Multi-instance semantic similarity transferring for knowledge distillation

Haoran Zhao [a,b], Xin Sun [c,*], Junyu Dong [b,*], Hui Yu [d], Gaige Wang [b]

[a] School of Information and Control Engineering, Qingdao University of Technology, Qingdao, PR China
[b] College of Information Science and Engineering, Ocean University of China, Qingdao, PR China
[c] Department of Aerospace and Geodesy, Technical University of Munich, Germany
[d] School of Creative Technologies, University of Portsmouth, Portsmouth, UK

Knowledge distillation is a popular paradigm for learning portable neural networks by transferring the knowledge from a large model into a smaller one. Most existing approaches enhance the student model by utilizing the similarity information between the categories of instance level provided by the teacher model. However, these works ignore the similarity correlation between different instances that plays an important role in confidence prediction. To tackle this issue, we propose a novel method in this paper, called multi-instance semantic similarity transferring for knowledge distillation (STKD), which aims to fully utilize the similarities between categories of multiple samples. Furthermore, we propose to better capture the similarity correlation between different instances by the mixup technique, which creates virtual samples by a weighted linear interpolation. Note that, our distillation loss can fully utilize the incorrect classes similarities by the mixed labels. The proposed approach promotes the performance of student model as the virtual sample created by multiple images produces a similar probability distribution in the teacher and student networks. Experiments and ablation studies on several public classification datasets including CIFAR-10, CIFAR-100, CINIC-10 and Tiny-ImageNet verify that this light-weight method can effectively boost the performance of the compact student model. It shows that STKD has substantially outperformed the vanilla knowledge distillation and achieved superior accuracy over the state-of-the-art knowledge distillation methods.

1. Introduction

Deep convolutional neural networks (CNNs) have made unprecedented advances in a wide range of computer vision applications such as image classification [1–3], object detection [4,5] and semantic segmentation [6–8]. However, these top-performing neural networks are usually developed with large depth, parameters and high complexity, which consume expensive computing resources and make it hard to be deployed on low-capacity edge devices. With the increasing demands for low cost networks and real-time response on resource-constrained devices, there is an urgently need for novel solutions that can reduce model complexities while keeping decent performance.

To tackle this problem, a large body of works has been proposed to accelerate or compress these deep neural networks in recent years. Generally, these solutions fall into the following perspectives: network pruning [9–11], network decomposition [12–14], network quantization and knowledge distillation [15,16]. Among these methods, the seminal work of knowledge distillation has attracted a lot of attention due to its ability of exploiting dark knowledge from the pre-trained large network.

Knowledge distillation (KD) is proposed by Hinton et al. [15] for supervising the training of a compact yet efficient student model by capturing and transferring the knowledge of a large teacher model to a compact one. Its success is attributed to the knowledge contained in class distributions provided by the teacher via soften softmax. Many follow up works [17–21] have been proposed since then focusing on different categories of knowledge with various kinds of distillation loss functions in the field of CNNs optimization.

Nevertheless, most existing methods extract some specific layer's outputs of the teacher network as knowledge based on individual instance level. The knowledge contained in the correlation of different instances is still not considered yet, which is the key motivation of our investigation in this paper. Our hypothesis is that the correlation between different instances is also valuable knowledge for promoting the performance of student.

This work was supported in part by the National Natural Science Foundation of China (No. 61971388, U1706218) and Alexander von Humboldt Foundation.

* Corresponding authors.
E-mail addresses: zhaohaoran@stu.ouc.edu.cn (H. Zhao), sunxin1984@ieee.org (X. Sun), dongjunyu@ouc.edu.cn (J. Dong), hui.yu@port.ac.uk (H. Yu), wgg@ouc.edu.cn (G. Wang).

In vanilla KD [15], the knowledge of teacher network is transferred to the student network by mimicking the class probabilities outputs, which are softened by setting a temperature hyperparameter in softmax. Inspired by KD, following works introduce the output of intermediate layers as knowledge to supervise the training of the student network. For example, FitNets [17] first introduces the intermediate representation as the hints to guide the training of the student network, which directly matches the feature maps of intermediate layers between the teacher network and the student network. Later, Zagoruyko et al. [18] introduce attention maps (AT) from the feature maps of intermediate layers as knowledge. FSP [19] designs a flow distillation loss to encourage the student to mimic the teacher's flow matrices within the feature maps between two layers. Recently, Park et al. [20] propose a relational knowledge distillation (RKD) method which draws mutual relations of data examples by the proposed distance-wise and angle-wise distillation losses. In particular, SP [22] preserves the pairwise similarities in student's representation space instead to mimic the representation space of the teacher. Besides, various approaches extend these works by matching other statistics, such as gradient [23] and distribution [24].

Recently, there have been some attempts to extend KD to other domains where KD also shows its potential. For example, Papernot et al. [25] introduce the defensive distillation for adversarial attack to reduce the effectiveness of adversarial samples on CNNs. Gupta et al. [26] transfer the knowledge among the images from different modalities. Besides, knowledge distillation can be employed to some task-specific methods such as object detection [27–29], semantic segmentation [30,31], face model compression [32,33], action recognition [34,35] and depth estimation [36].

From a theoretical perspective, Yuan et al. [37] interpret knowledge distillation in terms of Label Smoothing Regularization [38] and find the importance of soft targets regularization. They propose to manually design the regularization distribution to instead the teacher network. Notably, Hinton et al. [39] investigate the effect of label smoothing to knowledge distillation and observe that label smoothing can alter the performance of distillation when the teacher model is trained with label smoothing. Through visualization of penultimate layer's activation of CNNs, they find that the label smoothing could discard the similarity information of categories, causing poor distillation results. Thus, the similarity information of categories is vital for knowledge distillation. But as existing methods focusing on the investigation of knowledge distillation depending on the similarity of instance level, similarity information of categories has been ignored.

To fill this knowledge gap, we aim to exploit the privileged knowledge on similarity correlation between different instances using virtual samples, created by mix-samples. Generally speaking, the classification confidence represents over-confident predictions when the high-capacity teacher model has the excellent performance. Consequently, the incorrect classes contain low confidence which has low similarity information among categories. As shown in Fig. 1, the teacher from the Vanilla Knowledge Distillation [15] outputs over-confident prediction for the correct category and low confidence for incorrect classes. However, this category similarity information is exactly the most important knowledge that should be transferred to the student. For example, the probability of incorrectly classifying a cat as a car must be lower than the probability of misclassifying it as a dog. Thus, we argue that the image labels are more informative for the specific images. In other words, the teacher should guide the student to distinguish what the image is and what it looks like.

In this work, we employ the mixup [40] to create our virtual training samples by a weighted linear interpolation. Actually, Zhang et al. [40] propose the Mixup method as a data augmentation principle, which trains on virtual examples constructed as the linear interpolation of two random examples from the training set and their labels. However, we aim to present the similarity correlation between different instances by this technology. In this way, the virtual samples with mix-labels can provide additional intra- and inter-class relations in datasets. In contrast to existing approaches, the proposed method transfers the similarity correlation between different instances instead of similarity information of individual samples. It means that we force the virtual samples created by multiple mages to produce a similar probability distribution in the teacher and student networks. Due to the soften probability distribution from virtual samples, the teacher's knowledge could be easily learned by the compact student model that can further improve its robustness. We have pointed out that the core scientific issue of this work is to distill the knowledge efficiently. Thus, we propose to fully utilize the similarity correlation between instances. And the mixup technology has been employed as the tool to present such correlation information. In other words, the main contribution of the proposed method is that we have introduced to fully utilize he similarity correlation between different instances to supervise the training of the student network with a pre-trained teacher network.

In summary, our contributions in this paper can be summarized as follows:

- We propose a novel distillation framework named multi-instance semantic similarity transferring for knowledge distillation (STKD), which utilizes the similarity correlation between different instances to supervise the training of the student network with a pre-trained teacher network. It is the first time that the knowledge contained in the correlation of different instances is considered for distillation.
- We introduce the mixup technique to create the virtual samples, which could present the similarity correlation between different instances by the probability distribution outputs.
- Extensive experiments conducted on various public datasets such as CIFAR-10/100, CINIC-10 and Tiny-ImageNet demonstrate that our approach significantly outperforms the vanilla knowledge distillation and other SOTA approaches.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. And we present the proposed similarity transfer for knowledge distillation architecture in Section 3. Experimental results are presented in Section 4. Finally, Section 5 concludes this paper.

2. Related works

In this section, we first briefly introduce the backgrounds of network model compression and acceleration and then summarize existing works on knowledge distillation. Finally, we review existing works related to the mixup technique.

2.1. Model compression and acceleration

Recently, CNNs become predominant in many computer vision tasks [41–44]. Network architectures become deeper and wider which usually achieve better performances than the shallow ones. And the computational costs increase accordingly due to the increase of parameters of the deep networks, which hinders the real-time applications on mobile devices. Thus, there are demands for model compression and acceleration to reduce the model size and computation cost meanwhile maintaining high performance. One line of works directly designs small network structures by making modifications on the convolution operation strategy to reduce redundant parameters on the original deep

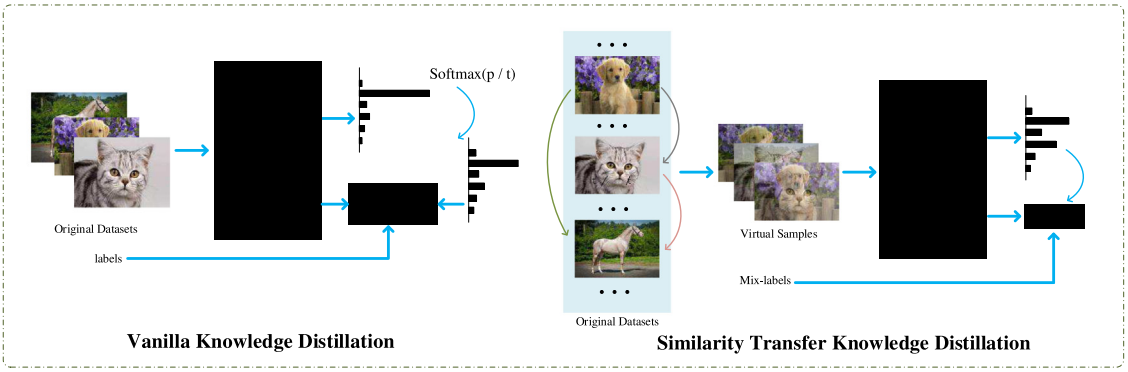

Fig. 1. The whole framework of our Similarity Transfer Knowledge Distillation vs. Vanilla Knowledge Distillation. STKD extracts more similarities information among categories by feeding the virtual samples with original samples of mixed labels. As shown in the right, STKD outputs more soften probability distributions from the teacher and transfers the similarities information to the student.

models. For example, MobileNet [45] introduces depth-wise separable convolution to build blocks instead of standard convolution. ShuffleNet [46] proposes point-wise group convolution and channel shuffle to maintain a decent performance without adding extra computation costs. These approaches focus on physically reducing internal redundancy of the model to obtain a compact network architecture. However, how to train the reduced network with high performance is yet an unresolved issue.

Besides, there has been another kind of method which attempts to remove the redundant information of teacher network. For example, network pruning aims to boost the speed of inference by getting rid of redundancy of the large CNN model. Han et al. [47] propose to boost the speed of inference by deleting unimportant connections. Network decomposition use matrix decomposition techniques to decompose the original convolution kernel in CNNs model. For example, Novikov et al. [48] propose to reduce the number of parameters while preserve the original performance of network by converting the dense weight matrices of the fully-connected layers to the Tensor Train format. Network quantization methods aim to represent the float weights with fewer bits. Yang et al. [49] propose to simulate the quantization process with a differentiable function. However, network pruning methods require many iterations to converge and the pruning threshold needs to be set manually.

2.2. Knowledge distillation

Different from above methods, knowledge distillation enriches and gets the student model by extracting kinds of knowledge from the fixed teacher model. To address the challenge of deploying CNNs in resource-constrained edge devices, Bucila et al. [50] first propose to transfer the knowledge of an ensemble of models to a small model. Then Caruana et al. [51] propose to train the student model by mimicking the teacher model's logits. Later, Hinton et al. [15] popularize the idea of knowledge distillation, which efficiently transfers knowledge from a large teacher network to a compact student network by mimicking the class probability outputs. Note that, the outputs of the teacher network are defined as the dark knowledge in KD, which provides similarity information as extra supervisions compared with one-hot labels. In other words, there are two sources of supervision used to train the student in KD with one from the ground-truth label, and the other from the soft targets of the teacher network.

Afterwards, some recent works [17,18,52] extend KD by distilling knowledge from intermediate feature representations instead of soft labels. For example, FitNets [17] propose to train the student network by mimicking the intermediate feature maps of the teacher network, which are defined as hints. Inspired by this, Zagoruyko et al. [18] propose to match the attention maps between the teacher and the student, which are defined from the original feature maps as knowledge. Wang et al. [53] propose to improve the performance of the student network by matching the distributions of spatial neuron activations between the teacher and the student. Recently, Heo et al. [54] introduce the activation boundary of the hidden neuron as knowledge for distilling the compact student network.

However, the aforementioned knowledge distillation methods only utilize the knowledge contained in the output of specific layers of the teacher network. More knowledge between different layers has been explored and utilized for knowledge distillation. For example, Yim et al. [19] propose to use Gram matrix between different feature layers as distilled knowledge, which is named flow of solution process (FSP) reflecting the relations of different features maps. Lee et al. [55] propose to extract valuable correlation information as knowledge between different feature maps using singular value decomposition (SVD). However, these methods only focus on the knowledge contained in the individual data samples but ignore the similarities between categories of multiple samples, which is also valuable for knowledge distillation.

Some very recent works [20,22] aim to explore the relationship between data samples. Park et al. [20] propose a relation knowledge distillation (RKD) method, which extracts mutual relations of data samples by the proposed distance-wise and angle-wise distillation losses. Tung et al. [22] propose a similarity preserving (SP) knowledge distillation method which transfers the similar activations of input pairs from the teacher network to the student network. Peng et al. [56] propose to capture the high order correlation between samples using the kernel method. However, the similarities information between instances is hard to learn for the student network due to the high dimension in the embedding space. Different from existing works, the proposed approach presents the similarity correlation between instances through the probability distribution outputs of virtual samples, which are created by the mixup technique. In contrast to the vanilla KD which softens the softmax outputs using a temperature hyperparameter, our approach directly outputs a softened probability distribution using the virtual samples.

2.3. Mixed samples data augmentation

There are many training strategies to further improve the performance of CNNs, such as data augmentation and regularization techniques. In particular, mixed sample data augmentation has gained increasing attention due to its outstanding performance. It trains the models using the augmented data set by combining data samples.

For example, Zhang et al. [40] first introduce a data augmentation approach named mixup, which could improve the generalization of CNNs by linearly interpolating a random pair of original training data to create a new training dataset and corresponding labels. Inspired by this, some mixup variants [57–59] have been proposed. Later, Cutmix [60] proposes to replace the removed regions with a patch from another image. FMix [61] proposes to use binary masks obtained by applying a threshold to low frequency images sampled from Fourier space. Following them, Pairing Samples [62] propose to take an average of two samples for each pixel. Recently, Cubuk et al. [63] employ automatic searching to improve data augmentation policies called AutoAugment. Although these methods have achieved remarkable performance to the training of CNNs, they suffer from the drawback of prohibitive training time.

3. Proposed method

In this section, we first revisit the definition of vanilla KD [15], which transfers knowledge from the high-capacity teacher network to the student network with soft labels. Then we describe details of the proposed similarity transfer for knowledge distillation. Fig. 1 compares the vanilla KD with STKD and describes the whole framework of our approach.

3.1. Problem definition

In our approach, we aim to explore a more effective knowledge, which contains in the similarity correlation between instances, to guide the training of the student model. Moreover, we employ mixup techniques to represent these knowledge by introducing virtual samples.

For simplicity, we define the large, complex *Teacher* deep neural network as T with learned parameters P_T, the less complex *Student* network as S with learned parameters P_S. The original training data consist of tuples of input data and target $(x, y) \in D$. In general, the conventional knowledge distillation methods train the *Student* by minimizing the following objective function with respect to the parameters P_S over the training samples $(x, y) \in D$:

$$\mathcal{L} = \mathcal{L}_{KD}(S(x, P_S), T(x, P_T)) + \lambda \mathcal{H}(y_s, y) \tag{1}$$

where λ is a balancing hyperparameter, $\mathcal{H}$ is the cross-entropy loss that computes on the predicted label y_s and the corresponding ground truth labels y. And $\mathcal{L}_{KD}$ is the distillation loss such as cross-entropy or mean square error. $T(x, P_T)$ and $S(x, P_S)$ represent the softmax output of the *Teacher* and *Student* respectively.

For example, Hinton et al. [15] use pre-softmax outputs for $T(x, P_T)$ and $S(x, P_S)$, and Kullback–Leibler divergence for $\mathcal{L}_{KD}$:

$$\sum_{x_i \in D} KL(softmax(\frac{T(x_i, P_T)}{t}), softmax(\frac{S(x_i, P_S)}{t})) \tag{2}$$

where t is a temperature hyperparameter to soften the softmax outputs of teacher network. Thus, we need to set a proper t manually. However, it is hard to obtain a proper temperature t due to the gap between the teacher and student networks.

Likewise, other works [17,19] involved an objective function that can also be formulated as a form of Eq. (1). However, these conventional KD methods only focus on the similarity correlation between categories of individual samples while ignore the similarity correlation between different samples. Moreover, the temperature hyperparameter t needs to be set manually which is used to soften the probability distribution.

3.2. Similarity transfer knowledge distillation

In this section, we introduce multi-instance semantic similarity transferring for knowledge distillation in detail. Usually, traditional knowledge distillation methods [15] utilize the temperature hyperparameter to soften the one-hot labels to soft labels. Different from existing works, STKD presents the similarity correlation between different samples through the probability distribution outputs of virtual samples, which are created by the mixup technique.

To be specific, we first employ the recently proposed mixup [40] to create the virtual samples. We randomly sample two samples (x_i, y_i) and (x_j, y_j) from D. Then it generates the new synthetic sample (x, y) by a weighted linear interpolation of these two samples:

$$x = \lambda x_i + (1 - \lambda) x_j \tag{3}$$

$$y = \lambda y_i + (1 - \lambda) y_j \tag{4}$$

where the merging coefficient $\lambda \in [0, 1]$ is a random number drawn from the $Beta(\alpha, \alpha)$ distribution. And y is the label of x which is a convexed combination of the labels from x_i and x_j.

As can be seen from Fig. 1, the virtual samples are created by this mixup technique. In other words, each sample from the virtual set is formed by two arbitrary original images by a weighted linear interpolation.

By Eq. (4), we turn the one-hot labels to mixed labels, which also contribute to the similarity information between different samples. Thus, the mixed labels of the virtual samples are employed in our framework. Therefore, the mix loss can be formulated as:

$$\mathcal{L}_{mix} = \lambda \mathcal{H}(y_{s_i}, y) + (1 - \lambda) \mathcal{H}(y_{s_j}, y) \tag{5}$$

where y_{s_i} and y_{s_j} are the predicted labels corresponding the ground truth labels y. And λ is a hyperparameter as same as the coefficient in the mixup technique. Note that, we supervise the training of the student network by using these mixed labels and the teacher network's logits.

Then we distill the similarity knowledge between multiple samples from the teacher network to the student network using the created virtual samples. Here we define the total loss function for similarity transfer knowledge distillation as follows:

$$\mathcal{L}_{total} = \lambda \mathcal{H}(y_{s_i}, y) + (1 - \lambda) \mathcal{H}(y_{s_j}, y) + \mathcal{L}_{KD}(S(x, P_S), T(x, P_T)) \tag{6}$$

Different from the vanilla KD method, we get rid of the temperature hyperparameter in our method. Thus, the probability distribution outputs of T and S are defined as $T(x, P_T) = softmax(T(x_i, P_T))$ and $S(x, P_S) = softmax(S(x_i, P_S))$ respectively. Note that the classification loss has the hyper-parameter λ to balance the weight of the mix labels.

Unlike previous methods, we define the similarity correlation between instances as knowledge and present it through the probability distribution of the virtual images. To this end, we employ the mixup technique to create virtual samples which is effective for our method. It generates mixup images that not only contains multiple images' similarity correlation, but also regularizes the student model through knowledge distillation to favor the simple linear behavior in-between training samples. Moreover, we could generate different mixup images by varying the coefficient λ. We study this coefficient in the ablation study.

In this way, teacher network transfers more valuable information to the student network than existing methods. That means the teacher's dark knowledge could be better transferred to the student.

3.3. Training procedure

Algorithm 1 describes the details of the whole training paradigm. We first obtain a pre-trained teacher model T with parameters P_T which is trained using standard back-propagation on the original dataset $(x, y) \in D$.

Then we iterate the following stages until the student S converges. (1) We create the virtual set instead of the original dataset using the mixup technique. Specifically, we use a single data loader to obtain the mini-batch and apply the mixup to the same mini-batch after random shuffling. (2) We feed these virtual samples to our knowledge distillation framework, which distills the similarity correlation between instances from the pre-trained teacher to the student network using our STKD loss in Eq. (6).

Fig. 1 illustrates the overall framework of STKD. Different from the previous methods, the proposed STKD regularizes the student network to favor the simple linear behavior under the teacher's guidance. In other words, we employ the mixup technique to provide more similarity information among classes and make the high dimensional representation of teacher more easily transferable to the student. We will demonstrate its effectiveness in the following section.

Algorithm 1 Multi-instance Semantic Similarity Transferring for Knowledge Distillation

Input: A pre-trained teacher model T with parameters P_T.
Input: The original training data $(x, y) \in D$.
Input: Hyper-parameter (learning rate, coefficient λ, etc).
Output: A compact student model S with parameters P_S.

Initialize: The student network S and training hyper-parameters.
Repeat:
  Stage 1: Creating the Virtual Samples by Mixup.
  Sample a batch (x, y) from the training dataset.
  Sample two samples (x_i, y_i) and (x_j, y_j) randomly from the batch.
  Synthetic the virtual samples by Eqs. (3) and (4).
  Stage 2: Training the student model.
  Sample a batch (x, y_i, y_j) from the virtual samples.
  Calculate the softmax outputs of teacher network and student network: $P_t(x)$, $P_s(x)$
  Calculate the knowledge distillation loss in Eq. (6).
  Update weights in P_S according to the gradient.
Until: convergence.

4. Experiments and results

In this section, extensive experiments are conducted to verify the effectiveness of the proposed approach. We first design comparison experiments with state-of-the-art (SOTA) KD methods (e.g., AT [18], SP [22] and RKD [20]) to test our approach. Additionally, different network architectures (e.g., ResNet [64], WideResNet [65], VGG [66]) are explored in ablation studies.

We implement the experiments using PyTorch on Nvidia 1080TI GPU devices. For all the experiments, the results are averaged over 5 trials using different randomly selected seeds.

4.1. Experiment settings

Datasets. To demonstrate our method under general situations of data diversity, we run experiments on CIFAR-10/100, CINIC-10 and Tiny-ImageNet which are popular benchmark datasets for image classification. CIFAR-10 [67] and CIFAR-100 both contain 50K training images and 10K test images. CIFAR-10 has 10 categories when CIFAR-100 contains 100 classes. The CINIC-10 dataset [68] consists of images from both CIFAR-10 dataset and ImageNet dataset, whose scale is closer to ImageNet. It is composed of 270,000 images at a spatial resolution of 32×32 via the addition of down-sampled ImageNet images. We adopt the CINIC-10 dataset for rapid experimentation. Tiny-ImageNet dataset [69] is a popular subset of the ImageNet dataset [1]. It contains 100k images with a resolution of 64×64 in 200 categories for training and 10K additional images for testing. Each class has 500 training images, 50 validation images, and 50 test images.

Network architecture. We use three state-of-the-art convolutional neural network architectures: ResNet [70], Wide Residual Network (WRN) [71] and VGG [72]. For each network family, we employ a deep or wide one as the teacher network and a shallow or thin one as the student network. Note that, WRN has a standard convolutional layer followed by three groups of residual blocks, each of size n. Additionally, it uses an additional widen factor m to increase the width, which could bring more representation ability. We denote WRN as WRN-n-m in our experiment. In the following experiments, we take ResNet-110, WRN-40-2 and VGG-13 as the teacher networks, ResNet-20, WRN-40-1 and VGG-8 as student networks. In addition, we use different network architectures (e.g., ResNet-50 as teacher and VGG-8 as student) in ablation studies.

Evaluation Metric. We adopt the classification accuracy as the evaluation metric. Note that, the accuracy is computed as the median of 5 runs with different seeds.

4.2. Experiment on CIFAR-10/CIFAR-100

On CIFAR dataset, we use three groups of teacher/student model pairs including ResNet-110/ResNet-20, WRN-40-2/WRN-40-1 and VGG-13/VGG-8. We apply a standard horizontal flip and random crop data augmentation scheme which is widely adopted for these datasets. For fair comparison, we keep the same data argumentation and configuration for all the compared methods. We set the weight decay to 10^{-4}., batch size to 128, and use stochastic gradient descent (SGD) with momentum 0.9. The initial learning rate is set to γ = 0.1 and divided by 10 at 80th, 100th, 150th epochs, totally 200 epochs. We use the same setting for training the teacher network and distilling the student network.

We conduct baseline comparisons with respect to the vanilla KD [15], AT [18], SP [22] and RKD [20]. For vanilla KD, we set temperature hyperparameter $t = 4$ following the experiments in [18]. We set the hyperparameter $\beta = 1000$ for AT, $\gamma = 0.1$ for SP respectively.

Table 1 summarizes the classification accuracy of several teacher/student model pairs on CIFAR-10/100 datasets. For the WRN network architecture, the teacher and student networks have the same depth but different width (WRN-40-2 teacher with WRN-40-1 student).

Our method obtains 94.44%, 93.52% and 93.68% of top-1 accuracy for WRN-40-1, ResNet-20 and VGG-8 on CIFAR-10, respectively. It substantially surpasses the vanilla KD by 1.6%, 1,9% and 1.8%. Compared to other baselines, the student network from STKD still significantly surpass the three related SOTA methods. For CIFAR-100, our method also has 1.6%, 1.7% and 1.8% top-1 accuracy over vanilla KD. Moreover, our approach still surpasses the three SOTA distillation related methods, which verifies the effectiveness of our approach. These results validate our intuition that the similarity correlation between instances encodes valuable knowledge from the teacher network and provides an important supervision for knowledge distillation.

Fig. 2 plots the accuracy change curves for WRN-40-2/WRN-40-1 experiments on CIFAR-10. As can be seen from Fig. 2(a), it shows the testing accuracy curves of the teacher network which is trained individually and the student network from our method.

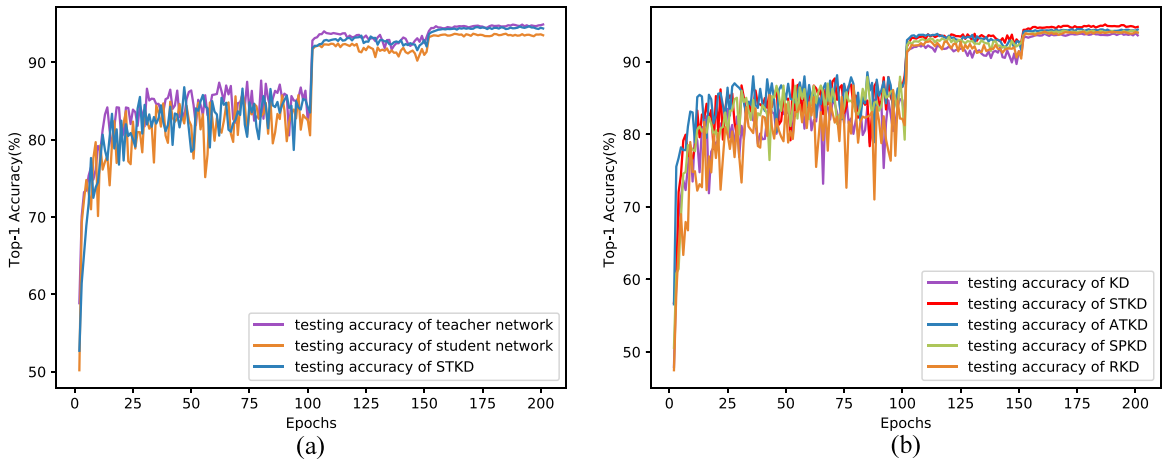

Fig. 2. (a) Testing accuracy of the pre-trained teacher, student from our method and student trains individually. (b) Testing accuracy of different knowledge distillation methods on CIFAR-10.

Table 1
Classification accuracy (%) on CIFAR-10 and CIFAR-100 datasets (5 runs). Baseline means the student network trains individually. STKD means the classification accuracy in our method. We report the standard deviation over 5 runs.

Dataset	Model (S/T)	Baseline	KD [15]	AT [18]	SP [22]	RKD [20]	STKD	Teacher
CIFAR-10	WRN-40-1 (0.56M) WRN-40-2 (2.20M)	93.84	94.17 ± 0.32	93.96 ± 0.31	94.21 ± 0.42	94.06 ± 0.49	**94.44 ± 0.27**	94.94
	ResNet-20 (0.27M) ResNet-110 (1.7M)	92.56	93.08 ± 0.27	93.21 ± 0.35	93.30 ± 0.38	93.20 ± 0.33	**93.52 ± 0.22**	94.51
	VGG-8 (0.27M) VGG-13 (1.7M)	92.76	93.14 ± 0.31	93.29 ± 0.37	93.08 ± 0.42	93.31 ± 0.34	**93.68 ± 0.29**	94.27
CIFAR-100	WRN-40-1 (0.56M) WRN-40-2 (2.20M)	72.08	73.34 ± 0.39	73.77 ± 0.33	73.46 ± 0.32	73.08 ± 0.41	**74.63 ± 0.24**	75.61
	ResNet-20 (0.27M) ResNet-110 (1.7M)	69.14	70.69 ± 0.33	70.97 ± 0.37	71.02 ± 0.48	70.77 ± 0.47	**72.14 ± 0.31**	74.37
	VGG-8 (0.27M) VGG-13 (1.7M)	70.39	72.87 ± 0.29	73.43 ± 0.31	73.49 ± 0.37	73.01 ± 0.39	**73.65 ± 0.29**	74.66

Table 2
Classification accuracy (%) on CIFAR-10 (5 runs). Experimental results on the varying sizes of student models giving the same teacher model under the different knowledge distillation methods. Baseline means the student network trains individually. STKD means the classification accuracy in our method.

Teacher	Student	Baseline	KD [15]	AT [18]	SP [22]	RKD [20]	STKD
WRN-40-2 (2.20M)	WRN-16-1 (0.17M)	91.28	92.30	92.46	92.32	92.38	**93.61**
	ResNet-20 (0.27M)	92.56	93.12	93.09	93.27	93.08	**93.52**
	VGG-8 (0.27M)	92.76	93.19	93.73	93.18	93.01	**93.68**
	WRN-40-1 (0.56M)	93.84	94.17	93.96	94.21	94.06	**94.44**

Table 3
Classification accuracy (%) on CIFAR-10 (5 runs). Baseline means the student network trains individually. STKD(Mixup) means the classification accuracy in our method. STKD(Cutmux) means the novel Cutmix method in STKD.

Teacher	Student	KD [15]	AT [18]	SP [22]	RKD [20]	STKD (Mixup [40])	STKD (Cutmix [60])
WRN-40-2 (2.20M)	WRN-40-1 (0.56M)	94.17	93.96	94.21	94.06	94.44	94.96
ResNet-110 (1.7M)	ResNet-20 (0.27M)	93.08	93.21	93.30	93.20	93.52	94.03
VGG-13 (1.7M)	VGG-8 (0.27M)	93.14	93.29	93.08	93.31	93.68	94.29

Note that, STKD gets a significant improvement compared to the student trained individually and its final accuracy value is close to that of the teacher. Fig. 2(b) shows the validation accuracy change curves of WRN-40-1 over time among different knowledge distillation approaches on CIFAR-10 dataset. We can observe that our method performs a significant improvement on the final accuracy and outperforms the other SOTA approaches.

To further investigate the effectiveness of the proposed method, we conduct additional experiments on the varying sizes of student models giving the same teacher model under different knowledge distillation methods. Table 2 presents the experimental results. We could find that the performances of different sizes of student models under the proposed method achieve superior accuracy over others comparison methods.

In addition, we also employ the novel Cutmix method [60] comparing the Mixup method in STKD. As can be seen from Table 3, the novel technology Cutmix could further improve the accuracy of the proposed method. It means that Cutmix provides more valuable similarities between categories of multiple samples.

From the perspective of the teacher model, our method also shows the potential to compress large networks into more compact ones with minimal accuracy loss. For example, we distill the knowledge from the pre-trained WRN-40-2 teacher network, which contains 2.20M parameters, to a much smaller

Table 4
Classification accuracy (%) on CINIC-10 (5 runs). WRN-40-2 is as the teacher network, WRN-40-1 as the student network. Baseline means the WRN-40-1 trains individually. STKD means the WRN-40-1 results in our method.

Type	Model	Params (M)	Acc (%)
Baseline	WRN-40-1	0.56	84.30
KD	WRN-40-1	0.56	85.04
AT	WRN-40-1	0.56	85.53
RKD	WRN-40-1	0.56	84.96
SP	WRN-40-1	0.56	85.16
STKD	WRN-40-1	0.56	**85.94**
Teacher	WRN-40-2	2.20	86.40

Table 5
Classification accuracy (%) on Tiny ImageNet (5 runs). Baseline means the WRN-40-1 trains individually. STKD means the WRN-40-1 results in our method.

Type	Model	Params (M)	Acc (%)
Baseline	WRN-40-1	0.56	56.93
KD	WRN-40-1	0.56	58.52
AT	WRN-40-1	0.56	58.43
RKD	WRN-40-1	0.56	57.02
STKD	WRN-40-1	0.56	**59.25**
Teacher	WRN-40-2	2.20	62.21

WRN-40-1 student network, which contains 0.56*M* parameters. The student network gets a 4× compression rate with only 0.5% loss in classification accuracy using off-the-shelf Pytorch.

4.3. Experiment on CINIC-10

In this section, we conduct experiments on CINIC-10 dataset, which has a similar scale with ImageNet. We use WRN-40-2 and WRN-40-1 as the teacher network and student network respectively. During the training phase, we apply CIFAR-style data augmentation with horizontal flips and random crops. The learning rate starts with 0.01 and decays by a factor of 10 after the 100th, 160th and 220th epochs. All the networks are trained for 240 epochs using the SGD with Nesterov momentum with a batch size of 96.

For fair comparison, we adopt the same setting as mentioned above for all baseline methods. We set the hyperparameters for the baseline methods ($t = 8$ for KD; $\beta = 100$ for AT, $\gamma = 0.1$ for SP). The results on CINIC-10 are shown in Table 4. Our method achieves a 85.94% classification accuracy, which surpasses the student network trained individually by 1.64%. Moreover, STKD substantially surpasses the vanilla KD by 0.9%. It also shows comparable performance with other SOTA approaches. These results imply that STKD method induces more meaningful dark knowledge such as the similarity information between instances than other baseline methods.

4.4. Experiment on tiny-ImageNet

For rapid experimentation, we conduct experiments on Tiny-ImageNet dataset, which is a popular subset of the ImageNet. We adopt the VGG network architecture for the following experiments. To be specific, VGG-13 is employed for the teacher network, which achieves 62.21% classification accuracy. VGG-8 is used for the student network. Both of the networks are trained for 240 epochs.

In our Tiny ImageNet classification experiments, we apply random rotation and horizontal flipping for data augmentation. We optimize the model using stochastic gradient descent(SGD) with mini-batch 128 and momentum 0.9. The learning rate starts from 0.1 and is multiplied by 0.2 after 60, 120, 160, 200, 250 epochs respectively. We train the network for a total of 300 epochs and adopt the deep and wide WRN (WRN-40-1) for a teacher model and WRN-16-1 as a student model.

In this section, ResNet with four blocks is selected with a channel size of 16, 32, 64 and 128. Random crop and horizontal flip are used for data augmentation. Table 5 shows the results, where baseline denotes the student network trained individually. As can be seen from Table 5, our method, as well as the other SOTA methods, gets higher classification accuracy than the original student network. Moreover, our STKD method achieves better performance than the KD and shows comparable performance with the other knowledge distillation methods.

Table 6
Additional experiments under different network architecture families on CIFAR-100. Classification accuracy (%) over five runs is reported. The best performance is shown in bold. Baseline means the student network trained individually. STKD means the performance of student network in our method.

Teacher Student	ResNet-50 MobileNetV2	ResNet-50 VGG-8	VGG-13 MobileNetV2
Baseline	64.64	70.40	64.64
KD	67.62	73.39	67.36
AT	58.58	71.14	59.83
SP	68.00	72.89	65.89
STKD	**69.07**	**73.81**	**68.22**
Teacher	79.34	79.34	74.31

Table 7
Classification accuracy (%) on CIFAR-100 (5 runs). Baseline means the ResNet-20 trains individually. STKD means the ResNet-110 results in our method.

λ	0.43	0.46	0.50	0.53
$1-\lambda$	0.57	0.54	0.50	0.47
Baseline	69.14	69.14	69.14	69.14
STKD	70.21	71.08	**72.14**	69.82
Teacher	74.37	74.37	74.37	74.37

4.5. Ablation study

Evaluation on Different Network Architectures. To further evaluate the performance of STKD, we perform additional experiments on different network architecture families for the student/teacher pairs on CIFAR-100. Table 6 shows the experimental results.

As can be seen from Table 6, STKD continuously outperforms KD and other SOAT methods, which indicates the effectiveness of our method. In particular, STKD achieves the accuracy of 69.07%, 73.81% and 68.22% for three network pairs of different network families respectively, obtaining a performance gain of 1.07%, 0.92% and 2.33% over SP. However, we observe that some methods such as AT [18] perform quite poorly when the teacher and student belong to different network architectures. Thus, the results validate that our approach is robust to teacher/student pairs. And we conclude that the knowledge contained in the similarity correlation between different instances helps to improve the robustness of the student network.

Impact of Different coefficients for mixup. In this subsection, we explore the impact of coefficients for mixup. We adopt ResNet-110 as the teacher network, ResNet-20 as the student network. The results on CIFAR-100 could be found in Table 7, which illustrates how the performance of STKD is affected by the choice of the coefficient λ. By setting different λ for STKD, we observe that STKD achieves the accuracy of 72.14% ($\lambda = 0.50$), which represents the best performance in our settings.

Furthermore, we show the mixup images from the same sample pairs by varying the mixup coefficient λ in Fig. 3. The outputs

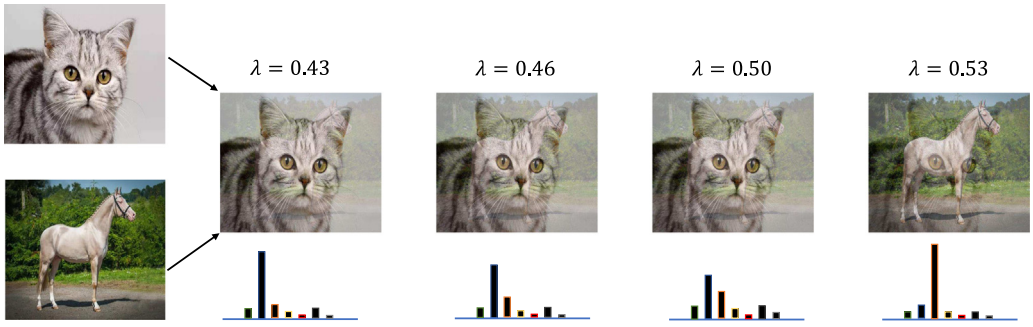

Fig. 3. Different mixup images from the same sample pairs by varying the mixup coefficient λ. We plot the probability distribution of each mixup image from the teacher network.

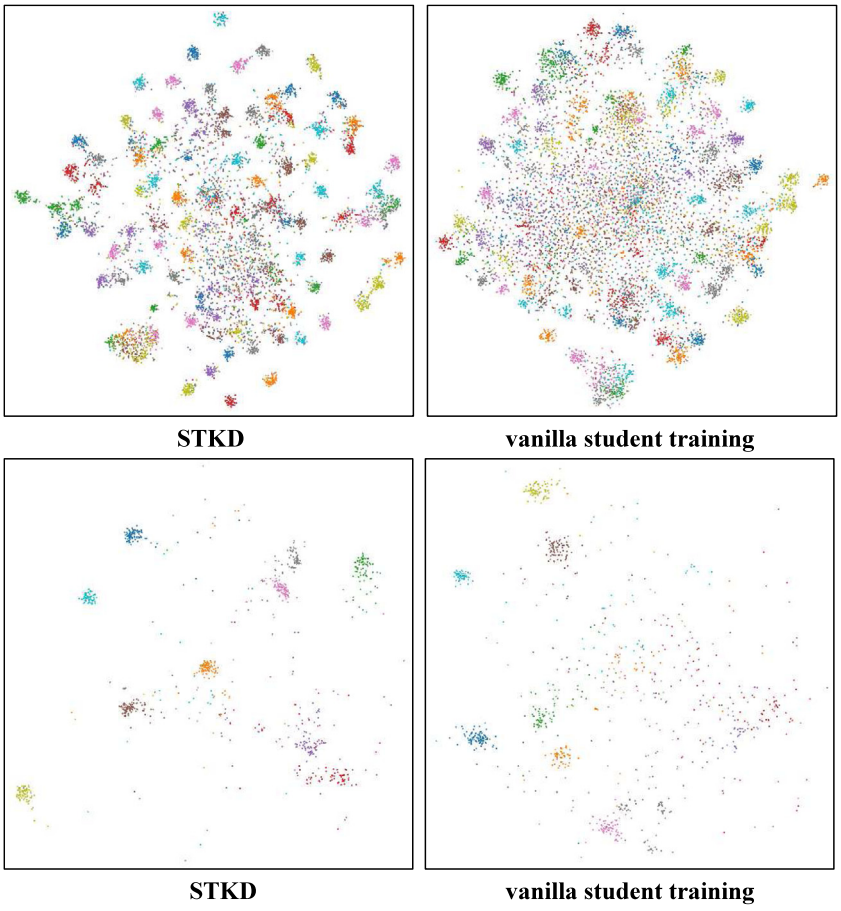

Fig. 4. The tSNE visualization over 100 classes of the STKD (left) and the vanilla student training (right). Each color represents a category, best viewed in color. The first row shows 100 classes together. The second row shows 10 sampled classes from CIFAR-100.

of probability distribution are also plotted under each mixup image. As can be seen from Fig. 3, when $\lambda = 0.50$, the mixup image obtains a smoother probability distribution than other settings. It means that the relative probability assigned to similar categories encodes semantic similarity between similar categories. Note that, previous conventional KD methods usually achieve this goal by setting a large temperature hyperparameter t.

Visualization the distribution of student networks. In this subsection, we visualize the output distribution of student from STKD and vanilla training methods on CIFAR-100. Fig. 4 shows the visualization results over 100 classes and 10 sampled classes using tSNE [73] respectively. Notably, the feature space of STKD is significantly more separable than that using the vanilla training manner. We visualize the high-dimensional features by the t-SNE algorithm for the proposed method and vanilla training method. Compared with the method based on traditional distillation loss, we can observe that the cluster with the same category becomes closer, while clusters with different categories become much separable. It means that the proposed method has substantially outperformed the vanilla knowledge distillation.

5. Conclusion

In this paper, we introduce a novel knowledge distillation method, called multi-instance semantic similarity transferring for knowledge distillation that aims to utilize the similarity correlation between different instances. Moreover, we adopt the mixup technique to encode semantic similarity between categories, which assigns the relative probability to different categories from instances. And the one-hot labels are replaced by the

mix labels in virtual samples, created by the mixup technique. Our experiments demonstrate that the dark knowledge from the teacher model could be better transferred to the student model and thus improve the training outcomes of the student. We also show that our approach can be regarded as the regularized term for the student in knowledge distillation. The experiment evaluation on three benchmarks CIFAR-10/100, CINIC-10 and Tiny-ImageNet shows that our approach has achieved the state-of-the-art accuracy for a variety of network architectures indicating the effectiveness of the proposed methods. One drawback of the proposed work is that we only consider the relationships between data samples. However, relationships between different layers also contain valuable information. For further work, we will explore the graph-based knowledge as the distilled knowledge.

CRediT authorship contribution statement

Haoran Zhao: Conceptualization, Data curation, Methodology, Writing – original draft. **Xin Sun:** Writing – review & editing, Supervision. **Junyu Dong:** Project administration, Investigation. **Hui Yu:** Supervision Resources, Data curation. **Gaige Wang:** Supervision, Data curation.

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.

Acknowledgments

We thank supports of the National Natural Science Foundation of China (No. 61971388, U1706218, 61976123, 61601427); Taishan Young Scholars Program of Shandong Province; and Alexander von Humboldt Foundation, Germany.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. (2015) 211–252.

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

[3] Z. Wang, J. Du, Joint architecture and knowledge distillation in CNN for Chinese text recognition, Pattern Recognit. 111 (2021) 107722.

[4] Q. Li, S. Jin, J. Yan, Mimicking very efficient network for object detection, in: The IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017.

[5] M. Feng, S.Z. Gilani, Y. Wang, L. Zhang, A. Mian, Relation graph network for 3D object detection in point clouds, IEEE Trans. Image Process. 30 (2021) 92–107, http://dx.doi.org/10.1109/TIP.2020.3031371.

[6] J. Xie, B. Shuai, J.-F. Hu, J. Lin, W.-S. Zheng, Improving fast segmentation with teacher-student learning, in: British Machine Vision Conference 2018 (BMVC), 2018, p. 205.

[7] J. Wu, R. Ji, J. Liu, M. Xu, J. Zheng, L. Shao, Q. Tian, Real-time semantic segmentation via sequential knowledge distillation, Neurocomputing 439 (2021) 134–145.

[8] T. Wu, S. Tang, R. Zhang, J. Cao, Y. Zhang, CGNet: A light-weight context guided network for semantic segmentation, IEEE Trans. Image Process. 30 (2021) 1169–1179, http://dx.doi.org/10.1109/TIP.2020.3042065.

[9] Y.L. Cun, J.S. Denker, S.A. Solla, Optimal brain damage, in: International Conference on Neural Information Processing Systems, 1989, pp. 598–605.

[10] H. Li, A. Kadav, I. Durdanovic, H. Samet, H.P. Graf, Pruning filters for efficient ConvNets, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, 2017.

[11] S. Lin, R. Ji, Y. Li, C. Deng, X. Li, Toward compact ConvNets via structure-sparsity regularized filter pruning, IEEE Trans. Neural Netw. Learn. Syst. 31 (2) (2020) 574–588.

[12] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, N.D. Freitas, Predicting parameters in deep learning, in: International Conference on Neural Information Processing Systems, 2013, pp. 2148–2156.

[13] Y.D. Kim, E. Park, S. Yoo, T. Choi, Y. Lu, D. Shin, Compression of deep convolutional neural networks for fast and low power mobile applications, Comput. Sci. 71 (2) (2015) 576–584.

[14] W. Ren, J. Zhang, L. Ma, J. Pan, X. Cao, W. Zuo, W. Liu, M. Yang, Deep non-blind deconvolution via generalized low-rank approximation, in: Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, 2018, pp. 295–305.

[15] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, Comput. Sci. 14 (7) (2015) 38–39.

[16] T.-B. Xu, P. Yang, X.-Y. Zhang, C.-L. Liu, LightweightNet: Toward fast and lightweight convolutional neural networks via architecture distillation, Pattern Recognit. 88 (2019) 272–284.

[17] A. Romero, N. Ballas, S.E. Kahou, A. Chassang, C. Gatta, Y. Bengio, FitNets: Hints for thin deep nets, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[18] S. Zagoruyko, N. Komodakis, Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, 2017.

[19] J. Yim, D. Joo, J. Bae, J. Kim, A gift from knowledge distillation: Fast optimization, network minimization and transfer learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7130–7138.

[20] W. Park, D. Kim, Y. Lu, M. Cho, Relational knowledge distillation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3967–3976.

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

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

[23] S. Srinivas, F. Fleuret, in: J. Dy, A. Krause (Eds.), Knowledge Transfer with Jacobian Matching, in: Proceedings of Machine Learning Research, vol. 80, PMLR, Stockholmsmässan, Stockholm Sweden, 2018, pp. 4723–4731.

[24] N. Passalis, A. Tefas, Learning deep representations with probabilistic knowledge transfer, in: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI, 2018, pp. 283–299.

[25] N. Papernot, P.D. McDaniel, X. Wu, S. Jha, A. Swami, Distillation as a defense to adversarial perturbations against deep neural networks, in: IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016, IEEE Computer Society, 2016, pp. 582–597.

[26] S. Gupta, J. Hoffman, J. Malik, Cross modal distillation for supervision transfer, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 2827–2836.

[27] J. Uijlings, S. Popov, V. Ferrari, Revisiting knowledge transfer for training object class detectors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1101–1110.

[28] X. Han, X. Song, Y. Yao, X.S. Xu, L. Nie, Neural compatibility modeling with probabilistic knowledge distillation, IEEE Trans. Image Process. 29 (2020) 871–882, http://dx.doi.org/10.1109/TIP.2019.2936742.

[29] W. Li, S. Gong, X. Zhu, Hierarchical distillation learning for scalable person search, Pattern Recognit. 114 (2021) 107862.

[30] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, J. Wang, Structured knowledge distillation for semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, 2019, pp. 2604–2613, http://dx.doi.org/10.1109/CVPR.2019.00271.

[31] J. Jiao, Y. Wei, Z. Jie, H. Shi, R.W.H. Lau, T.S. Huang, Geometry-aware distillation for indoor semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, 2019, pp. 2869–2878, http://dx.doi.org/10.1109/CVPR.2019.00298.

[32] P. Luo, Z. Zhu, Z. Liu, X. Wang, X. Tang, et al., Face model compression by distilling knowledge from neurons, in: AAAI, 2016, pp. 3560–3566.

[33] Y. Zhao, Y. Liu, C. Shen, Y. Gao, S. Xiong, MobileFAN: Transferring deep hidden representation for face alignment, Pattern Recognit. 100 (2020) 107114.

[34] C. Bian, W. Feng, L. Wan, S. Wang, Structural knowledge distillation for efficient skeleton-based action recognition, IEEE Trans. Image Process. 30 (2021) 2963–2976, http://dx.doi.org/10.1109/TIP.2021.3056895.

[35] W. Hao, Z. Zhang, Spatiotemporal distilled dense-connectivity network for video action recognition, Pattern Recognit. 92 (2019) 13–24.

[36] A. Pilzer, S. Lathuilière, N. Sebe, E. Ricci, Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, 2019, pp. 9768–9777, http://dx.doi.org/10.1109/CVPR.2019.01000.

[37] L. Yuan, F.E.H. Tay, G. Li, T. Wang, J. Feng, Revisiting knowledge distillation via label smoothing regularization, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, IEEE, 2020, pp. 3902–3910, http://dx.doi.org/10.1109/CVPR42600.2020.00396.

[38] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 2818–2826, http://dx.doi.org/10.1109/CVPR.2016.308.

[39] R. Müller, S. Kornblith, G.E. Hinton, When does label smoothing help? in: Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019, pp. 4694–4703.

[40] H. Zhang, M. Cissé, Y.N. Dauphin, D. Lopez-Paz, Mixup: Beyond empirical risk minimization, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net, 2018.

[41] Y. Ming, X. Meng, C. Fan, H. Yu, Deep learning for monocular depth estimation: A review, Neurocomputing 438 (2021) 14–33.

[42] A. Zhao, L. Qi, J. Li, J. Dong, H. Yu, A hybrid spatio-temporal model for detection and severity rating of Parkinson's disease from gait data, Neurocomputing 315 (2018) 1–8.

[43] J. Lou, Y. Wang, C. Nduka, M. Hamedi, I. Mavridou, F. Wang, H. Yu, Realistic facial expression reconstruction for VR HMD users, IEEE Trans. Multimed. 22 (3) (2020) 730–743.

[44] X. Li, J. Wu, Z. Sun, Z. Ma, J. Cao, J.H. Xue, BSNet: Bi-similarity network for few-shot fine-grained image classification, IEEE Trans. Image Process. 30 (2021) 1318–1331, http://dx.doi.org/10.1109/TIP.2020.3043128.

[45] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient convolutional neural networks for mobile vision applications, 2017, CoRR abs/1704.04861. URL http://arxiv.org/abs/1704.04861.

[46] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.

[47] S. Han, J. Pool, J. Tran, W.J. Dally, Learning both weights and connections for efficient neural network, in: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, 2015, pp. 1135–1143.

[48] A. Novikov, D. Podoprikhin, A. Osokin, D.P. Vetrov, Tensorizing neural networks, in: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, 2015, pp. 442–450.

[49] J. Yang, X. Shen, J. Xing, X. Tian, H. Li, B. Deng, J. Huang, X. Hua, Quantization networks, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, 2019, pp. 7308–7316.

[50] C. Bucila, R. Caruana, A. Niculescu-Mizil, Model compression, in: Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006, 2006, pp. 535–541.

[51] J. Ba, R. Caruana, Do deep nets really need to be deep? in: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, 2014, pp. 2654–2662.

[52] H. Zhang, Z. Hu, W. Qin, M. Xu, M. Wang, Adversarial co-distillation learning for image recognition, Pattern Recognit. 111 (2021) 107659.

[53] Z. Huang, N. Wang, Like what you like: Knowledge distill via neuron selectivity transfer, 2017, CoRR abs/1707.01219. URL http://arxiv.org/abs/1707.01219.

[54] B. Heo, M. Lee, S. Yun, J.Y. Choi, Knowledge transfer via distillation of activation boundaries formed by hidden neurons, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI Honolulu, Hawaii, USA, January 27 - February 1, 2019, pp. 3779–3787.

[55] S.H. Lee, D.H. Kim, B.C. Song, Self-supervised knowledge distillation using singular value decomposition, in: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VI, 2018, pp. 339–354.

[56] B. Peng, X. Jin, D. Li, S. Zhou, Y. Wu, J. Liu, Z. Zhang, Y. Liu, Correlation congruence for knowledge distillation, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, 2019, pp. 5006–5015.

[57] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, Y. Bengio, Manifold mixup: Better representations by interpolating hidden states, in: Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, 2019, pp. 6438–6447.

[58] C. Summers, M.J. Dinneen, Improved mixed-example data augmentation, in: IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019, 2019, pp. 1262–1270.

[59] R. Takahashi, T. Matsubara, K. Uehara, RICAP: Random image cropping and patching data augmentation for deep CNNs, in: Proceedings of the 10th Asian Conference on Machine Learning, ACML 2018, Beijing, China, November 14-16, 2018, 2018, pp. 786–798.

[60] S. Yun, D. Han, S. Chun, S.J. Oh, Y. Yoo, J. Choe, CutMix: Regularization strategy to train strong classifiers with localizable features, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, 2019, pp. 6022–6031.

[61] E. Harris, A. Marcu, M. Painter, M. Niranjan, A. Prügel-Bennett, J.S. Hare, Understanding and enhancing mixed sample data augmentation, 2020, CoRR abs/2002.12047. URL https://arxiv.org/abs/2002.12047.

[62] H. Inoue, Data augmentation by pairing samples for images classification, 2018, CoRR abs/1801.02929. URL http://arxiv.org/abs/1801.02929.

[63] E.D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, Q.V. Le, AutoAugment: Learning augmentation policies from data, 2018, CoRR abs/1805.09501. URL http://arxiv.org/abs/1805.09501.

[64] 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.

[65] S. Zagoruyko, N. Komodakis, Wide residual networks, in: Proceedings of the British Machine Vision Conference, 2016.

[66] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd International Conference on Learning Representations, ICLR, 2015.

[67] A. Krizhevsky, G. Hinton, Learning Multiple Layers of Features from Tiny Images, Tech. rep., 2009.

[68] L.N. Darlow, E.J. Crowley, A. Antoniou, A.J. Storkey, CINIC-10 is not ImageNet or CIFAR-10, 2018, CoRR abs/1810.03505. URL http://arxiv.org/abs/1810.03505.

[69] Y. Le, X. Yang, Tiny imagenet visual recognition challenge, in: Stanford Class CS 231N, 2015.

[70] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, 2016, pp. 770–778.

[71] S. Zagoruyko, N. Komodakis, Wide residual networks, in: Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016, 2016.

[72] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[73] L. van der Maaten, K.Q. Weinberger, Stochastic triplet embedding, in: IEEE International Workshop on Machine Learning for Signal Processing, MLSP 2012, Santander, Spain, September 23-26, 2012, 2012, pp. 1–6.