Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Dual teachers for self-knowledge distillation

Zheng Li [a], Xiang Li [a], Lingfeng Yang [b], Renjie Song [c], Jian Yang [a,*], Zhigeng Pan [d]

[a] PCA Lab, VCIP, College of Computer Science, Nankai University, Tianjin, 300350, China
[b] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, 210094, China
[c] Megvii Technology Limited Corporation, Beijing, 100190, China
[d] School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, 210044, China

ARTICLE INFO

Keywords:
Model compression
Image classification
Self-knowledge distillation
Dual teachers

ABSTRACT

We introduce an efficient self-knowledge distillation framework, Dual Teachers for Self-Knowledge Distillation (DTSKD), where the student receives self-supervisions by dual teachers from two substantially different fields, i.e., the past learning history and the current network structure. Specifically, DTSKD trains a considerably lightweight multi-branch network and acquires predictions from each, which are simultaneously supervised by a historical teacher from the previous epoch and a structural teacher under the current iteration. To our best knowledge, it is the first attempt to jointly conduct historical and structural self-knowledge distillation in a unified framework where they demonstrate complementary and mutual benefits. The Mixed Fusion Module (MFM) is further developed to bridge the semantic gap between deep stages and shallow branches by iteratively fusing multi-stage features based on the top-down topology. Extensive experiments prove the effectiveness of our proposed method, showing superior performance over related state-of-the-art self-distillation works on three datasets: CIFAR-100, ImageNet-2012, and PASCAL VOC.

1. Introduction

Knowledge distillation was originally proposed by Hinton et al. [1] to exploit the knowledge of a heavily pre-trained teacher, aiming to improve the generalization ability of a compact student. Traditional methods [2–5] follow a typical two-stage paradigm that starts with a cumbersome teacher and then distills the knowledge into a compact student. Self-Knowledge Distillation (Self-KD) [6–8] methods have significantly improved the efficiency of the distillation process by eliminating the need to construct an additional and individual heavy teacher model as a knowledge provider. Unlike the two-stage traditional distillation method, Self-KD achieves distillation through a single-stage training process. Existing distillation methods primarily employ a self-teacher, which is constructed using various sources of information such as sample relationships [8,9], features [7], and historical predictions [6], to regularize the training of the current model. DDGSD [9] is a method that transfers knowledge between different augmented versions of the same training data without the requirement of an additional teacher. By leveraging the diverse augmented samples, DDGSD achieves efficient distillation without the need for an external teacher model. Another self-distillation method, DKS [10], constructs multiple structural branches within the backbone network. These branches enable bidirectional teaching within the structural hierarchy itself, facilitating the transfer of knowledge between different levels of abstraction. PS-KD [6] is another approach that utilizes the historical predictions from previous learning epochs to guide the learning of the current network. By leveraging the model's past performance, PS-KD offers a self-distillation mechanism that enhances the student model's knowledge retention and generalization ability. After the distillation process, the well-trained student model is ready for final deployment, where it can be utilized in various real-world applications.

Although current self-knowledge distillation methods have shown promising results, they primarily center their efforts on creating a singular, unitary teacher model. This emphasis on a single-teacher model has inadvertently restricted the diversity of guidance and perspectives available for the student model to learn from. In light of this limitation, it is worth considering the way human beings accumulate knowledge — by drawing insights from a multitude of experts across various domains. This prompts us to pose a fundamental question: could a student model derive greater advantages and broader insights by incorporating the teachings of two or more significantly distinct teacher models when engaging in the process of self-knowledge distillation?

To answer the above question, in this work, we explore a novel self-knowledge distillation framework, **D**ual **T**eachers for **S**elf-**K**nowledge **D**istillation (DTSKD), where the student network receives self-supervis-

[*] Corresponding author.
E-mail addresses: zhengli97@mail.nankai.edu.cn (Z. Li), xiang.li.implus@nankai.edu.cn (X. Li), yanglfnjust@njust.edu.cn (L. Yang), songrenjie@megvii.com (R. Song), csjyang@nankai.edu.cn (J. Yang), 443922077@qq.com (Z. Pan).

https://doi.org/10.1016/j.patcog.2024.110422
Received 27 May 2022; Received in revised form 2 November 2023; Accepted 9 March 2024
Available online 12 March 2024

ions by dual teachers from two dramatically distinct fields, i.e., the past learning history and the current network structure. Specifically, we present an efficient multi-branch variant of a given target backbone network by attaching very *lightweight* auxiliary branches to different backbone stages. We treat the past corresponding prediction as a historical learning target and the current hierarchical probability as a structural distillation target. Each branch is jointly optimized by two targets in our simple and unified self-distillation framework.

To alleviate the semantic gaps [11] of knowledge transfer at different depth layers in a multi-branch network, **M**ixed **F**usion **M**odule (MFM) is further introduced to fill the gap by iteratively fusing multi-stage features based on the top-down topology. In the test, the target network is acquired by simply removing the redundant branches, nullifying any extra costs for final deployment.

Our contributions can be summarized as follows:

- To our best knowledge, we are the first work to jointly conduct historical and structural self-knowledge distillation in a simple and unified framework, where they demonstrate complementary advantages and mutual benefits.
- We demonstrate that an extremely lightweight design of auxiliary branches outperforms the previous heavy counterparts in both efficiency and accuracy.
- We show that the semantic gap exists between different stages within the network, and the Mixed Fusion Module (MFM) is further proposed to fill the gap between deep stages and shallow auxiliary branches by iteratively fusing multi-stage features from the top-down topology.
- Extensive experiments prove the effectiveness of our proposed framework on three datasets: CIFAR-100, ImageNet-2012, and PASCAL VOC.

2. Related work

2.1. Traditional knowledge distillation

Deep convolutional networks have achieved remarkable success in various computer vision applications [12,13]. With the growing number of model parameters, a large amount of computational resources is required to achieve state-of-the-art accuracy. However, networks with millions of parameters are hard to deploy to platforms with limited computing resources. To address this issue, a variety of network compression approaches such as quantization [14,15], binarization [16,17] and knowledge distillation [1,4], have been exploited to obtain a small network that can work as well as the large network while effectively reducing the computational costs and memory consumption.

Knowledge distillation [1] aims at learning a comparable and lightweight student by utilizing the knowledge of a pre-trained cumbersome teacher. The traditional distillation paradigm usually consists of two stages. In the first stage, we first pre-train a large teacher model. In the second stage, the existing teacher model is used to train the student model by aligning their output representations (e.g., feature [4,17], logit [5,18]). Since the student model has inherited the knowledge of the teacher, it can replace the over-parameterized teacher model for fast inference. FitNet [19] proposes to let the student mimic the intermediate representations of the teacher. AT [20] tries to transfer the attention map of a teacher to a student. FSP [2] proposes to generate the FSP matrix from the intermediate layer feature and use this matrix to guide the learning of the student. UNIX [18] proposes to adaptively mix samples based on uncertainty. Different levels of mixup are applied depending on the degree of sample uncertainty. Recent works [8,21,22] also adopt contrastive representation learning to obtain better performance. The main idea is to pull two positive pairs closer while negative pairs far away, by utilizing different data augmentation techniques. CRD [21] maximizes the mutual information between a teacher and a student via contrastive learning. SSKD [23] exploits the self-supervised features of the teacher to transfer richer knowledge to the student. In addition to the classic image classification task [24], knowledge distillation has also been used in many vision tasks, such as object detection [25,26], semantic segmentation [27,28], and human pose estimation [29].

Existing two-stage distillation methods [30,31] demonstrate that learning from multiple teachers can significantly improve the student. Li et al. [31] introduce inter-domain and intra-domain teachers to help one student perform cardiac segmentation. Dong et al. [32] let the student learn to detect all the foreground objects in the base and novel classes under the supervision of dual teachers. In incremental learning, DT-ID [30] transfers the knowledge from two teachers, which are trained in the old and new classes, to one student. The above works use multiple teachers trained on *different data sources* to perform two-stage distillation. Different from that, in this work, we explore a one-stage multi-teacher self-distillation framework trained on a single data source, but from distinct aspects (i.e., history and structure).

TAKD [11] reveals that a larger teacher cannot always lead to a better student due to the semantic gap between different capacity models. DGKD [33] proposes multiple teacher assistants with gradually decreased size to bridge the gap. SemCKD [34] proposes to perform the semantic calibration of intermediate representations when conducting feature distillation.

2.2. Self-knowledge distillation

The self-knowledge distillation methods [6,7,35] significantly improves the training efficiency by eliminating the need for a cumbersome teacher network. The student can be trained even without a static teacher. Self-KD can be roughly divided into three types: sample-based [8,9,35], structure-based [7,10] and history-based [36, 37] method. Sample-based Self-KD focuses on exploiting information between samples. DDGSD [9] proposes to learn consistent feature or posterior distributions between different augmented versions of the same training samples. CS-KD [35] proposes a novel class-wise regularization term that penalizes the differences in predictions between samples of the same class. Yu et al. [8] introduce the self-distillation method into the Partial Label Learning (PLL) problem by weightily aggregating cross-sample knowledge. Structure-based Self-KD proposes to utilize the high-level information encoded in different stages of the backbone feature. BYOT [7] proposes to attach auxiliary branches to different backbone stages. The deep structural information encoded in deep stages can be further distilled into shallow branches. DKS [10] explores the possibility of utilizing structural knowledge learned by shallow layers to regularize the learning of the backbone network in reverse. FRSKD [38] constructs a heavy self-teacher network and distills refined knowledge to the original backbone network. History-based self-KD proposes to use historical information (e.g., soft labels, models) to regularize the training of the current network. BAN [36] explores a multi-step way to let the model guide its own training process. It first pre-trains a network as a teacher. Then, at each consecutive step, a new identical model is trained under the supervision of the previous generation. In PS-KD [6], the student is trained with a soft target computed as the linear combination of one-hot ground-truth labels and historical predictions from the previous epoch.

While prior research endeavors have predominantly delved into the paradigm of employing a singular, unitary teacher model for self-distillation, our work diverges from this convention by introducing a novel approach. In this study, we propose the concurrent utilization of two distinct self-teachers, a departure from the established practice, to realize enhanced self-distillation outcomes.

Notably, to the best of our knowledge, this paper marks the inaugural foray into the domain of self-distillation that jointly integrates both historical and structural self-teaching mechanisms within a unified and lightweight framework. This pioneering endeavor represents a significant departure from the prevailing literature and opens up new avenues for advancing the field of self-knowledge distillation.

3. Proposed method

In this section, we briefly review the basic concept of knowledge distillation, and then we describe the details of our dual-teacher self-knowledge distillation framework.

3.1. Traditional knowledge distillation

Traditional two-stage knowledge distillation procedure always starts with a pre-trained cumbersome teacher. Then a lightweight student will be trained under the supervision of a heavy teacher in the form of soft predictions [1] or intermediate representations. After the distillation, the student can master the expertise of the teacher, which is thus used for efficient deployment. Given the labeled classification dataset $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{I}$, the Kullback–Leibler (KL) divergence loss is used to minimize the discrepancy between the softened output probabilities of the student network and the teacher network:

$$L_{KL}(p_t, p_s) = \sum_{i=1}^{I} \tau^2 KL(\sigma(p_t^i/\tau), \sigma(p_s^i/\tau)), \tag{1}$$

where i denotes the sample index, τ os the temperature parameter and $\sigma(\cdot)$ is the softmax function. p_t^i and p_s^i denote the softened probability produced by the teacher and the student, respectively. A larger temperature τ will make the probability distribution softer.

To train a multi-class classification network, traditional methods also minimize the Cross-Entropy (CE) loss between the predicted probabilities p_s^i and the ground-truth one-hot label y_i of each training sample:

$$L_{CE}(p_s) = \sum_{i=1}^{I} CE(p_s^i, y_i), \tag{2}$$

With both hard labels and soft labels, the final loss function of the conventional knowledge distillation is written with the balancing parameter λ as follows:

$$L_{classic} = L_{CE} + \lambda L_{KL}. \tag{3}$$

3.2. Dual teachers for self-knowledge distillation

As opposed to previous self-distillation works that only utilize a unitary teacher, in this work, we explore an efficient self-distillation approach that utilizes both historical and structural teachers in a unified framework, where they demonstrate complementary advantages and mutual benefits.

An overview of our proposed self-distillation framework is illustrated in Fig. 1. It mainly consists of three components: (1) The target backbone network T with B stages for deployment, which meets the requirement of the target complexity. (2) $B-1$ Auxiliary hierarchical branches in different stages, each of which receives the historical signal from the previous learning epoch (i.e., t-1 epoch) and high-level structural guidance from the current network. Each branch consists of an average pooling layer, a Mixed Fusion Module, and a fully connected layer. It acts as an independent classifier to generate the soft probability for distillation. These auxiliary branches are only utilized in training and can be removed for final deployment. (3) $B-1$ Mixed Fusion Modules, which consist of both addition and concatenation operators, as shown in Fig. 3. It learns to bridge the semantic gap between deep stages and shallow auxiliary branches by iteratively fusing multi-stage backbone features based on the top-down topology. Fig. 2 illustrates the architectural differences between four representative self-distillation approaches, where our efficient structure design demands less computational cost than the previous networks.

After the feed forward computation at tth epoch, we can obtain the backbone feature set $T = \{T_1, T_2, \ldots, T_B\}$ and the output probability distribution $p_{0,t}(\mathbf{x})$ given a natural input image $\mathbf{x}$. For $b \in [1, B-1]$, the backbone features T_b is first down-sampled to a size of 1×1 through the global average pooling layer which largely reduces the computation complexity. Then, it is fused with semantically stronger features F_{b+1} via the Mixed Fusion Module (MFM). The details of the MFM are elaborated in the following section. In this way, using the hierarchical feature set $F = \{F_1, F_2, \ldots, F_B\}$ as the initial input of the auxiliary branches, we can obtain the corresponding prediction set $p = \{p_{1,t}(\mathbf{x}), p_{2,t}(\mathbf{x}), \ldots, p_{(B-1),t}(\mathbf{x})\}$ at tth epoch.

Structural Distillation. The structural knowledge encoded in the last stage can effectively guide the learning of shallow stages. In our DTSKD, we align the predictions between the main network $p_{0,t}$ and three auxiliary branches $p_{b,t}$ by minimizing the KL divergence loss. The optimization process for structural distillation at tth epoch can be formulated as follows:

$$L_{stru} = \sum_{b=1}^{B-1} L_{KL}\big(p_{0,t}(\mathbf{x}) \parallel p_{b,t}(\mathbf{x})\big), \tag{4}$$

where $p_{0,t}(\mathbf{x})$ and $p_{b,t}(\mathbf{x})$ are the prediction produced by the backbone network #0 and auxiliary branch #b at tth epoch, respectively. The temperature T is set to 4 as suggested in [38].

Historical Distillation. Historical prediction from the past epoch can be regarded as the twin result of the same instance under different augmentation and network parameters. Inspired by contrastive learning [39], it can be modeled as a simple and effective teacher. Unlike traditional distillation work that requires a static teacher to give a priori, we use past prediction $p_{b,(t-1)}(\mathbf{x})$ to build a dynamic one to guide the learning of the current network. In particular, in tth epoch of training, the optimization target $p'_{b,t}(\mathbf{x})$ for input $\mathbf{x}$ is the weighted linear combination of one-hot ground-truth label $\mathbf{y}$ and its past prediction $p_{b,(t-1)}(\mathbf{x})$ with hyper-parameter β. The optimization process for historical distillation at tth epoch can be formulated as:

$$L_{his} = \sum_{b=0}^{B-1} \lambda_b \cdot L_{KL}\big(p'_{b,t}(\mathbf{x}) \parallel p_{b,t}(\mathbf{x})\big), \tag{5}$$

$$p'_{b,t}(\mathbf{x}) = \beta_t \cdot \mathbf{y} + (1-\beta_t) \cdot p_{b,(t-1)}(\mathbf{x}), \beta_t \in [0,1], \tag{6}$$

where $p_{b,(t-1)}(\mathbf{x})$ denotes the prediction generated by Branch #b at $(t-1)$-th epoch, λ_b is the hyper-parameter to balance the loss terms between the backbone and three auxiliary branches. The temperature T is set to 1 in Eq. (5). For a four-stage DTSKD framework, we have three auxiliary branches and set $\lambda_0 = 3$, $\lambda_1 = \lambda_2 = \lambda_3 = 1$. Note that the ground-truth label $\mathbf{y}$ is contained in Eq. (6), so that the network can be well optimized towards the ground-truth label. The backbone network is denoted as Branch #0.

The hyper-parameter β determines the proportion of learning from ground truth and history information. At the beginning of training, the network needs to learn meaningful representations from the dataset, so it is reasonable to maintain the main supervision from ground truth labels and then gradually increase the proportion of historical supervision (i.e., gradually decrease the β value). In this work, we employ a cosine decay strategy, where the hyper-parameter β at tth epoch is computed as follows:

$$\beta_t = \beta_{min} + (\beta_{max} - \beta_{min}) \cdot \frac{1}{2} \cdot (1 + cos(\frac{t}{t_{total}} \cdot \pi)), \tag{7}$$

where β_{max} and β_{min} are ranges for the hyper-parameter β, and t_{total} denotes the total epoch for training.

3.3. Mixed fusion module

Semantic gaps [11] exist not only between two networks with different capacities but also between different stages within the whole network. To validate this issue, we conduct experiments by directly aligning the predictions between the shallow and deep stages without any top-down connections. As shown in Table 7, it significantly hurts the network performance and make the backbone perform even worse than the baseline results due to the semantic gap between

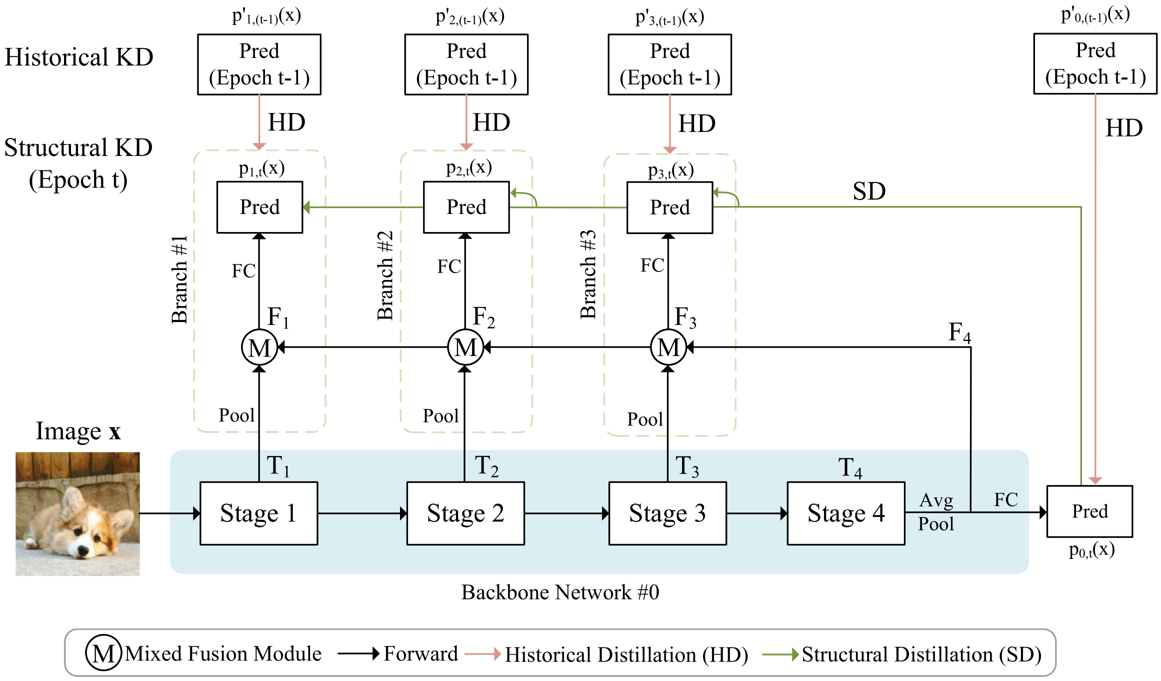


Fig. 1. An overview of our proposed self-distillation framework, Dual Teachers for Self-Knowledge Distillation (DTSKD). We divide the backbone network #0 into four stages and construct three hierarchical sub-networks (denoted as Branch #1, #2, and #3) in an FPN-like way by sharing various stages of the backbone features. Each branch is supervised by a history teacher from the previous learning epoch and a structural teacher from the current iteration. After training, only the blue part (i.e., Backbone Network #0) is utilized for efficient deployment. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

different stages. We demonstrate that the semantic gap is a critical issue in the self-distillation method, which drives us to design efficient and hierarchical communications between deep and shallow representations.

Further inspired by MLN [40], we propose the Mixed Fusion Module (MFM) to fill the gap between different stages by iteratively fusing multi-stage features based on the top-down topology. Previous works [10,41] repeat the building blocks of the backbone network to construct heavy branches that bridge the semantic gap when performing cross-layer distillation. Instead of such a complex design, the Mixed Fusion Module, which consists of both addition and concatenation operations, effectively enhances the representation ability of the branch network by iteratively fusing multi-stage features at the top-down paths. The architecture of our proposed MFM is depicted in Fig. 3.

Specifically, with a spatially coarser feature T_b, we first downsample the spatial resolution to a size of 1 × 1 through the average pooling layer. The following 1 × 1 convolution operation is used to align the channel dimension between two input features. Then the pooled features are fused with the semantically stronger features F_{b+1} through the addition and concatenation operations, which can be calculated as:

$$F_b = Conv\Big(\big(f_t(T_b) + f_s(F_{b+1})\big) \parallel f_s(F_{b+1})\Big), b \in [1, B-1] \tag{8}$$

where the symbols "+" and "‖" denote addition and concatenation operations, respectively. $f_t(\cdot)$ denotes the function of average pooling layer and the 1 × 1 *conv* block. $f_s(\cdot)$ denotes the function of the 1 × 1 *conv* block. *Conv* is a 1 × 1 convolution operation that merges the features after the concatenation operation and halves the number of feature channels to C_{b+1}, as shown in Fig. 3. This feature fusion process is iterated until the latest feature F_1 is generated.

Through our top-down feature fusion process, even the shallowest auxiliary branch can still obtain sufficient semantic information, bridging the gap between deep stages and shallow branches during structural distillation. Multiple strong branch networks can make the backbone learn to provide robust intermediate representations during training, resulting in better representation ability.

3.4. Optimization

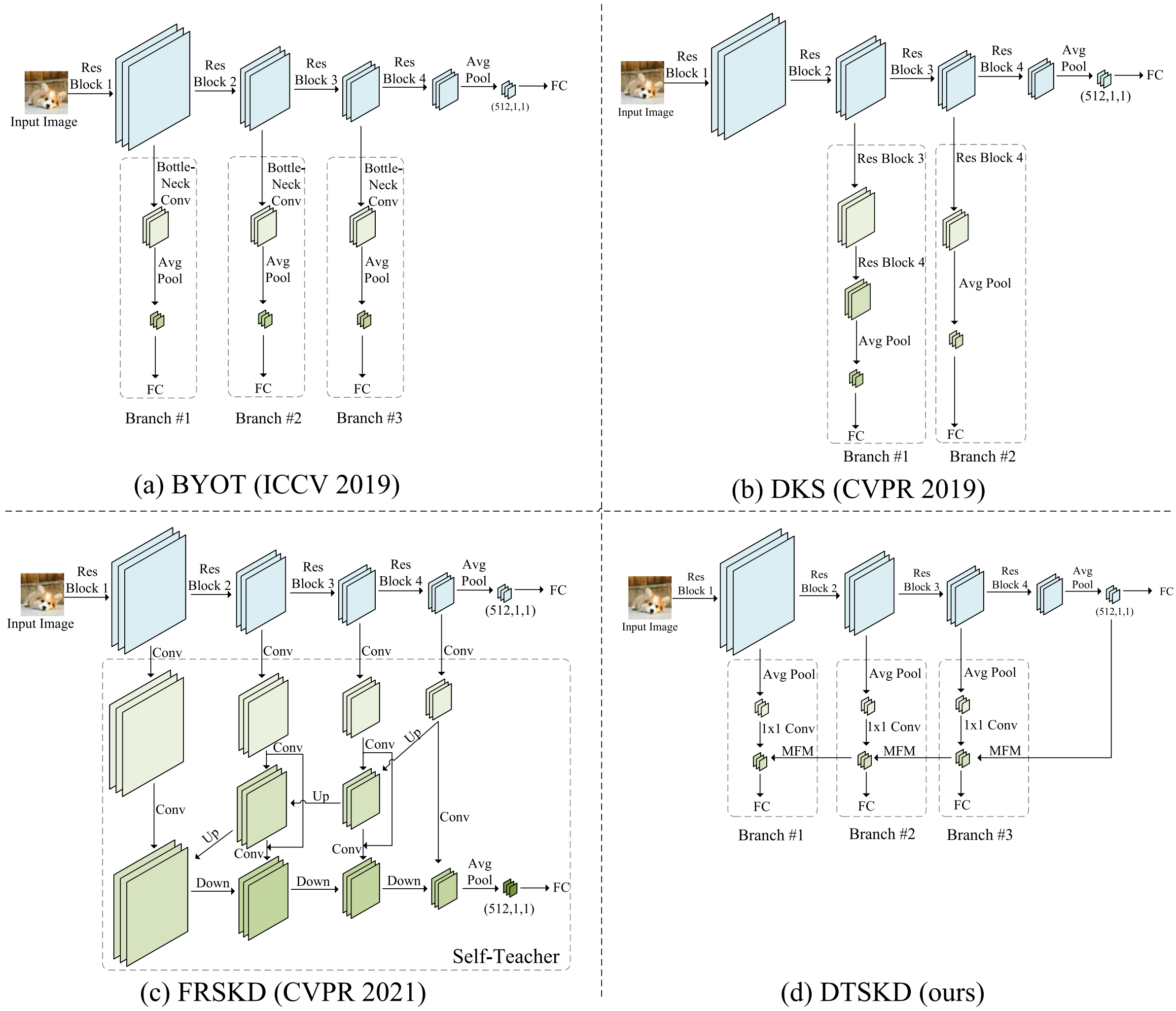
To get a better understanding of our method, we describe the whole training procedure in Algorithm 1. In our method, the student model is jointly supervised by self-teachers from two dramatically distinct fields, i.e., the past learning history and the current network structure. The whole objective function can be formulated as follows:

$$L_{total} = \alpha_1 L_{his} + \alpha_2 L_{stru}. \tag{9}$$

where α_1 and α_2 are the hyper-parameters to control the impact of each loss term and also satisfy $\alpha_1 + \alpha_2 = 1$. After distillation, the target model can be obtained by simply removing three redundant auxiliary branches. No additional computational cost will be introduced during model deployment.

Algorithm 1 Dual Teachers for Self-Knowledge Distillation

```
Input: Labeled training dataset $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n}$; Training epoch number $t_{total}$; A target backbone network $\theta_0$ with three auxiliary branches $\{\theta_1, \theta_2, \theta_3\}$;
Initialize: Epoch t=1; Randomly initialize $\{\theta_i\}_{i=0}^{3}$;
while $t \leq t_{total}$ do
    Feed forward propagation and obtain the intermediate representations $\{F_i\}_{i=1}^{3}$ through the MFM with Eqn. (8).
    Obtain the classification predictions $\{p_{i,t}(x)\}_{i=0}^{3}$.
    Compute historical labels $\{p'_{i,t}(x)\}_{i=0}^{3}$ with Eqn. (6).
    Compute the historical distillation loss $L_{his}$ in Eqn. (5).
    Compute the structural distillation loss $L_{stru}$ in Eqn. (4).
    Update the network parameters $\{\theta_i\}_{i=0}^{3}$ by minimizing $L_{total}$ in Eqn. (9).
    t=t+1.
end while
Output: A well-trained target network $\theta_0$;
```



(a) BYOT (ICCV 2019)

(b) DKS (CVPR 2019)

(c) FRSKD (CVPR 2021)

(d) DTSKD (ours)

Fig. 2. Architectural comparison of four representative self-distillation approaches. (a) In BYOT, additional bottlenecks and a fully connected layer are added after each block, forming the auxiliary branches. (b) DKS [10] repeats the building blocks of the backbone network to construct heavy auxiliary branches. (c) FRSKD [38] introduces a complex self-teacher network that distills the refined knowledge into the backbone. (d) In DTSKD, each branch only has an average pooling layer, a Mixed Fusion Module, and a fully connected layer. Our proposed framework is more efficient than its previous counterparts.

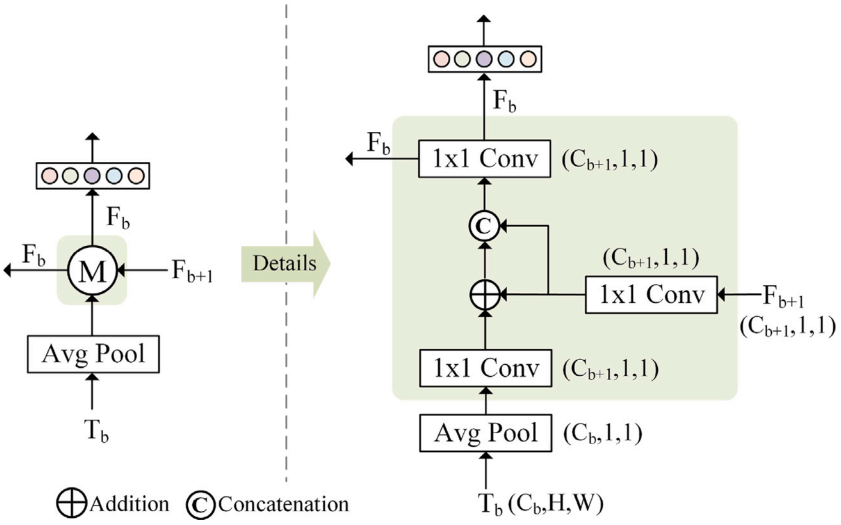


Fig. 3. An overview of our proposed Mixed Fusion Module (MFM). It utilizes convolutional modules to align the dimensions of features from different paths and dynamically fuse features through addition and multiplication operations.

Table 1
Top-1 accuracy (%) comparison of various distillation approaches on CIFAR-100. The best-performing model is indicated in boldface. Our method achieves the best results among existing history-based and structure-based self-kd methods.

Models	Baseline	BYOT	DKS	FRSKD	CS-KD	Tf-KD	PS-KD	MixSKD	DTSKD
VGG-11	69.32	72.79	72.81	70.08	72.01	71.84	72.97	73.52	**74.03**
VGG-16	73.07	75.56	74.34	73.76	74.56	75.32	75.12	76.10	**76.72**
ResNet-18	76.27	78.21	79.56	79.34	79.41	78.80	79.11	80.22	**80.46**
ResNeXt-18	79.22	79.47	80.16	80.12	80.18	80.67	80.87	81.06	**81.49**
SENet-34	78.67	80.03	80.22	78.93	79.16	80.27	80.07	80.17	**80.56**
ShuffleNetV2	70.77	72.18	72.78	73.86	71.18	73.89	73.06	74.03	**74.30**

4. Experiments

In this section, we evaluate our proposed method on five popular backbone networks (VGG [42], ResNet [12], ResNeXt [43], SENet [44], ShuffleNetV2 [45]) and three datasets (CIFAR-100, ImageNet-2012 and PASCAL VOC) to demonstrate the effectiveness of our proposed method. We compare our work with six closely related state-of-the-art distillation works including BYOT [7], DKS [10], FRSKD [38], CS-KD [35], Tf-KD$_{self}$ [46], PS-KD [6] and MixSKD [47]. Detailed ablation studies on the network components were also conducted to demonstrate its effectiveness. All evaluations are made in comparison to state-of-the-art approaches according to their official implementations based on standard experimental settings and we report the average best top-1 validation accuracy (%) over three runs. For a fair comparison, the experiments with all methods are performed under the same settings for the data pre-processing method, the batch size, the number of training epochs, the optimizer setting, the learning rate schedule, and the other commonly used hyper-parameters. The unique hyper-parameter settings in the previous works remain the same.

4.1. Experimental settings

Datasets. (1) CIFAR-100 dataset consists of colored natural images with 32 × 32 pixels. It contains 50K/10K training/test samples drawn from 100 classes. Each class has 600 images. There are 500 training images and 100 testing images per class. Same as previous works [38], the network structures are modified to fit the tiny images in CIFAR-100.

(2) ImageNet-2012 classification dataset is more challenging than CIFAR-100. It contains 1.2M images for training, and 50K for validation, from 1K classes. The resolution of input images after pre-processing is 224 × 224 pixels.

(3) The PASCAL VOC dataset is widely used for image recognition and object detection. The PASCAL Visual Object Classes (VOC) 2007 contains 9963 photos in all, with 24,640 labeled samples. The data is divided into 50% for training/validation and 50% for testing. The Pascal VOC 2012 dataset has 11,530 images in the train/val data set, including 27,450 ROI-tagged objects.

Implementation Details. All the methods are implemented in Pytorch. For CIFAR-100, we adopt the standard data augmentation scheme for all training images as in PS-KD, i.e., random cropping and horizontal flipping. We use the stochastic gradient descents (SGD) as the optimizer with momentum 0.9 and weight decay 5e−4 during training. The learning rate starts from 0.1 and is divided by 10 at 150th and 225th epochs, for a total of 300 epochs. The mini-batch size is set to 128. We set β_{max} and β_{min} to 0.9 and 0.0 in Eq. (7), which are the best settings validated in the following ablation studies based on ResNet-18. For ResNet-18, α_1 and α_2 is set to 0.2 and 0.8 in Eq. (9).

For ImageNet-2012, the initial learning rate is set to 0.1 and decayed by a factor of 10 at the 30th, and 60th epochs. The total training epoch is 90 and weight decay is set to 1e−4. The mini-batch size is 256 on 8 NVIDIA 2080Ti GPUs. We set α_1 and α_2 to 0.9 and 0.1. λ_0 is set to 4. The hyper-parameter β_{max} and β_{min} is set to 1.0 and 0.8.

Table 2
Network efficiency comparison of existing structure-based self-distillation approaches on the CIFAR-100 dataset. "Time": The time required for one epoch of training. The comparison is performed among the methods which adopt the auxiliary branches or structures. Our method shows clear advantages in both network complexity and training time.

Models	Metric	Baseline	BYOT	DKS	FRSKD	Tf-KD	DTSKD
VGG-16	MACs	0.32G	0.32G	0.45G	0.40G	0.64G	**0.32G**
	Time	13 s	16 s	18 s	30 s	17 s	**14 s**
ResNet-18	MACs	0.56G	0.60G	1.25G	0.95G	1.12G	**0.56G**
	Time	22 s	29 s	38 s	61 s	28 s	**23 s**
ResNeXt-18	MACs	1.11G	1.15G	2.47G	1.50G	2.22G	**1.11G**
	Time	35 s	38 s	61 s	70 s	44 s	**37 s**

4.2. Image classification

CIFAR-100. Table 1 reports the comparison of six heterogeneous networks with different capacities trained by state-of-the-art distillation methods on the CIFAR-100 dataset. For BYOT and DKS, we report the accuracy of the main backbone network. The baseline represents the performance of the vanilla model after training. From this table, we can see that the recently proposed method MixSKD [47] achieves the second-best results on various models. By utilizing information from two different modalities for distillation, our method successfully achieves the best results on the CIFAR-100 dataset. Specifically, the performance of our method on two important networks, ResNet-18-18 and VGG-16, is 0.24% and 0.62% higher than MixSKD, respectively. Compared with PS-KD, a method that only uses historical information for distillation, our method shows obvious advantages, being 1.35% and 1.60% higher on ResNet-18 and VGG-16.

Fig. 4 shows an illustrative comparison of top-1 accuracy curves at different epochs of training two backbones (VGG-16 and ResNet-18) with three different training methods (Baseline, PS-KD, and DTSKD). From this figure, we can see that DTSKD, the one represented by the red line, is always higher than the other two methods during the training process, and finally converges to the best result.

To quantitatively evaluate the computational complexity, we compare the network efficiency (MACs and Time) among recent knowledge distillation approaches that both adopt the auxiliary branches/structures, as shown in Table 2. "Time" is evaluated on a GeForce 2080Ti GPU based on the standard training settings. Note that Tf-KD$_{self}$ [46] follows a classic two-stage distillation paradigm. It first pre-trains a student in a normal way and then uses the pre-trained student to supervise the training of another student network. So it requires two networks and has twice the MACs of the baseline network in training.

In terms of MACs values, due to our straightforward auxiliary branch design, our computational complexity is close to the baseline method. For methods such as DKS and FRSKD, which introduce complex branch structures, their calculation costs are 133% and 69% higher than the baseline respectively on ResNet-18. In terms of training time, our method brings almost no additional time increase, and only increases by about 1 s compared to the baseline method. Previous methods, however, require longer training times per epoch, ranging from 28 s to 61 s on ResNet-18. According to the above results, our

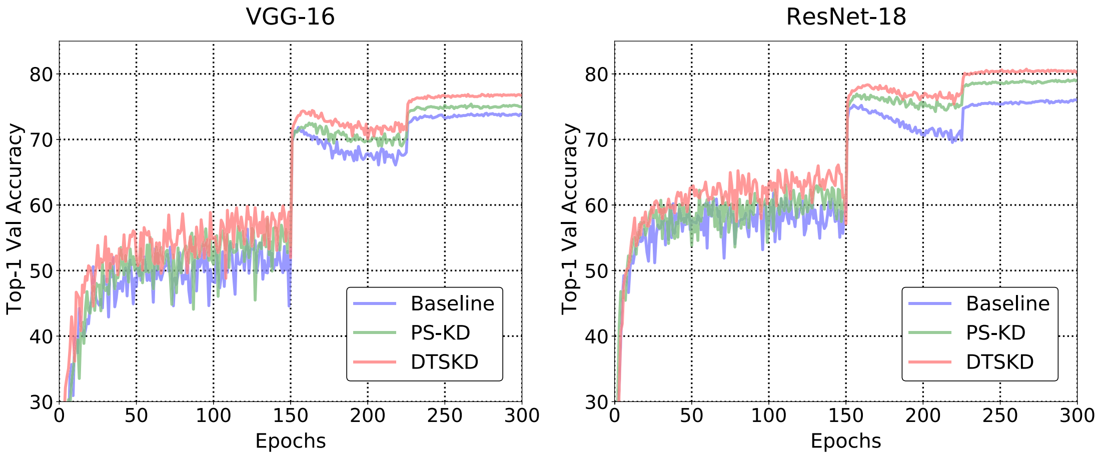


Fig. 4. Curves of top-1 accuracy (%) of the VGG-16 and ResNet-18 networks trained on the CIFAR-100 dataset. Compared to the independent training method (Baseline) and PS-KD, DTSKD shows stably better performance during the whole training procedure and finally converges with the best accuracy. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

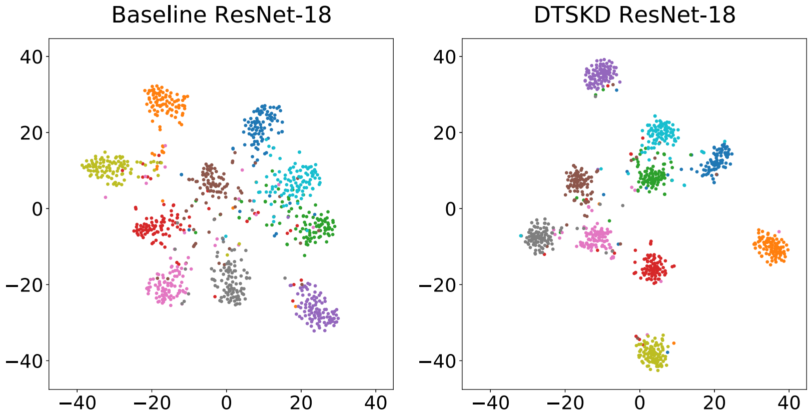


Fig. 5. Visualization of t-SNE for both Baseline ResNet-18 and DTSKD ResNet-18 on the CIFAR-100 dataset. The representations of our method are more separable than the Baseline method, proving that Our DTSKD benefits the discriminability of deep features.

method is more efficient in both model complexity and calculation time and achieves higher performance than previous methods.

We present the visualization results of t-Distributed Stochastic Neighbour Embedding (t-SNE) for both Baseline and DTSKD output logits on the CIFAR-100 dataset, as illustrated in Fig. 5. This figure shows that our DTSKD method obtains more clustered clusters than the baseline method, and maintains a longer distance between clusters. This shows that our method learns more discriminative features than the baseline method.

ImageNet-2012. To further verify the effectiveness of our method on large-scale data sets, we conduct several experiments on ImageNet-2012 using two networks with different architectures as depicted in Table 3. Note that the official implementation of PS-KD uses additional data augmentation methods to train students. In this experiment, we use standard data augmentation methods to re-implement PS-KD for fair comparison. From this table, we can see that our DTSKD consistently outperforms the current state-of-the-art self-distillation approaches, achieving a top-1 validation accuracy of 70.39% and 72.87% with ResNet-18 and VGG-16, respectively.

Table 3
Top-1 validation accuracy (%) comparison of various distillation approaches on the ImageNet-2012 dataset. Our DTSKD achieves the best results with two different models.

Models	Baseline	FRSKD	Tf-KD	PS-KD	DTSKD
ResNet-18	69.71	70.23	70.11	69.94	**70.39**
VGG-16	72.40	72.73	72.64	72.87	**72.98**

4.3. Object detection

PASCAL VOC. To verify the impact of the model trained by our method on downstream tasks, we further extend our method to the object detection task. We adopt Faster R-CNN [48] as our basic detection framework. Specifically, Faster R-CNN is a single-stage model that is trained end-to-end. It uses a novel region proposal network (RPN) for generating region proposals, which saves time compared to traditional algorithms like selective search. The Faster R-CNN detector consists of a CNN backbone, an ROI pooling layer, and fully connected layers followed by two sibling branches for classification and bounding box regression.

The improvement of detection performance (mAP) is examined by using the ResNet-152 network pretrained by different methods including LS, CS-KD, Tf-KD, PS-KD, and our proposed DTSKD. We follow the same detection implementation of PS-KD [6]. We use the 5k VOC 2007 trainval and 15k VOC 2012 trainval for training, VOC 2007 test for validation. The Faster R-CNN backbone network is fine-tuned for 10 epochs with a mini-batch size of 1, an initial learning rate of 0.001 decayed by a factor of 10 at 5 epochs. As shown in Table 4, our DTSKD significantly improves the object detection performance by 1.46%, 1.60%, 1.65%, and 0.43% of the mean Average Precision (mAP) compared to ResNet-152 with LS, CS-KD, Tf-KD, and PS-KD. These

Table 4
Effect of ResNet-152 as a pretrained backbone network for Faster R-CNN on the PASCAL VOC dataset. Our method generalizes better to downstream object detection tasks.

Models	Baseline	LS	CS-KD	Tf-KD	PS-KD	DTSKD
ResNet-152	78.26	78.44	78.33	78.28	79.50	**79.93**

Table 5
Top-1 accuracy comparison with traditional distillation method on the CIFAR-100 dataset. The one-stage distillation method we proposed exceeds the traditional two-stage distillation method and has higher training efficiency.

Teacher	Student	Baseline	KD	FitNet	AT	SP	DTSKD
ResNet-101	ResNet-18	76.27	78.99	79.25	79.28	79.71	**80.46**

results show that our method can adapt well to downstream tasks and provide excellent pre-trained models for fine-tuning.

4.4. Comparison with traditional distillation

In Table 5, we compare the classification accuracy of our proposed DTSKD with four traditional distillation methods on the CIFAR-100 dataset, including vanilla KD, FitNet [19], AT [20] and SP [49]. According to Table 5, it is shown that our proposed DTSKD surpasses the two-stage traditional distillation method with an additional teacher model. In particular, our method outperforms the SP and AT methods by 0.75% and 1.18% respectively. The traditional distillation method needs to train a cumbersome teacher network at first, then distill the knowledge to the lightweight student network. Such a two-stage learning procedure is complex and time-consuming. Instead, our method trains the student in a one-stage manner, eliminating the need for the pre-trained teacher while we also achieve higher performance than the traditional two-stage approach.

4.5. Ablation study

Historical and Structural Distillation. To verify the effectiveness of each component and demonstrate their complementary performance, we evaluate the distillation accuracy of these two elements when used alone and when used simultaneously in Table 6. The first row indicates the baseline performance. When we only perform historical distillation, we set $\alpha_1 = 1$ and $\alpha_2 = 0$ in Eq. (9). When we only perform structural distillation, we set $\beta_{min} = \beta_{max} = 1$ in Eq. (6). In this situation, we do not set $\alpha_1 = 0$ since each branch in the network still requires supervision from ground-truth labels.

In Table 6, we can see that when only using separate types of knowledge for distillation, the model performance is significantly weaker than the combined method. Distillation using historical information alone all exhibits better performance than distillation using structural information alone on three different models. The fourth line demonstrates that the cooperation of two elements can achieve better performance than a single element. Specifically, the combined method is 1.38% and 0.66% higher than the method using only structural information and the method using only historical information on the ResNet-18 model, respectively. It means that historical information and structural information can be complementary, and their combination allows the network to learn diverse information from different perspectives at the same time.

We compare our method with its structure-based counterpart BYOT when we only perform the structural KD. Our method still outperforms BYOT by about 0.8% on ResNet-18 and ShuffleNetV2. This is because BYOT ignores the quality of the student network. The student branch in the network only shares a limited number of underlying convolutional blocks, resulting in insufficient representation ability. The backbone teacher network cannot obtain sufficient and effective feedback from multiple students during optimization. This table shows that our lightweight approach also has superior distillation performance with a single structural teacher.

Table 6
Ablation study: Top-1 accuracy (%) of different elements distilled with three heterogeneous networks. The combination of historical information and structural information further improves model performance.

Structural KD	Historical KD	VGG-16	ResNet-18	ShuffleNetV2
		73.07	76.27	70.77
	✓	76.15	79.80	73.16
✓		75.27	79.08	73.01
✓	✓	**76.72**	**80.46**	**74.30**

Semantic Gaps Between Stages. We experimentally study the negative self-distillation effect between different stages caused by semantic gaps. To reveal its existence, we conduct several experiments with a three-branch ResNet-18, where each branch consists of a 1 × 1 convolutional block, an average pooling layer, and a fully connected layer. In Table 7, we directly distill the structural knowledge from the backbone network (i.e., the last stage) to shallow branches without any top-down connections. However, directly aligning the predictions between two stages does not work as effectively as we expect. When we use the Backbone #0 to train Branch #1 and #2, it does not bring any improvement to the model. On the contrary, such a distillation process seriously affects the learning process. The model performance dropped from 76.27% to 70.68% and 73.16% respectively. When we use the Backbone #0 to train Branch #3, the distillation operation begins to bring positive effects, and the model performance improves from 76.27% to 77.87%. From these experimental results, we can conclude that subnetworks with different capacities tend to learn different levels of abstract information. Directly performing shallow branch distillation operations will mislead the learning process of the auxiliary branch and pose a negative impact on the backbone network through their shared underlying structures. The performance of the backbone network is even worse than the baseline results for some distillation associations, which indicates the existence of large semantic gaps. The shallower the branch network, the larger the semantic gap with the backbone network.

The results in the last column demonstrate the effectiveness of our method. Through the fast top-down link we introduced, sufficient semantic information can be transmitted to each shallow stage through the feature fusion process, which effectively fills the gap between different stages and brings better distillation performance. Specifically, the method we proposed significantly improved the performance of the "#0→#1#2#3" case, from 74.13% to 79.08%, exceeding the baseline accuracy of 76.27%.

Mixed Fusion Module. MFM consists of both addition and concatenation operators, which sufficiently bridge the gap between the deep stages and shallow auxiliary branches by iteratively fusing multi-stage features from the top-down topology. To further evaluate the effectiveness of our proposed MFM, as shown in Table 8, we conduct our ablation experiments on CIFAR-100 with the following two settings:

(1) MFM-Concat (i.e., UNet Style). Inspired by UNet, the concatenation operation is applied to merge the features from two paths. Another 1 × 1 convolutional layer is added to halve the number of feature channels after the concatenation operation. The concatenation may introduce redundancy when it concatenates too many raw features from the backbone network, resulting in 0.34% and 0.26% performance degradation for VGG-16 and ResNet-18 networks.

(2) MFM-Add (i.e., FPN Style). Similar to FPN, we merge the pooled features F_b from the backbone network and the features F_{b+1} with an addition operation. Simply adding these two features may impede information flow and reduce the VGG-16 and ResNet-18 performance by 0.30% and 0.17%, respectively.

Decay Strategy. Hyper-parameter β controls the composition of historical soft labels and one-hot ground-truth labels in final optimization target $p'_{b,t}(\mathbf{x})$. As shown in Table 9, we compare the performances

Table 7
Ablation study: Illustration of negative distillation effect with ResNet-18 on CIFAR-100. The setting of "#0→ #i" corresponds to distilling the structural knowledge from Backbone #0 to Branch #i, where $i \in \{1, 2, 3\}$. The difference between "#0→#1#2#3" and "DTSKD" is that there is no top-down feature fusion process through MFM in the "#0→#1#2#3" setting. Our DTSKD weakens the impact of semantic gaps on distillation, significantly surpassing the baseline method.

Structural KD	Baseline	#0→#1	#0→#2	#0→#3	#0→#1#2#3	DTSKD (w/o HD)
ResNet-18	76.27	70.68	73.16	77.87	74.13	**79.08**

Table 8
Ablation study: Impact of different connection operations in MFM. Removing one of these operations reduces model performance.

Method	VGG-16	ResNet-18
Baseline	73.07	76.27
MFM-Concat (i.e., UNet Style)	76.38	80.20
MFM-Add (i.e., FPN Style)	76.42	80.29
MFM (DTSKD)	**76.72**	**80.46**

Table 9
Ablation study: Different hyper-parameter decay strategies. Fixed ratio distillation performs poorly. The Cosine Decay strategy works best for our method.

Method	VGG-16	ResNet-18	ShuffleNetV2
Baseline	73.07	76.27	70.77
Fixed Ratio (0.6)	75.43	79.55	74.07
Fixed Ratio (0.7)	75.81	79.48	73.95
Fixed Ratio (0.8)	75.33	79.61	73.61
Linear Decay	76.13	80.04	74.12
Cosine Decay (ours)	**76.72**	**80.46**	**74.30**

Table 10
Ablation study for decay-range grid search conducted with ResNet-18 on the CIFAR-100 dataset. The setting of [0.9, 0.0] achieves the best results.

[β_{max}, β_{min}]	[1.0, 0.0]	[0.9, 0.0]	[0.8, 0.0]	[0.7, 0.0]	[0.6, 0.0]
Top-1 Acc	80.28	**80.46**	80.45	80.32	80.29
[β_{max}, β_{min}]	[1.0, 0.1]	[1.0, 0.2]	[1.0, 0.3]	[1.0, 0.4]	[1.0, 0.5]
Top-1 Acc	80.07	80.24	79.62	79.42	79.46

Table 11
Ablation study: Top-1 classification accuracy (%) comparison of different loss weights of Historical Distillation (HD) and Structural Distillation (SD) on the CIFAR-100 dataset. Our DTSKD has robust performance against different loss weights.

HD/SD	0.2/0.8	0.4/0.6	0.6/0.4	0.8/0.2
VGG-16	76.68	76.44	**76.72**	76.50
ResNeXt-18	81.14	81.22	81.27	**81.49**

of different hyper-parameter decay strategies. The setting of "Fixed Ratio (0.6)" indicates that we fix the β_t value in Eq. (6) to 0.6 and do not perform any decay strategy during training. This means that the sum ratio of soft and hard labels remains constant during training. We also compare against the linear decay strategy, as adopted in [6]. In the fixed-ratio experiment, the results reveal that for multiple different networks, the optimal ratio values are not the same. A ratio of 0.6 is optimal for the VGG-16 network, but sub-optimal for the other two networks. By using the dynamic proportion strategy, we observe that the model performance is clearly better than the fixed proportion case, whether it is a linear or cosine strategy. In the following comparison of the two dynamic methods, the cosine strategy is consistently better than the linear strategy, improving by 0.59%, 0.42%, and 0.18% on VGG-16, ResNet-18, and ShuffleNetV2 respectively. Note that in this experiment, we set β_{max} and β_{min} to 0.9 and 0.1 by default.

Decay Range. In Table 10, we demonstrate the results of using different ranges for the cosine decay strategy. The setting of "[1.0, 0.0]" indicates that the initial hyper-parameter β_{max} and β_{min} are set to 1.0 and 0.0, and the β value gradually decreases from 1.0 to 0.0. In this table, we conduct experiments in which β_{max} is fixed and β_{min} is changed, and vice versa. When β decays to 0, it means that model training completely relies on the historical soft labels. When β is 1, it means that model training relies entirely on the ground truth labels. From this table, we can see that the setting of "[0.9, 0.0]" achieves the best performance, which is 0.18% higher than the result of "[1.0, 0.0]". At the same time, in the fourth row of the table, we can observe that the proportion of soft labels has a clear impact on the final performance of the model. Having a higher soft label ratio has better performance than a lower ratio.

Loss Weights. In our proposed method, we have two loss terms: historical distillation loss and structural distillation loss. Table 11 demonstrates how the performance of our proposed method is affected by the choice of loss weights. We demonstrate the results of all the weight ratios at 0.2 intervals in Table 11. We can see that the optimal ratio is not the same for different networks. But our method can still have robust performance for different loss weights, ranging from 0.2 to 0.8.

4.6. Branch comparison with BYOT

Branch-based methods usually introduce multiple auxiliary branches to facilitate the learning of the backbone network by sharing the structural backbone. The target backbone network can make improvements by learning from the feedback of multiple hierarchical branches through their shared backbone features. BYOT also introduces multiple student branch networks that share the underlying backbone network.

We compare the classification accuracy of branch networks between our proposed DTSKD (w/o Historical Distillation) and BYOT, as shown in Table 12. In BYOT, Branch #1 has the poorest performance since it only shares the backbone network's shallowest layer. But in our DTSKD, through our proposed top-down pathway, rich semantic information can be transmitted to the bottom, so that even the shallowest student network can still get better classification accuracy. Our method is 10.59%, 4.47%, and 1.57% higher than BYOT on Branches #1, #2, and #3 respectively. This table illustrates the effectiveness of the top-down pathway in our approach.

4.7. Discussion

Theoretical explanations. (1) Historical Distillation. The Historical Distillation shares great similarity with contrastive learning [21]. Contrastive-based methods employ a distance loss to make two positive augmented samples have similar predictions based on the current model. The process is similar to Historical Distillation, except that one of the two input flows comes from the recorded predictions of historical model parameters, instead of the current model. The effectiveness is also confirmed in R-Drop, where optimizing the consistency of two forward passes with different sub-networks (usually achieved by Dropout) can have regularization benefits. In our case, the two different architectures can be regarded as the previous and current models, respectively.

(2) Structural Distillation. For visually similar categories, deep layers in the network tend to learn more discriminative representations than shallow layers. During structural distillation, high-level guidance from deep layers enables shallow layers to obtain more category information and pay more attention to the details, resulting in better representation ability. Fig. 6 is an example of how our DTSKD improves prediction accuracy. The bottom image whose label is "Catamaran"

Table 12
Top-1 accuracy (%) comparison of auxiliary branches on the CIFAR-100 dataset. Our DTSKD achieves higher accuracy than BYOT on all auxiliary branches.

Network	Method	Branch #1	Branch #2	Branch #3	Backbone #0
ResNet-18	BYOT	68.53	74.70	77.65	78.08
	DTSKD (w/o HD)	79.12	79.17	79.22	79.23

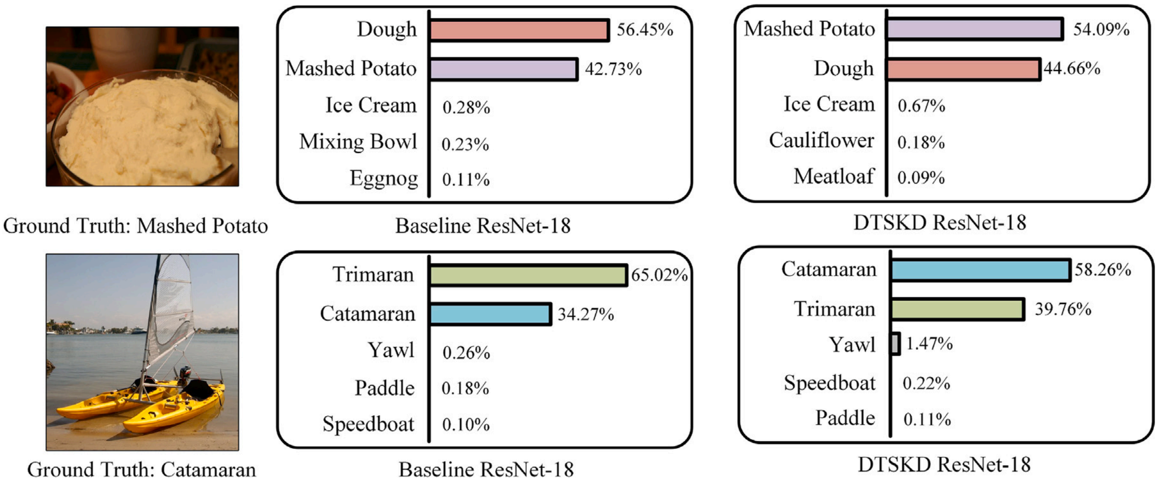


Fig. 6. Top-5 predicted probabilities for samples in ImageNet-2012. Compared with the baseline method, our method can identify similar categories better.

has very similar visual characteristics to the "Trimaran". The baseline network incorrectly predicted this image as a "Trimaran", with 65.02% confidence. But our DTSKD successfully captures the visual nuances and produces correct predictions. It has 58.26% confidence in the Catamaran category, which is higher than the 39.76% in the Trimaran category

Limitations. While our approach has demonstrated remarkable achievements in the realm of self-knowledge distillation, it is crucial to acknowledge the existence of certain limitations within our methodology. First and foremost, our method necessitates the maintenance of a substantial matrix to store the historical soft labels associated with each sample. This requirement for a large memory footprint can pose challenges, particularly in scenarios involving extensive datasets, as it can significantly augment the memory demands and, consequently, exacerbate the complexity of model training. As a prospective avenue for further research, addressing this issue involves finding efficient ways to leverage historical information with minimal memory overhead.

The second limitation of our approach revolves around the inherent intricacies related to model training, notably the presence of multiple hyperparameters that necessitate meticulous selection. This intricacy can potentially limit the versatility and adaptability of our method when applied to diverse scenarios, such as varying backbone networks and datasets. Striking a balance between hyperparameter selection and achieving optimal performance remains an ongoing challenge that warrants attention in future research endeavors.

Furthermore, the third limitation pertains to the fact that, despite our method's significant gains in self-distillation, it still lags behind the performance levels achievable through the two-stage distillation approach. In the latter, an independent, pre-trained teacher model can furnish more extensive and diversified supervisory information to facilitate the training of a superior student model. This discrepancy in performance is inherently tied to the constraints of the self-distillation framework itself, highlighting the potential for further innovation in this domain.

5. Conclusions

In this paper, we propose an efficient self-knowledge distillation approach, named Dual Teachers for Self-Knowledge Distillation (DTSKD), which unifies structural and historical distillation into one framework, allowing the network to learn diverse information from two different perspectives. The lightweight structure design is adopted in our multi-branch framework, enabling the target network to receive supervision from both structural and historical soft labels. During distillation, the semantic gaps between different stages are bridged by iteratively fusing multi-stage features from the top-down topology through our proposed Mixed Fusion Module(MFM). Through our careful design, our proposed DTSKD successfully achieves state-of-the-art performance among existing self-distillation methods and is more efficient than previous counterparts. Extensive experiments show that a variety of deep networks can benefit from our DTSKD approach on three popular datasets: CIFAR-100, ImageNet-2012, and PASCAL VOC.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities, China 070-63233084, the Young Scientists Fund of the National Natural Science Foundation of China (Grant No. 62206134), the National Natural Science Fund of China (No. 62361166670, No. 62072150), Zhejiang Provincial Natural Science Foundation (No. LZ21F020008), and the Tianjin Key Laboratory of Visual Computing and Intelligent Perception (VCIP). Computation is supported by the Supercomputing Center of Nankai University (NKSC).

References

[1] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, 2015, arXiv preprint arXiv:1503.02531.

[2] J. Yim, D. Joo, J. Bae, J. Kim, A gift from knowledge distillation: Fast optimization, network minimization and transfer learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4133–4141.

[3] Y. Liu, W. Zhang, J. Wang, Adaptive multi-teacher multi-level knowledge distillation, Neurocomputing 415 (2020) 106–113.

[4] J. Song, Y. Chen, J. Ye, M. Song, Spot-adaptive knowledge distillation, IEEE Trans. Image Process. 31 (2022) 3359–3370.

[5] Z. Li, X. Li, L. Yang, B. Zhao, R. Song, L. Luo, J. Li, J. Yang, Curriculum temperature for knowledge distillation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2023, pp. 1504–1512.

[6] K. Kim, B. Ji, D. Yoon, S. Hwang, Self-knowledge distillation with progressive refinement of targets, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6567–6576.

[7] L. Zhang, C. Bao, K. Ma, Self-distillation: Towards efficient and compact neural networks, IEEE Trans. Pattern Anal. Mach. Intell. 44 (8) (2021) 4388–4403.

[8] X. Yu, S. Sun, Y. Tian, Self-distillation and self-supervision for partial label learning, Pattern Recognit. (2023) 110016.

[9] T.-B. Xu, C.-L. Liu, Data-distortion guided self-distillation for deep neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2019, pp. 5565–5572.

[10] D. Sun, A. Yao, A. Zhou, H. Zhao, Deeply-supervised knowledge synergy, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6997–7006.

[11] S.I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, H. Ghasemzadeh, Improved knowledge distillation via teacher assistant, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 5191–5198.

[12] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[13] Z. Tian, C. Shen, H. Chen, T. He, FCOS: A simple and strong anchor-free object detector, IEEE Trans. Pattern Anal. Mach. Intell. 44 (4) (2020) 1922–1933.

[14] R. Dong, Z. Tan, M. Wu, L. Zhang, K. Ma, Finding the task-optimal low-bit sub-distribution in deep neural networks, in: International Conference on Machine Learning, PMLR, 2022, pp. 5343–5359.

[15] J. Chen, S. Bai, T. Huang, M. Wang, G. Tian, Y. Liu, Data-free quantization via mixed-precision compensation without fine-tuning, Pattern Recognit. (2023) 109780.

[16] H. Qin, Y. Ding, M. Zhang, Q. Yan, A. Liu, Q. Dang, Z. Liu, X. Liu, Bibert: Accurate fully binarized bert, in: International Conference on Learning Representations, 2022, pp. 1–24.

[17] H. Qin, X. Zhang, R. Gong, Y. Ding, Y. Xu, X. Liu, Distribution-sensitive information retention for accurate binary neural network, Int. J. Comput. Vis. 131 (1) (2023) 26–47.

[18] G. Xu, Z. Liu, C.C. Loy, Computation-efficient knowledge distillation via uncertainty-aware mixup, Pattern Recognit. 138 (2023) 109338.

[19] A. Romero, N. Ballas, S.E. Kahou, A. Chassang, C. Gatta, Y. Bengio, Fitnets: Hints for thin deep nets, in: International Conference on Learning Representations, 2014, pp. 1–13.

[20] S. Zagoruyko, N. Komodakis, Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer, in: International Conference on Learning Representations, 2016, pp. 1–13.

[21] Y. Tian, D. Krishnan, P. Isola, Contrastive representation distillation, in: International Conference on Learning Representations, 2019, pp. 1–19.

[22] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, G.E. Hinton, Big self-supervised models are strong semi-supervised learners, Adv. Neural Inf. Process. Syst. 33 (2020) 22243–22255.

[23] G. Xu, Z. Liu, X. Li, C.C. Loy, Knowledge distillation meets self-supervision, in: European Conference on Computer Vision, Springer, 2020, pp. 588–604.

[24] Z. Huang, S. Yang, M. Zhou, Z. Li, Z. Gong, Y. Chen, Feature map distillation of thin nets for low-resolution object recognition, IEEE Trans. Image Process. 31 (2022) 1364–1379.

[25] Y. Zhang, Y. Zhang, R. Tian, Z. Zhang, Y. Bai, W. Zuo, M. Ding, ThumbDet: One thumbnail image is enough for object detection, Pattern Recognit. 138 (2023) 109424.

[26] R. Tang, Z. Liu, Y. Li, Y. Song, H. Liu, Q. Wang, J. Shao, G. Duan, J. Tan, Task-balanced distillation for object detection, Pattern Recognit. 137 (2023) 109320.

[27] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, J. Wang, Structured knowledge distillation for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2604–2613.

[28] Y. Feng, X. Sun, W. Diao, J. Li, X. Gao, Double similarity distillation for semantic image segmentation, IEEE Trans. Image Process. 30 (2021) 5363–5376.

[29] Z. Li, J. Ye, M. Song, Y. Huang, Z. Pan, Online knowledge distillation for efficient pose estimation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 11740–11750.

[30] Y. Choi, M. El-Khamy, J. Lee, Dual-teacher class-incremental learning with data-free generative replay, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3543–3552.

[31] K. Li, S. Wang, L. Yu, P.-A. Heng, Dual-teacher: Integrating intra-domain and inter-domain teachers for annotation-efficient cardiac segmentation, in: Medical Image Computing and Computer Assisted Intervention, Springer, 2020, pp. 418–427.

[32] N. Dong, Y. Zhang, M. Ding, G.H. Lee, Bridging non co-occurrence with unlabeled in-the-wild data for incremental object detection, Adv. Neural Inf. Process. Syst. 34 (2021) 30492–30503.

[33] W. Son, J. Na, J. Choi, W. Hwang, Densely guided knowledge distillation using multiple teacher assistants, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9395–9404.

[34] C. Wang, D. Chen, J.-P. Mei, Y. Zhang, Y. Feng, C. Chen, SemCKD: Semantic calibration for cross-layer knowledge distillation, IEEE Trans. Knowl. Data Eng. (2022) http://dx.doi.org/10.1109/TKDE.2022.3171571.

[35] S. Yun, J. Park, K. Lee, J. Shin, Regularizing class-wise predictions via self-knowledge distillation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13876–13885.

[36] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, Born again neural networks, in: International Conference on Machine Learning, PMLR, 2018, pp. 1607–1616.

[37] C. Yang, L. Xie, C. Su, A.L. Yuille, Snapshot distillation: Teacher-student optimization in one generation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2859–2868.

[38] M. Ji, S. Shin, S. Hwang, G. Park, I.-C. Moon, Refine myself by teaching myself: Feature refinement via self-knowledge distillation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10664–10673.

[39] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9650–9660.

[40] W. Wang, X. Li, J. Yang, T. Lu, Mixed link networks, 2018, arXiv preprint arXiv:1802.01808.

[41] A. Yao, D. Sun, Knowledge transfer via dense cross-layer mutual-distillation, in: European Conference on Computer Vision, Springer, 2020, pp. 294–311.

[42] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.

[43] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1492–1500.

[44] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

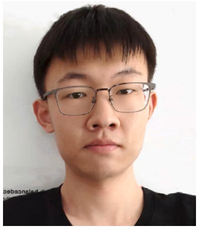
[45] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient cnn architecture design, in: European Conference on Computer Vision, 2018, pp. 116–131.

[46] L. Yuan, F.E. Tay, G. Li, T. Wang, J. Feng, Revisiting knowledge distillation via label smoothing regularization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3903–3911.

[47] C. Yang, Z. An, H. Zhou, L. Cai, X. Zhi, J. Wu, Y. Xu, Q. Zhang, Mixskd: Self-knowledge distillation from mixup for image recognition, in: European Conference on Computer Vision, Springer, 2022, pp. 534–551.

[48] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).

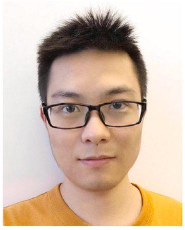
[49] F. Tung, G. Mori, Similarity-preserving knowledge distillation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1365–1374.



Zheng Li is currently a Ph.D. student at Tianjin Key Laboratory of Visual Computing and Intelligent Perception (VCIP), Nankai University, China. His research interests include vision-language models and efficient model computing.

Xiang Li obtained the Ph.D. degree from Nanjing University of Science and Technology, Jiangsu, China, in 2020. His research interests include CNN/Transformer backbone, object detection, knowledge distillation and self-supervised learning. He has published 20+ papers in top journals and conferences such as T-PAMI, CVPR, NeurIPS, etc.

Lingfeng Yang received his B.S. degree from Nanjing University of Science and Technology, China in 2020. He is currently a Ph.D. student at the Department of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include object detection, defect detection, and fine-grained visual categorization.



Renjie Song is currently a researcher at Megvii Research Nanjing. He graduated from Nanjing University of Science and Technology, China in 2015, and then received Master's degree from Nanjing University, China in 2018. His current research interests include deep learning and computer vision.

Jian Yang received the PhD degree from Nanjing University of Science and Technology (NUST) in 2002, majoring in pattern recognition and intelligence systems. From 2003 to 2007, he was a Postdoctoral Fellow at the University of Zaragoza, Hong Kong Polytechnic University and New Jersey Institute of Technology, respectively. From 2007 to present, he is a professor in the School of Computer Science and Technology of NUST. Currently, he is also a visiting distinguished professor in the College of Computer Science of Nankai University. He is the author of more than 300 scientific papers in pattern recognition and computer vision. His papers have been cited over 40000 times in the Scholar Google. His research interests include pattern recognition and computer vision. Currently, he is/was an associate editor of Pattern Recognition, Pattern Recognition Letters, IEEE Trans. Neural Networks and Learning Systems, and Neurocomputing. He is a Fellow of IAPR.

Zhigeng Pan, Dean, School of Artificial Intelligence, Nanjing University of Information Science and Technology, China. He received the Ph.D Degree in 1993 from Zhejiang University. He has published more than 200 papers on international journals, national journals and international conferences. His research interests include virtual reality, intelligent system and HCI. Currently, he is a member of SIGGRAPH, IEEE. He is on the director board of the International Society of VSMM (Virtual System and Multimedia), a member of IFIP Technical Committee on Entertainment Computing (acting as representative from China).