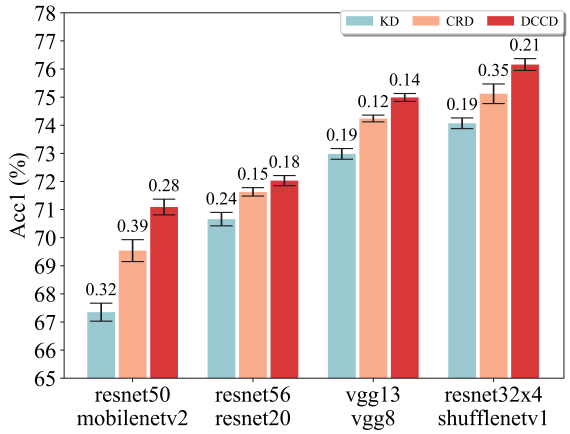


Fig. 7. Performance for KD, CRD, and our DCCD to transfer across various teacher and student networks on CIFAR-100. The standard deviations of the different random seed results are marked in the figure.

TABLE III

PERFORMANCE FOR RESNET110 → RESNET32 DISTILLATION WITH DIFFERENT LOSS CONFIGURATIONS ON CIFAR-100

Student	KD	CCD		DKD	Acc1(%)
		imitation	contrastive		
resnet32					71.14
resnet32	✓				73.08
resnet32		✓			73.63
resnet32			✓		73.45
resnet32		✓	✓		73.85
resnet32	✓	✓	✓		73.98
resnet32	✓			✓	73.82
resnet32	✓	✓	✓	✓	74.21

knowledge to an instance can be simply and effectively transferred through the original KD, all methods are combined with the original KD to obtain better results. In both scenarios (student and teacher are in the same or different network architecture style), contrastive-based methods (CRD, WCoRD, SSKD, and DCCD) outperform other distillation frameworks. For comparison between CCDs, our method DCCD achieved better performance than CRD and WCoRD, but slightly worse than SSKD in some teacher-student pairs. SSKD transfers richer instance contrastive information on the one hand, more importantly, applies random rotation data augmentation on the other hand. Under the same data augmentation setting, our method DCCD* can achieve higher accuracy on CIFAR-100.

2) Experiments on ImageNet: We also conduct experiments on the challenging ImageNet [47] dataset. We use ResNet-34 as the teacher and ResNet-18 as the student, and start with a learning rate of 0.1, divide it by 10 at 30, 60, and 90 epochs, and terminate training at 100 epochs. The results are shown in Table II. CCD alone achieves the same level of results as CRD,SSKD and WCoRD, and CCD + KD gets further improvement. DKD + KD improves the original KD with more detailed knowledge and reduces its error rate significantly. Finally, our whole method DCCD achieves 28.16% of top-1 error with ResNet-18. With rotational data augmentation, DCCD* further improves accuracy by 0.11%.

3) Ablation Study: We report results using different loss configurations for resnet110 → resnet32 on CIFAR-100 in Table III. It can be seen that the imitation term and the contrastive term of $\mathcal{L}_{\text{CCD}}$ both improve the performance of the student network, and the student network achieves better performance while combining them. DKD facilitates this 0.72% improvement for traditional KD by increasing the final response sensitivity of the student network. Our whole framework DCCD combines the advantages of CCD and DKD and achieves 74.21% top-1 test accuracy. The same effect can also be verified on ImageNet as shown in Table II.

In addition, we experiment with four network sets of teacher-student pairs on CIFAR-100 under different random seeds. The averages and the standard deviations of test top-1 accuracy with different distillation methods (original KD, CRD, and our method DCCD) are labeled in Fig. 7. Our method is proven more effective than traditional KD and CRD on average. It has about the same or even smaller standard deviation compared to the other two methods.

B. Cross-Model Transfer

1) Tiny-ImageNet → STL10: Tiny-ImageNet dataset is a miniature of ImageNet classification challenge. STL10 dataset [48] is an image recognition dataset for developing unsupervised feature learning, deep learning, and self-taught learning algorithms. In particular, each class has fewer labeled training examples than in CIFAR-100, but a very large set of unlabeled examples is provided to learn image models prior to supervised training. Transferring knowledge from Tiny-ImageNet to STL10 should help improve the performance of the model trained on STL10. Following WCoRD, we first map images in the RGB space to the Lab color space (L: Luminance, ab: Chrominance), then train teacher L-Net ResNet-18 on the Luminance dimension of Labeled Tiny-ImageNet. The student ab-Net ResNet-18 is distilled on the Chrominance dimension of the unlabeled STL-10 dataset with different objective functions. Finally, we train a linear classification module on top of features extracted from different layers in the student network for ten-category classification.

TABLE IV

PERFORMANCE FOR AB-NET RESNET-18 DISTILLED FROM SAME ARCHITECTURE L-NET TEACHER, WHICH IS TRAINED WITH THE LUMINANCE VIEW OF TINY-IMAGENET

Layer	1	2	3	4
CRD	**55.00**	63.64	73.76	74.75
WCoRD	54.60	**63.70**	74.23	75.43
CCD	54.82	63.54	**74.55**	**75.85**

TABLE V

PERFORMANCE FOR DIFFERENT TEACHER AND STUDENT TRANSFORM STRUCTURES, WE USE TEACHER RESNET56 AND STUDENT RESNET20 ON DATASET CIFAR-100

Teacher transform	**Student transform**	**Acc** (%)
Identity	Single linear layer	72.06
Identity	Multiple linear layers	72.35
Multiple linear layers	Single linear layer	71.56
Multiple linear layers	Multiple linear layers	71.85

TABLE VI

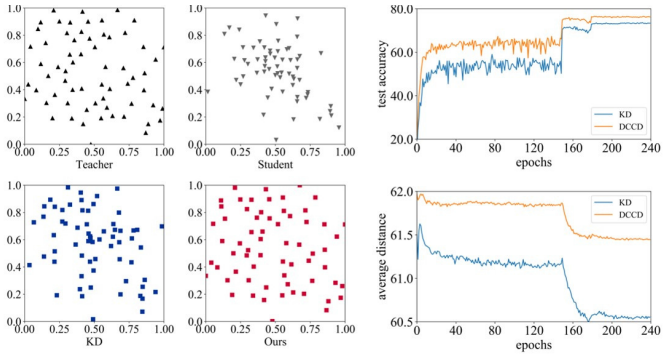
PERFORMANCE FOR VGG13 → VGG8 DISTILLATION ON CIFAR-100 WITH DIFFERENT DATA AUGMENTATION SETTINGS

Acc (%)	**teacher**	**student**	**KD**	**DCCD**
norm	74.64	70.36	72.98	74.90
mixup	76.04	70.91	73.98	75.75
randaugment	75.46	71.68	74.24	75.58
randerasing	75.89	71.37	73.85	75.51

In experiments, we compare test accuracy between three contrastive-based methods CRD, WCoRD, and CCD. Table IV shows the results on features extracted from different layers. Our method outperforms CRD and WCoRD when training linear classification on third and fourth residual blocks. Deeper neural networks can accommodate richer feature expression, and CCD loss can reduce its feature redundancy for better performance.

C. Further Analysis

1) Visualization Results: To show the improvement of our method for reducing feature redundancy, we performed a visual analysis of channel features. As shown in Fig. 8(a), we perform a uniform T-SNE (t-distributed stochastic neighbor embedding) dimensionality reduction visualization of channel features obtained from different networks. The baseline student network's channels are more aggregated and overlapping than the teacher network. With the original KD, the student network mimics the teacher network to obtain a similar distribution of channel feature meanings, but some channels still express overlapping meanings, as shown in the bottom left part. In contrast, our method combines imitation and contrastive terms, so the student network can obtain a more decentralized and meaningful representation of channel features.



(a) (b)

Fig. 8. (a) T-SNE visualization of teacher and student networks' features. We experiment with resnet20 → resnet14 on CIFAR-100 classification, the distribution of network's features after dimensionality reduction is shown in the figure. (b) Top-1 accuracy (%) for CIFAR-100 classification and Mean distance between network channels' feature responses during res32 × 4 → resnet8 × 4 distillation training.

To further demonstrate the usefulness of our method for reducing the redundancy of channel features, we compute the geometric median of the network channels during distillation training and record the average distance of all channels from the geometric median as Fig. 8(b) shows. Our method maintains a larger average distance during distillation training and keeps less information redundancy between the student network channels.

Finally, the relationship matrix between the responses in one minibatch is painted in different degrees of color according to cosine similarity in Fig. 1. Because the responses are presented after **ReLU** layer, they are non-negative, and the range of cosine similarity $\in [0, 1]$. Cooler color means that the feature expression distance between two instances is farther away. Our method DCCD broadens the channel feature expression space and increases the response sensitivity of the student network. Therefore the entire relationship matrix is bluer than the baseline and the original KD models. This shows that our method enables the student network to reduce its own channel feature overlap and obtain more differential expressions between instances.

2) Transform Module Structure: In our method, student transform module $\mathbf{M}_s$ is used as knowledge transfer between the teacher and the student. We explore the influence of transform module structure on accuracy like overhaul [27] do. Results are shown in Table V. We note that the uniqueness of feature expression in teacher network channels will disappear while using multiple linear layers transform. So we do not add extra processing for teacher features. For the student transform, multiple linear layers bring enough hidden expression space to represent student features and are more suitable for our framework.

3) Data Augmentation: Both of our proposed methods greatly profit from the randomness of data augmentation. It's

TABLE VII
SUPPLEMENTARY EXPERIMENTS FOR TABLE I. OUR REPORTED RESULTS ARE AVERAGED OVER FIVE RUNS

Teacher **Student**	**WRN-40-2** **WRN-40-1**	**resnet110** **resnet20**	**resnet110** **resnet32**	**ResNet50** **MobileNetV2**	**res32x4** **ShuffleNetV1**	**WRN-40-2** **ShuffleNetV1**
Teacher	75.61	74.31	74.31	79.34	79.42	79.34
Student	71.98	69.06	71.14	64.60	70.50	70.36
KD	73.59	70.92	73.08	67.35	74.07	74.83
FitNet	73.71	70.95	73.21	68.54	74.82	75.55
AT	73.92	71.03	73.29	69.28	74.76	75.61
SP	73.85	71.15	73.12	68.99	73.80	75.56
CCKD	73.69	70.96	73.06	68.95	73.63	75.63
VID	73.95	70.93	73.19	68.88	74.28	75.36
RKD	73.76	70.98	73.25	68.46	74.20	75.45
PKT	73.89	71.08	73.32	68.44	74.06	75.51
AB	73.76	70.95	73.16	69.32	76.24	76.58
FT	74.02	71.03	73.21	69.01	74.31	75.18
NST	73.62	71.14	73.21	68.92	74.51	75.02
CRD	74.38	71.56	73.75	69.54	75.12	76.27
WCoRD	74.72	71.88	74.20	70.12	75.77	76.68
SSKD	**76.13**	71.48	73.64	**72.57**	78.44	77.40
DCCD	75.25	71.90	74.21	71.20	76.64	76.81
DCCD*	75.94	**72.17**	**75.03**	72.36	**79.01**	**78.49**

TABLE VIII
ABLATION EXPERIMENTS FOR DKD ON CIFAR-100. OUR REPORTED RESULTS ARE AVERAGED OVER FIVE RUNS

Teacher **Student**	**WRN-40-2** **WRN-16-2**	**resnet56** **resnet20**	**resnet32x4** **resnet8x4**	**vgg13** **vgg8**	**vgg13** **MobileNetV2**	**ResNet50** **vgg8**	**resnet32x4** **ShuffleNetV2**
Teacher	75.61	72.34	79.42	74.64	74.64	79.34	79.42
Student	73.26	69.06	72.50	70.36	64.60	70.36	71.82
KD	74.92	70.66	73.33	72.98	67.37	73.81	74.45
CRD	75.64	71.63	75.46	74.29	69.94	74.58	76.05
KD+DKD	76.01 (↑1.09)	71.79 (↑1.13)	74.59 (↑1.26)	74.20 (↑1.22)	68.40 (↑1.03)	74.14 (↑0.33)	75.88 (↑1.34)
DCCD (KD+DKD+CCD)	76.56 (↑1.64)	72.35 (↑1.69)	76.57 (↑3.24)	74.90 (↑1.92)	70.01 (↑2.64)	75.71 (↑1.90)	77.41 (↑2.87)

worth studying the effectiveness of our method under different data augmentation settings. In Table VI, we experimented on four different data augmentation settings: 1) ***norm***: the most common augmentation on CIFAR-100, including *crop*, *flip* and *normalize*; 2) ***mixup*** [49]: convex combinations of pairs of examples and their labels; 3) ***randaugment*** [50]: practical automated data augmentation with a reduced search space; 4) ***randerasing*** [51]: randomly select a rectangle region in an image and erase its pixels with random values. More sophisticated data augmentation brings improvements in normal training and original KD. However, our method shows a bigger improvement when used in more complex data processing, this proves that our method can effectively utilize the randomness of data augmentation and work with different types of data augmentation settings. Finally, our methods achieved sufficiently excellent results that the student network is abreast of the teacher network on *Accuracy* with different data augmentation.

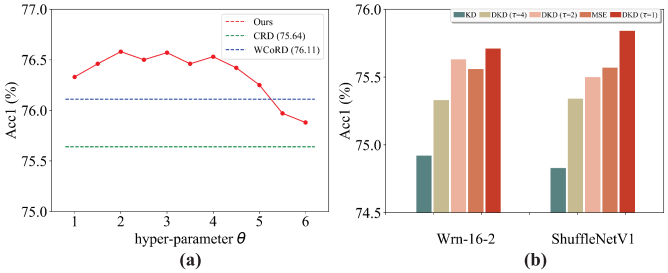


Fig. 9. Sensitivity analysis experiments for our method and both use WRN-40-2 as the teacher. (a) Performance for student WRN-16-2 with different θ in $\mathcal{L}_{\text{CCD}}$. (b) Performance for students WRN-16-2 and ShuffleNetV1 with different settings in $\mathcal{L}_{\text{DKD}}$.

4) Sensitivity Analysis: In contrastive learning, the balance of positive and negative samples learning weight is an important factor for model performance. We set the hyper-parameter θ as the learning ratio in our framework. The sensitivity of DCCD to the hyper-parameter θ is tested with teacher WRN-40-2 and student WRN-16-2. As shown in Fig. 9(a), our method surpasses CRD and WCoRD in general and usually performs better when $\theta \in [1.5, 4.5]$. Therefore, θ is set as 2.0 for all the experiments in this article.

Besides, as a supplement to the original KD, DKD also uses $\mathcal{L}_{\mathrm{KL}}$ to narrow the distance between teacher output and student output. We naturally consider whether *MSE* loss or higher temperature τ could strengthen DKD. The results on WRN-40-2 $\rightarrow$ WRN-16-2 and WRN-40-2 $\rightarrow$ ShuffleNetV1 are summarized in Fig. 9(b). We can see that $\mathcal{L}_{\mathrm{KL}}$ works better than $\mathcal{L}_{\mathrm{MSE}}$ and DKD does not need for higher temperature. DKD + KD achieves a nearly 1% accuracy improvement compared with the original KD.

V. Conclusion and Future Work

In this article, we propose a novel distillation method named DCCD, which focuses on reducing the information redundancy of the student network during distillation training. We propose channel contrastive loss $\mathcal{L}_{\mathrm{CCD}}$ to establish imitation and contrast relationships between the teacher's and student's channels and difference distillation loss $\mathcal{L}_{\mathrm{DKD}}$ to supplement the traditional KD with dynamic difference knowledge. The framework reduces the overlap between the internal channels and increases the sensitivity of the external response to detailed changes. Our method meets or exceeds other state-of-the-art distillation methods on various datasets and tasks. Experiments and visualizations demonstrate that our proposed method has unique effectiveness in reducing student network redundancy and improving model performance compared with other distillation methods. For future work, we are interested in applying DCCD to other tasks to distill from extremely deep teachers into compact students. Moreover, the channel-based CCD DCCD we proposed differs from the previous instance-level CCDs in terms of implementation principle and effect. Studying the essential difference between these two types of methods and whether they can be merged into a unified contrastive distillation framework is an interesting topic for us to explore.

Appendix

A. Supplementary Experiments

In Table I, we present our experimental data for the CIFAR-100 dataset, including two scenarios: 1) the student and the teacher share the same network architecture and 2) different network architectures are used. In order to better verify the stability and validity of DCCD, we add more experimental data in Table VII. It can be seen that our method still achieves better effects than other CCDs CRD, SSKD, and WCoRD in more teacher-student network pairs.

Due to the space limitation of the article, we did not conduct a detailed analysis on the role of DKD, except in Tables II and III. As an improvement to traditional distillation,

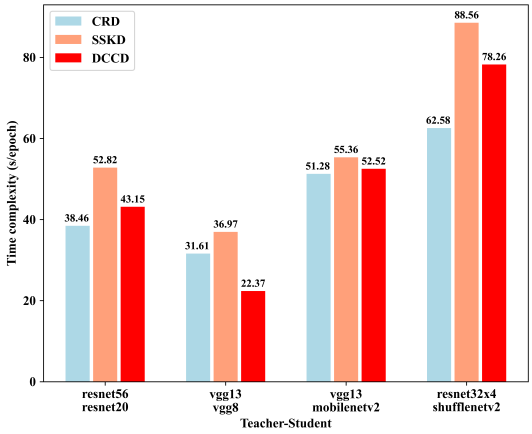


Fig. 10. Training time per epoch for CRD, SSKD, and our DCCD for different teacher-student network pairs on CIFAR-100.

DKD introduces more dynamic knowledge and makes the student network more sensitive to subtle response changes in all categories. The premise of DKD is that the responses of the teacher network and the student network are similar, so DKD must work on the basis of KD. Therefore, the specific performance of DKD is difficult to separate from KD. We conducted more supplementary experiments on the effect of KD + DKD in Table VIII. After introducing the DKD loss, compared with traditional distillation, KD + DKD achieves a relatively high accuracy improvement in the CIFAR-100 experimental results. Under the same experimental setting, KD + DKD can achieve a similar accuracy level as the CRD method.

B. Time Complexity Analysis

Since the teacher network and additional knowledge transfer modules can be removed after KD training, the student network can achieve better performance in inference without computational burden. However, we employ data augmentations on the same image to obtain two augmented inputs and rely on the randomness of data augmentations to transfer more teacher network's dark knowledge. So it is worth analyzing the time complexity during training. We measure the time complexity of CRD, SSKD, and DCCD on a single GeForce RTX 2080Ti GPU (For all we know right now, we do not find an open-source project for WCoRD). The results are summarized in Fig. 10. Compared with other contrastive-based methods, our method adds little extra training time per epoch on four experiments for different teacher-student settings. CRD needs to maintain a negative sample memory bank and requires an additional random sample operation for each batch. SSKD trains its Self-Supervision Prediction module for extra epochs and stacks four rotation angle inputs for 4$\times$ training batch size. Our method DCCD also stacks augmented inputs to reduce the network forward times.

As shown in Fig. 11, CCDs (CRD, SSKD, and DCCD) generally consume longer training time per epoch than traditional distillation methods (KD, AT, and CCKD). This is mainly due to additional module components that need to be trained, more data augmentation steps and more complex loss

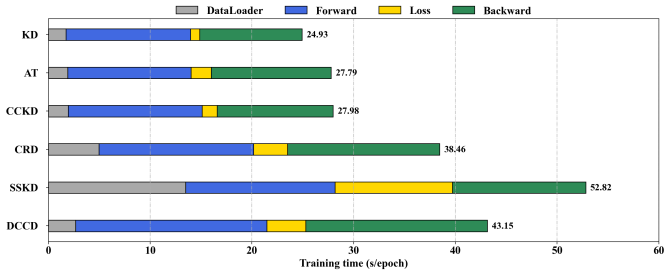


Fig. 11. Analysis of the composition of training time complexity. Results are obtained on GeForce RTX 2080Ti with teacher network ResNet-56, student network ResNet-20 and dataset CIFAR-100.

calculation processes. Although our method introduces once additional data augmentations and extra knowledge transfer module $\mathbf{M}_s$, the overall training time complexity is similar to CRD and slightly better than SSKD.

REFERENCES

[1] E. L. Denton, W. Zaremba, J. Bruna, Y. Le Cun, and R. Fergus, "Exploiting linear structure within convolutional networks for efficient evaluation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1269–1277.

[2] X. Yu, T. Liu, X. Wang, and D. Tao, "On compressing deep models by low rank and sparse decomposition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7370–7379.

[3] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," 2015, *arXiv:1506.02626*.

[4] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. Peter Graf, "Pruning filters for efficient ConvNets," 2016, *arXiv:1608.08710*.

[5] W. Chen, J. Wilson, S. Tyree, K. Weinberger, and Y. Chen, "Compressing neural networks with the hashing trick," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2285–2294.

[6] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-NET: ImageNet classification using binary convolutional neural networks," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 525–542.

[7] J. Chen, L. Liu, Y. Liu, and X. Zeng, "A learning framework for n-Bit quantized neural networks toward FPGAs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 3, pp. 1067–1081, Mar. 2020.

[8] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[9] M. Gao, Y. Shen, Q. Li, and C. C. Loy, "Residual knowledge distillation," 2020, *arXiv:2002.09168*.

[10] K. Xu, L. Rui, Y. Li, and L. Gu, "Feature normalized knowledge distillation for image classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 1, 2020, pp. 664–680.

[11] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for thin deep nets," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015, pp. 1–13.

[12] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," 2016, *arXiv:1612.03928*.

[13] J. Yim, D. Joo, J. Bae, and J. Kim, "A gift from knowledge distillation: Fast optimization, network minimization and transfer learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4133–4141.

[14] B. Peng et al., "Correlation congruence for knowledge distillation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 5007–5016.

[15] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–19.

[16] G. Xu, Z. Liu, X. Li, and C. C. Loy, "Knowledge distillation meets self-supervision," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*. Glasgow, U.K.: Springer, Aug. 2020, pp. 588–604.

[17] L. Chen, D. Wang, Z. Gan, J. Liu, R. Henao, and L. Carin, "Wasserstein contrastive representation distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16296–16305.

[18] Y. Aflalo, A. Noy, M. Lin, I. Friedman, and L. Zelnik, "Knapsack pruning with inner distillation," 2020, *arXiv:2002.08258*.

[19] Z. Chen, J. Niu, L. Xie, X. Liu, L. Wei, and Q. Tian, "Network adjustment: Channel search guided by flops utilization ratio," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10658–10667.

[20] N. Aghli and E. Ribeiro, "Combining weight pruning and knowledge distillation for CNN compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 3191–3198.

[21] T. Li, J. Li, Z. Liu, and C. Zhang, "Few sample knowledge distillation for efficient network compression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 14639–14647.

[22] L. Yao, R. Pi, H. Xu, W. Zhang, Z. Li, and T. Zhang, "Joint-DetNAS: Upgrade your detector with NAS, pruning and dynamic distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10175–10184.

[23] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 5191–5198.

[24] A. Malinin, B. Mlodozeniec, and M. Gales, "Ensemble distribution distillation," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–24.

[25] J. Kim, S. Park, and N. Kwak, "Paraphrasing complex network: Network compression via factor transfer," in *Proc. NIPS*, 2018, pp. 1–10.

[26] B. Heo, M. Lee, S. Yun, and J. Y. Choi, "Knowledge transfer via distillation of activation boundaries formed by hidden neurons," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 33, no. 1, 2019, pp. 3779–3787.

[27] B. Heo, J. Kim, S. Yun, H. Park, N. Kwak, and J. Y. Choi, "A comprehensive overhaul of feature distillation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 1921–1930.

[28] S. Lee and B. C. Song, "Graph-based knowledge distillation by multi-head attention network," 2019, *arXiv:1907.02226*.

[29] H. Chen, Y. Wang, C. Xu, C. Xu, and D. Tao, "Learning student networks via feature embedding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 25–35, Jan. 2020.

[30] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3967–3976.

[31] Y. Li, J. Yang, Y. Song, L. Cao, J. Luo, and L.-J. Li, "Learning from noisy labels with distillation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1910–1918.

[32] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[33] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 766–774.

[34] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[35] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," 2019, *arXiv:1906.00910*.

[36] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9729–9738.

[37] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[38] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," 2021, *arXiv:2103.03230*.

[39] A. Krizhevsky, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[40] F. Tung and G. Mori, "Similarity-preserving knowledge distillation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1365–1374.

[41] S. Ahn, S. X. Hu, A. Damianou, N. D. Lawrence, and Z. Dai, "Variational information distillation for knowledge transfer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9163–9171.

[42] N. Passalis and A. Tefas, "Learning deep representations with probabilistic knowledge transfer," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 268–284.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[44] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[45] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[46] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShufflesNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[48] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, G. Gordon, D. Dunson, and M. Dudík, Eds. Fort Lauderdale, FL, USA, vol. 15, Apr. 2011, pp. 215–223.

[49] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*.

[50] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2020, pp. 702–703.

[51] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 13001–13008.

Yuang Liu received the B.Eng. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2020, where he is currently pursuing the M.S. degree with the Department of Control Science and Engineering, Institute of Cyber Systems and Control.

His research interests include neural network compression and deep learning.

Jun Chen received the B.S. degree from the Department of Mechanical and Electrical Engineering, China Jiliang University, Hangzhou, China, in 2016, and the M.S. degree from the Zhejiang University, Hangzhou, in 2020, where he is currently pursuing the Ph.D. degree with the Department of Control Science and Engineering, Institute of Cyber Systems and Control.

His research interests include neural network quantization and deep learning.

Yong Liu (Member, IEEE) received the B.S. degree in computer science and engineering and the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 2001 and 2007, respectively.

He is currently a Professor with the Department of Control Science and Engineering, Institute of Cyber Systems and Control, Zhejiang University. He has published more than 30 research papers in machine learning, computer vision, information fusion, and robotics. His latest research interests include machine learning, robotics vision, and information processing and granular computing.