Similarity-Preserving Knowledge Distillation

Frederick Tung[1,2] and Greg Mori[1,2]
[1]Simon Fraser University [2]Borealis AI
ftung@sfu.ca, mori@cs.sfu.ca

Abstract

Knowledge distillation is a widely applicable technique for training a student neural network under the guidance of a trained teacher network. For example, in neural network compression, a high-capacity teacher is distilled to train a compact student; in privileged learning, a teacher trained with privileged data is distilled to train a student without access to that data. The distillation loss determines how a teacher's knowledge is captured and transferred to the student. In this paper, we propose a new form of knowledge distillation loss that is inspired by the observation that semantically similar inputs tend to elicit similar activation patterns in a trained network. Similarity-preserving knowledge distillation guides the training of a student network such that input pairs that produce similar (dissimilar) activations in the teacher network produce similar (dissimilar) activations in the student network. In contrast to previous distillation methods, the student is not required to mimic the representation space of the teacher, but rather to preserve the pairwise similarities in its own representation space. Experiments on three public datasets demonstrate the potential of our approach.

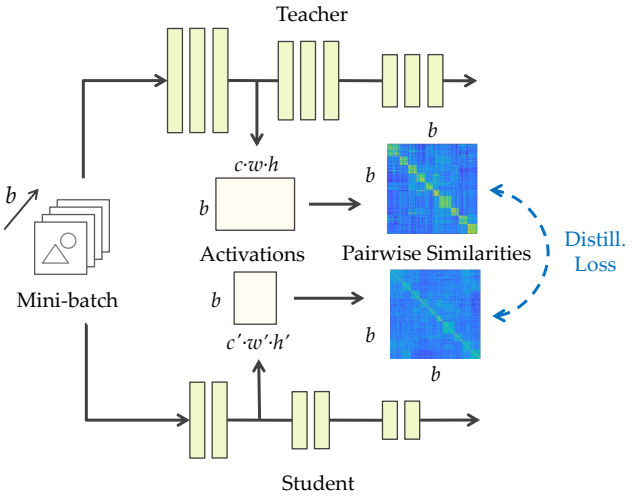

Figure 1. *Similarity-preserving knowledge distillation* guides the training of a student network such that input pairs that produce similar (dissimilar) activations in the pre-trained teacher network produce similar (dissimilar) activations in the student network. Given an input mini-batch of b images, we derive $b \times b$ pairwise similarity matrices from the activation maps, and compute a distillation loss on the matrices produced by the student and the teacher.

1. Introduction

Deep neural networks are being used to solve an increasingly wide array of computer vision problems. While the general trend in deep learning is towards deeper, wider, and more complex networks, deploying deep learning solutions in the real world requires us to consider the computational cost. A mobile robot or self-driving vehicle, for example, has limited memory and power. Even when resources are abundant, such as when a vision system is hosted in the cloud, more resource-efficient deep networks mean more clients can be served at a lower cost. When performing transfer learning in the real world, data privilege and privacy issues may restrict access to data in the source domain. It may be necessary to transfer the knowledge of a network trained on the source domain assuming access only to training data from the target task domain.

Knowledge distillation is a general technique for supervising the training of "student" neural networks by capturing and transferring the knowledge of trained "teacher" networks. While originally motivated by the task of neural network compression for resource-efficient deep learning [12], knowledge distillation has found wider applications in such areas as privileged learning [21], adversarial defense [25], and learning with noisy data [19]. Knowledge distillation is conceptually simple: it guides the training of a student network with an additional distillation loss that encourages the student to mimic some aspect of a teacher network. Intuitively, the trained teacher network provides a richer supervisory signal than the data supervision (e.g. annotated class labels) alone.

The conceptual simplicity of knowledge distillation belies the fact that *how* to best capture the knowledge of the teacher to train the student (i.e. how to define the distillation

loss) remains an open question. In traditional knowledge distillation [12], the softened class scores of the teacher are used as the extra supervisory signal: the distillation loss encourages the student to mimic the scores of the teacher. FitNets [31] extend this idea by adding hints to guide the training of intermediate layers. In flow-based knowledge distillation [38], the extra supervisory signal comes from the inter-layer "flow" – how features are transformed between layers. The distillation loss encourages the student to mimic the teacher's flow matrices, which are derived from the inner product between feature maps in two layers, such as the first and last layers in a residual block. In attention transfer [41], the supervisory signal for knowledge distillation is in the form of spatial attention. Spatial attention maps are computed by summing the squared activations along the channel dimension. The distillation loss encourages the student to produce similar normalized spatial attention maps as the teacher, intuitively paying attention to similar parts of the image as the teacher.

In this paper, we present a novel form of knowledge distillation that is inspired by the observation that semantically similar inputs tend to elicit similar activation patterns in a trained neural network. Similarity-preserving knowledge distillation guides the training of a student network such that input pairs that produce similar (dissimilar) activations in the trained teacher network produce similar (dissimilar) activations in the student network. Figure 1 shows the overall procedure. Given an input mini-batch of b images, we compute pairwise similarity matrices from the output activation maps. The $b \times b$ matrices encode the similarities in the activations of the network as elicited by the images in the mini-batch. Our distillation loss is defined on the pairwise similarity matrices produced by the student and the teacher.

To support the intuition of our distillation loss, Figure 2 visualizes the average activation of each channel in the last convolutional layer of a WideResNet-16-2 teacher network (we adopt the standard notation WideResNet-d-k to refer to a wide residual network [40] with depth d and width multiplier k), on the CIFAR-10 test images. We can see that images from the same object category tend to activate similar channels in the trained network. The similarities in activations across different images capture useful semantics learned by the teacher network. We study whether these similarities provide an informative supervisory signal for knowledge distillation.

The contributions of this paper are:

- We introduce *similarity-preserving knowledge distillation*, a novel form of knowledge distillation that uses the pairwise activation similarities within each input mini-batch to supervise the training of a student network with a trained teacher network.

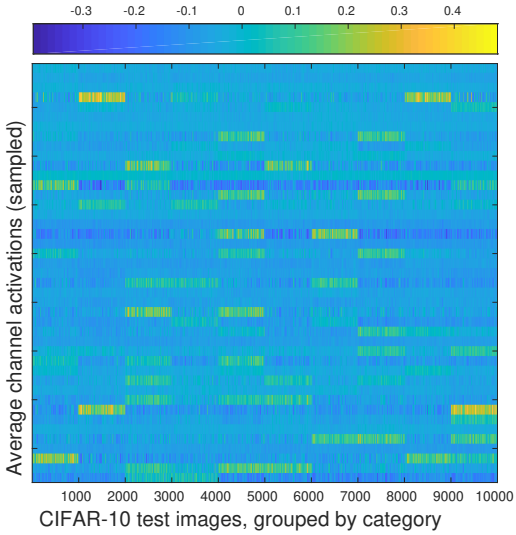

Figure 2. Semantically similar inputs tend to elicit similar activation patterns in a trained neural network. This visualization shows channel-wise average activations sampled from the last convolutional layer of a WideResNet-16-2 network on the CIFAR-10 test images. Activation patterns are largely consistent within the same category (e.g. columns 1 to 1000) and distinctive across different categories (e.g. columns 1 to 1000 vs. columns 1001 to 2000).

- We experimentally validate our approach on three public datasets. Our experiments show the potential of similarity-preserving knowledge distillation, not only for improving the training outcomes of student networks, but also for complementing traditional methods for knowledge distillation.

2. Method

The goal of knowledge distillation is to train a student network under the guidance of a trained teacher network, which acts as an extra source of supervision. For example, in neural network compression, the student network is computationally cheaper than the teacher: it may be shallower, thinner, or composed of cheaper operations. The trained teacher network provides additional semantic knowledge beyond the usual data supervision (e.g. the usual one-hot vectors for classification). The challenge is to determine how to encode and transfer the teacher's knowledge such that student performance is maximized.

In traditional knowledge distillation [12], knowledge is encoded and transferred in the form of softened class scores. The total loss for training the student is given by

$$\mathcal{L} = (1-\alpha)\mathcal{L}_{\text{CE}}(\mathbf{y}, \sigma(\mathbf{z}_S)) + 2\alpha T^2 \mathcal{L}_{\text{CE}}(\sigma(\frac{\mathbf{z}_S}{T}), \sigma(\frac{\mathbf{z}_T}{T})) \,, \tag{1}$$

where $\mathcal{L}_{\text{CE}}(\cdot, \cdot)$ denotes the cross-entropy loss, $\sigma(\cdot)$ denotes the softmax function, $\mathbf{y}$ is the one-hot vector indicating the ground truth class, $\mathbf{z}_S$ and $\mathbf{z}_T$ are the output logits of the

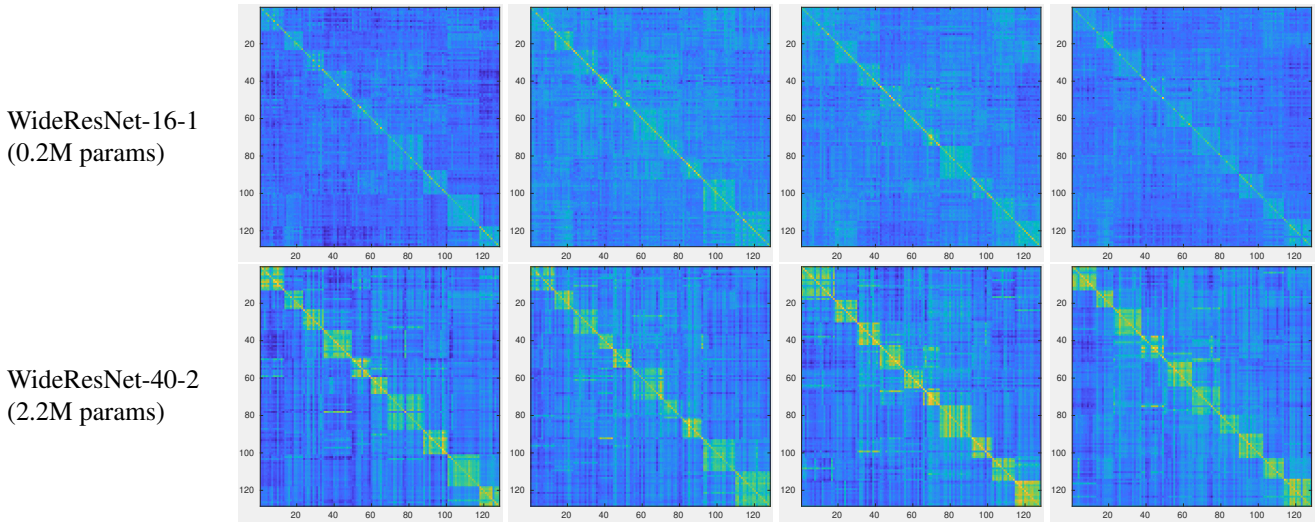

Figure 3. Activation similarity matrices G (Eq. 2) produced by trained WideResNet-16-1 and WideResNet-40-2 networks on sample CIFAR-10 test batches. Each column shows a single batch with inputs grouped by ground truth class along each axis (batch size = 128). Brighter colors indicate higher similarity values. The blockwise patterns indicate that the elicited activations are mostly similar for inputs of the same class, and different for inputs across different classes. Our distillation loss (Eq. 4) encourages the student network to produce G matrices closer to those produced by the teacher network.

student and teacher networks, respectively, T is a temperature hyperparameter, and α is a balancing hyperparameter. The first term in Eq.1 is the usual cross-entropy loss defined using data supervision (ground truth labels), while the second term encourages the student to mimic the softened class scores of the teacher.

Recall from the introduction and Figure 2 that semantically similar inputs tend to elicit similar activation patterns in a trained neural network. In Figure 2, we can observe that activation patterns are largely consistent within the same object category and distinctive across different categories. Might the correlations in activations encode useful teacher knowledge that can be transferred to the student? Our hypothesis is that, if two inputs produce highly similar activations in the teacher network, it is beneficial to guide the student network towards a configuration that also results in the two inputs producing highly similar activations in the student. Conversely, if two inputs produce dissimilar activations in the teacher, we want these inputs to produce dissimilar activations in the student as well.

Given an input mini-batch, denote the activation map produced by the teacher network T at a particular layer l by $A_T^{(l)} \in \mathbf{R}^{b\times c\times h\times w}$, where b is the batch size, c is the number of output channels, and h and w are spatial dimensions. Let the activation map produced by the student network S at a corresponding layer l' be given by $A_S^{(l')} \in \mathbf{R}^{b\times c'\times h'\times w'}$. Note that c does not necessarily have to equal c', and likewise for the spatial dimensions. Similar to attention transfer [41], the corresponding layer l' can be the layer at the same depth as l if the student and teacher share the same depth, or the layer at the end of the same block if the student and teacher have different depths. To guide the student towards the activation correlations induced in the teacher, we define a distillation loss that penalizes differences in the L2-normalized outer products of $A_T^{(l)}$ and $A_S^{(l')}$. First, let

$$\tilde{G}_T^{(l)} = Q_T^{(l)} \cdot Q_T^{(l)\top}; \;\; G_{T[i,:]}^{(l)} = \tilde{G}_{T[i,:]}^{(l)} \,/\, ||\tilde{G}_{T[i,:]}^{(l)}||_2 \quad (2)$$

where $Q_T^{(l)} \in \mathbf{R}^{b\times chw}$ is a reshaping of $A_T^{(l)}$, and therefore $\tilde{G}_T^{(l)}$ is a $b\times b$ matrix. Intuitively, entry (i,j) in $\tilde{G}_T^{(l)}$ encodes the similarity of the activations at this teacher layer elicited by the ith and jth images in the mini-batch. We apply a row-wise L2 normalization to obtain $G_T^{(l)}$, where the notation $[i,:]$ denotes the ith row in a matrix. Analogously, let

$$\tilde{G}_S^{(l)} = Q_S^{(l)} \cdot Q_S^{(l)\top}; \;\; G_{S[i,:]}^{(l)} = \tilde{G}_{S[i,:]}^{(l)} \,/\, ||\tilde{G}_{S[i,:]}^{(l)}||_2 \quad (3)$$

where $Q_S^{(l')} \in \mathbf{R}^{b\times c'h'w'}$ is a reshaping of $A_S^{(l')}$, and $G_S^{(l')}$ is a $b\times b$ matrix. We define the similarity-preserving knowledge distillation loss as:

$$\mathcal{L}_{\text{SP}}(G_T, G_S) = \frac{1}{b^2} \sum_{(l,l')\in\mathcal{I}} ||G_T^{(l)} - G_S^{(l')}||_F^2 \,, \quad (4)$$

where $\mathcal{I}$ collects the (l,l') layer pairs (e.g. layers at the end of the same block, as discussed above) and $||\cdot||_F$ is the Frobenius norm. Eq. 4 is a summation, over all (l,l') pairs, of the mean element-wise squared difference between the $G_T^{(l)}$ and $G_S^{(l')}$ matrices. Finally, we define the total loss for training the student network as:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}(\mathbf{y}, \sigma(\mathbf{z}_S)) + \gamma\, \mathcal{L}_{\text{SP}}(G_T, G_S) \,, \quad (5)$$

Group name	Output size	WideResNet-16-k	WideResNet-40-k
conv1	32×32	$3 \times 3, 16$	$3 \times 3, 16$
conv2	32×32	$\begin{bmatrix} 3 \times 3, 16k \\ 3 \times 3, 16k \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 16k \\ 3 \times 3, 16k \end{bmatrix} \times 6$
conv3	16×16	$\begin{bmatrix} 3 \times 3, 32k \\ 3 \times 3, 32k \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 32k \\ 3 \times 3, 32k \end{bmatrix} \times 6$
conv4	8×8	$\begin{bmatrix} 3 \times 3, 64k \\ 3 \times 3, 64k \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64k \\ 3 \times 3, 64k \end{bmatrix} \times 6$
	1×1	average pool, 10-d fc, softmax	

Table 1. Structure of WideResNet networks used in CIFAR-10 experiments. Downsampling is performed by strided convolutions in the first layers of conv3 and conv4.

where γ is a balancing hyperparameter.

Figure 3 visualizes the G matrices for several batches in the CIFAR-10 test set. The top row is produced by a trained WideResNet-16-1 network, consisting of 0.2M parameters, while the bottom row is produced by a trained WideResNet-40-2 network, consisting of 2.2M parameters. In both cases, activations are collected from the last convolution layer. Each column represents a single batch, which is identical for both networks. The images in each batch have been grouped by their ground truth class for easier interpretability. The G matrices in both rows show a distinctive blockwise pattern, indicating that the activations at the last layer of these networks are largely similar within the same class and dissimilar across different classes (the blocks are differently sized because each batch has an unequal number of test samples from each class). Moreover, the blockwise pattern is more distinctive for the WideResNet-40-2 network, reflecting the higher capacity of this network to capture the semantics of the dataset. Intuitively, Eq. 4 pushes the student network towards producing G matrices closer to those produced by the teacher network.

Differences from previous approaches. The similarity-preserving knowledge distillation loss (Eq. 4) is defined in terms of activations instead of class scores as in traditional distillation [12]. Activations are also used to define the distillation losses in FitNets [31], flow-based distillation [38], and attention transfer [41]. However, a key difference is that these previous distillation methods encourage the student to mimic different aspects of the representation space of the teacher. Our method is a departure from this common approach in that it aims to preserve the pairwise activation similarities of input samples. Its behavior is unchanged by a rotation of the teacher's representation space, for example. In similarity-preserving knowledge distillation, the student is not required to be able to express the representation space of the teacher, as long as pairwise similarities in the teacher space are well preserved in the student space.

3. Experiments

We now turn to the experimental validation of our distillation approach on three public datasets. We start with CIFAR-10 as it is a commonly adopted dataset for comparing distillation methods, and its relatively small size allows multiple student and teacher combinations to be evaluated. We then consider the task of transfer learning, and show how distillation and fine-tuning can be combined to perform transfer learning on a texture dataset with limited training data. Finally, we report results on the larger CINIC-10 dataset.

3.1. CIFAR-10

CIFAR-10 consists of 50,000 training images and 10,000 testing images at a resolution of 32x32. The dataset covers ten object classes, with each class having an equal number of images. We conducted experiments using wide residual networks (WideResNets) [40] following [4, 41]. Table 1 summarizes the structure of the networks. We adopted the standard protocol [40] for training wide residual networks on CIFAR-10 (SGD with Nesterov momentum; 200 epochs; batch size of 128; and an initial learning rate of 0.1, decayed by a factor of 0.2 at epochs 60, 120, and 160). We applied the standard horizontal flip and random crop data augmentation. We performed baseline comparisons with respect to traditional knowledge distillation (softened class scores) and attention transfer. For traditional knowledge distillation [12], we set $\alpha = 0.9$ and $T = 4$ following the CIFAR-10 experiments in [4, 41]. Attention transfer losses were applied for each of the three residual block groups. We set the weight of the distillation loss in attention transfer and similarity-preserving distillation by held-out validation on a subset of the training set ($\beta = 1000$ for attention transfer, $\gamma = 3000$ for similarity-preserving distillation).

Table 2 shows our results experimenting with several student-teacher network pairs. We tested cases in which the student and teacher networks have the same width but different depth (WideResNet-16-1 student with WideResNet-40-1 teacher; WideResNet-16-2 student with WideResNet-40-2 teacher), the student and teacher networks have the same depth but different width (WideResNet-16-1 student with WideResNet-16-2 teacher; WideResNet-16-2 student with WideResNet-16-8 teacher), and the student and teacher have different depth and width (WideResNet-40-2 student with WideResNet-16-8 teacher). In all cases, transferring the knowledge of the teacher network using similarity-preserving distillation improved student training outcomes. Compared to conventional training with data supervision (i.e. one-hot vectors), the student network consistently obtained lower median error, from 0.5 to 1.2 absolute percentage points, or 7% to 14% relative, with no additional network parameters or operations. Similarity-preserving distillation also performed favorably with respect to the tra-

Student	Teacher	Student	KD [12]	AT [41]	SP (ours)	Teacher
WideResNet-16-1 (0.2M)	WideResNet-40-1 (0.6M)	8.74	8.48	8.30	**8.13**	6.51
WideResNet-16-1 (0.2M)	WideResNet-16-2 (0.7M)	8.74	7.94	8.28	**7.52**	6.07
WideResNet-16-2 (0.7M)	WideResNet-40-2 (2.2M)	6.07	6.00	5.89	**5.52**	5.18
WideResNet-16-2 (0.7M)	WideResNet-16-8 (11.0M)	6.07	5.62	5.47	**5.34**	4.24
WideResNet-40-2 (2.2M)	WideResNet-16-8 (11.0M)	5.18	4.86	**4.47**	4.55	4.24

Table 2. Experiments on CIFAR-10 with three different knowledge distillation losses: softened class scores (traditional KD), attention transfer (AT), and similarity preserving (SP). The median error over five runs is reported, following the protocol in [40, 41]. The best result for each experiment is shown in bold. Brackets indicate model size in number of parameters.

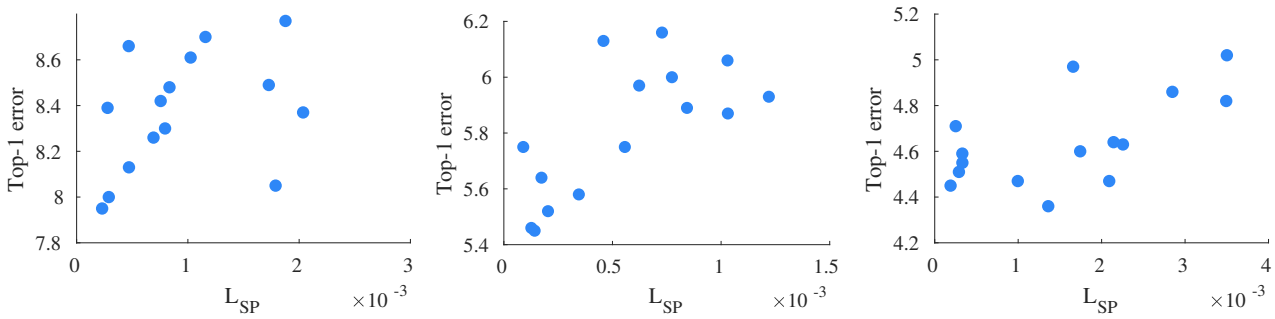

Figure 4. $\mathcal{L}_{SP}$ vs. error for (from left to right) WideResNet-16-1 students trained with WideResNet-16-2 teachers, WideResNet-16-2 students trained with WideResNet-40-2 teachers, and WideResNet-40-2 students trained with WideResNet-16-8 teachers, on CIFAR-10.

ditional (softened class scores) and attention transfer baselines, achieving the lowest error in four of the five cases. This validates our intuition that the activation similarities across images encode useful semantics learned by the teacher network, and provide an effective supervisory signal for knowledge distillation.

Figure 4 plots $\mathcal{L}_{SP}$ vs. error for the WideResNet-16-1/WideResNet-16-2, WideResNet-16-2/WideResNet-40-2, and WideResNet-40-2/WideResNet-16-8 experiments (left to right, respectively), using all students trained with traditional KD, AT, and SP. The plots verify that $\mathcal{L}_{SP}$ and performance are correlated.

While we have presented these results from the perspective of improving the training of a student network, it is also possible to view the results from the perspective of the teacher network. Our results suggest the potential for using similarity-preserving distillation to compress large networks into more resource-efficient ones with minimal accuracy loss. In the fifth test, for example, the knowledge of a trained WideResNet-16-8 network, which contains 11.0M parameters, is distilled into a much smaller WideResNet-40-2 network, which contains only 2.2M parameters. This is a $5\times$ compression rate with only 0.3% loss in accuracy, using off-the-shelf PyTorch without any specialized hardware or software.

The above similarity-preserving distillation results were produced using only the activations collected from the last convolution layers of the student and teacher networks. We also experimented with using the activations at the end of each WideResNet block, but found no improvement in performance. We therefore used only the activations at the final convolution layers in the subsequent experiments. Activation similarities may be less informative in the earlier layers of the network because these layers encode more generic features, which tend to be present across many images. Progressing deeper in the network, the channels encode increasingly specialized features, and the activation patterns of semantically similar images become more distinctive. We also experimented with using post-softmax scores to determine similarity, but this produces worse results than using activations. We found the same when using an oracle, suggesting that the soft teacher signal is important.

3.2. Transfer learning combining distillation with fine-tuning

In this section, we explore a common transfer learning scenario in computer vision. Suppose we are faced with a novel recognition task in a specialized image domain with limited training data. A natural strategy to adopt is to transfer the knowledge of a network pre-trained on ImageNet (or another suitable large-scale dataset) to the new recognition task by fine-tuning. Here, we combine knowledge distillation with fine-tuning: we initialize the student network with source domain (in this case, ImageNet) pretrained weights, and then fine-tune the student to the target domain using both distillation and cross-entropy losses (Eq. 5).

We analyzed this scenario using the describable textures dataset [3], which is composed of 5,640 images covering

Student	Teacher	Student	AT [41]	SP (win:loss)	Teacher
MobileNet-0.25 (0.2M)	MobileNet-0.5 (0.8M)	42.45	42.39	**41.30** (7:3)	36.76
MobileNet-0.25 (0.2M)	MobileNet-1.0 (3.3M)	42.45	41.89	**41.76** (5:5)	34.10
MobileNet-0.5 (0.8M)	MobileNet-1.0 (3.3M)	36.76	35.61	**35.45** (7:3)	34.10
MobileNetV2-0.35 (0.5M)	MobileNetV2-1.0 (2.2M)	41.25	41.60	**40.29** (8:2)	36.62
MobileNetV2-0.35 (0.5M)	MobileNetV2-1.4 (4.4M)	41.25	41.04	**40.43** (8:2)	35.35
MobileNetV2-1.0 (2.2M)	MobileNetV2-1.4 (4.4M)	36.62	36.33	**35.61** (8:2)	35.35

Table 3. Transfer learning experiments on the describable textures dataset with attention transfer (AT) and similarity preserving (SP) knowledge distillation. The median error over the ten standard splits is reported. The best result for each experiment is shown in bold. The (win:loss) notation indicates the number of splits in which SP outperformed AT. The (*M) notation indicates model size in number of parameters.

47 texture categories. Image sizes range from 300x300 to 640x640. We applied ImageNet-style data augmentation with horizontal flipping and random resized cropping during training. At test time, images were resized to 256x256 and center cropped to 224x224 for input to the networks. For evaluation, we adopted the standard ten training-validation-testing splits. To demonstrate the versatility of our method on different network architectures, and in particular its compatibility with mobile-friendly architectures, we experimented with variants of MobileNet [13] and MobileNetV2 [32]. Tables 1 and 2 in the supplementary summarize the structure of the networks.

We compared with an attention transfer baseline. Softened class score based distillation is not directly comparable in this setting because the classes in the source and target domains are disjoint. Similarity-preserving distillation can be applied directly to train the student, without first fine-tuning the teacher, since it aims to preserve similarities instead of mimicking the teacher's representation space. The teacher is run in inference mode to generate representations in the new domain. This capacity is useful when the new domain has limited training data, when the source domain is not accessible to the student (e.g. in privileged learning), or in continual learning where trained knowledge needs to be preserved across tasks [1]. We set the hyperparameters for attention transfer and similarity-preserving distillation by held-out validation on the ten standard splits. All networks were trained using SGD with Nesterov momentum, a batch size of 96, and for 60 epochs with an initial learning rate of 0.01 reduced to 0.001 after 30 epochs.

Table 3 shows that similarity-preserving distillation can effectively transfer knowledge across different domains. For all MobileNet and MobileNetV2 student-teacher pairs tested, applying similarity-preserving distillation during fine-tuning resulted in lower median student error than fine-tuning without distillation. Fine-tuning MobileNet-0.25 with distillation reduced the error by 1.1% absolute, and fine-tuning MobileNet-0.5 with distillation reduced the error by 1.3% absolute, compared to fine-tuning without distillation. Fine-tuning MobileNetV2-0.35 with distillation reduced the error by 1.0% absolute, and fine-tuning MobileNetV2-1.0 with distillation reduced the error by 1.0% absolute, compared to fine-tuning without distillation.

For all student-teacher pairs, similarity-preserving distillation obtained lower median error than the spatial attention transfer baseline. Table 3 incudes a breakdown of how similarity-preserving distillation compares with spatial attention transfer on a per-split basis. On aggregate, similarity-preserving distillation outperformed spatial attention transfer on 19 out of the 30 MobileNet splits and 24 out of the 30 MobileNetV2 splits. The results suggest that there may be a challenging domain shift in the important image areas for the network to attend. Moreover, while attention transfer summarizes the activation map by summing out the channel dimension, similarity-preserving distillation makes use of the full activation map in computing the similarity-based distillation loss, which may be more robust in the presence of a domain shift in attention.

3.3. CINIC-10

The CINIC-10 dataset [5] is designed to be a middle option relative to CIFAR-10 and ImageNet: it is composed of 32x32 images in the style of CIFAR-10, but at a total of 270,000 images its scale is closer to that of ImageNet. We adopted CINIC-10 for rapid experimentation because several GPU-months would have been required to perform full held-out validation and training on ImageNet for our method and all baselines.

For the student and teacher architectures, we experimented with variants of the state-of-the-art mobile architecture ShuffleNetV2 [23]. The ShuffleNetV2 networks are summarized in Table 3 in the supplementary. We used the standard training-validation-testing split and set the hyperparameters for similarity-preserving distillation and all baselines by held-out validation (KD: $\{\alpha = 0.6, T = 16\}$;

[1] In continual (or lifelong) learning, the goal is to extend a trained network to new tasks while avoiding the catastrophic forgetting of previous tasks. One way to prevent catastrophic forgetting is to supervise the new model (the student) with the model trained for previous tasks (the teacher) via a knowledge distillation loss [42].

Student	Teacher	Student	KD [12]	AT [41]	SP (ours)	KD+SP	AT+SP	Teacher
Sh.NetV2-0.5 (0.4M)	Sh.NetV2-1.0 (1.3M)	20.09	18.62	18.50	18.56	18.35	**18.20**	17.26
Sh.NetV2-0.5 (0.4M)	Sh.NetV2-2.0 (5.3M)	20.09	18.96	18.78	19.09	18.88	**18.43**	15.63
Sh.NetV2-1.0 (1.3M)	Sh.NetV2-2.0 (5.3M)	17.26	16.01	15.95	15.95	16.11	**15.89**	15.63
M.NetV2-0.35 (0.4M)	M.NetV2-1.0 (2.2M)	17.12	16.27	16.10	16.57	16.17	**15.66**	14.05

Table 4. Experiments on CINIC-10 with three different knowledge distillation losses: softened class scores (traditional KD), attention transfer (AT), and similarity preserving (SP). The best result for each experiment is shown in bold. Brackets indicate model size in number of parameters.

AT: $\beta = 50$; SP: $\gamma = 2000$; KD+SP: $\{\alpha = 0.6, T = 16, \gamma = 2000\}$; AT+SP: $\{\beta = 30, \gamma = 2000\}$). All networks were trained using SGD with Nesterov momentum, a batch size of 96, for 140 epochs with an initial learning rate of 0.01 decayed by a factor of 10 after the 100th and 120th epochs. We applied CIFAR-style data augmentation with horizontal flips and random crops during training.

The results are shown in Table 4 (top). Compared to conventional training with data supervision only, similarity-preserving distillation consistently improved student training outcomes. In particular, training ShuffleNetV2-0.5 with similarity-preserving distillation reduced the error by 1.5% absolute, and training ShuffleNetV2-1.0 with similarity-preserving distillation reduced the error by 1.3% absolute. On an individual basis, all three knowledge distillation approaches achieved comparable results, with a total spread of 0.12% absolute error on ShuffleNetV2-0.5 (for the best results with ShuffleNetV2-1.0 as teacher) and a total spread of 0.06% absolute error on ShuffleNetV2-1.0. However, the lowest error was achieved by combining similarity-preserving distillation with spatial attention transfer. Training ShuffleNetV2-0.5 combining both distillation losses reduced the error by 1.9% absolute, and training ShuffleNetV2-1.0 combining both distillation losses reduced the error by 1.4% absolute. This result shows that similarity-preserving distillation complements attention transfer and captures teacher knowledge that is not fully encoded in spatial attention maps. Table 4 (bottom) summarizes additional experiments with MobileNetV2. The results are similar: SP does not outperform the individual baselines but complements traditional KD and AT.

Sensitivity analysis. Figure 5 illustrates how the performance of similarity-preserving distillation is affected by the choice of hyperparameter γ. We plot the top-1 errors on the CINIC-10 test set for ShuffleNetV2-0.5 and ShuffleNetV2-1.0 students trained with γ ranging from 10 to 64,000. We observed robust performance over a broad range of values for γ. In all experiments, we set γ by held-out validation.

3.4. Different student and teacher architectures

We performed additional experiments with students and teachers from different architecture families on CIFAR-10. Table 5 shows that, for both MobileNetV2 and ShuffleNetV2, SP outperforms conventional training as well as the traditional KD and AT baselines.

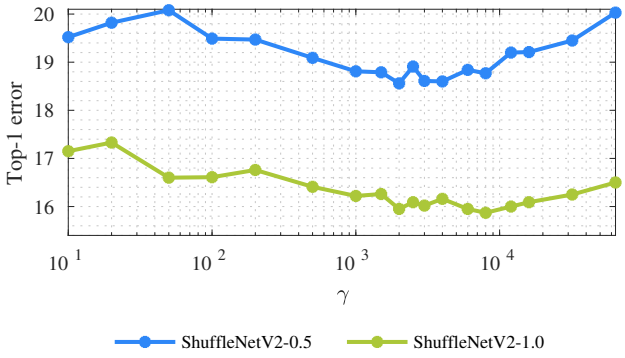

Figure 5. Sensitivity to γ on the CINIC-10 test set for ShuffleNetV2 students.

4. Related Work

We presented in this paper a novel distillation loss for capturing and transferring knowledge from a teacher network to a student network. Several prior alternatives [12, 31, 38, 41] are described in the introduction and some key differences are highlighted in Section 2. In addition to the knowledge capture (or loss definition) aspect of distillation studied in this paper, another important open question is the architectural design of students and teachers. In most studies of knowledge distillation, including ours, the student network is a thinner and/or shallower version of the teacher network. Inspired by efficient architectures such as MobileNet and ShuffleNet, Crowley et al. [4] proposed to replace regular convolutions in the teacher network with cheaper grouped and pointwise convolutions in the student. Ashok et al. [1] developed a reinforcement learning approach to learn the student architecture. Polino et al. [28] demonstrated how a quantized student network can be trained using a full-precision teacher network.

There is also innovative orthogonal work exploring alternatives to the usual student-teacher training paradigm. Wang et al. [34] introduced an additional discriminator network, and trained the student, teacher, and discrimina-

Student	Teacher	Student	KD [12]	AT [41]	SP (ours)	Teacher
ShuffleNetV2-0.5 (0.4M)	WideResNet-40-2 (2.2M)	8.95	9.09	8.55	**8.19**	5.18
MobileNetV2-0.35 (0.4M)	WideResNet-40-2 (2.2M)	7.44	7.08	7.43	**6.62**	5.18

Table 5. Additional experiments with students and teachers from different architecture families on CIFAR-10. The median error over five runs is reported, following the protocol in [40, 41]. The best result for each experiment is shown in bold. Brackets indicate model size in number of parameters.

tor networks together using a combination of distillation and adversarial losses. Lan et al. [18] proposed the on-the-fly native ensemble teacher model, in which the teacher is trained together with multiple students in a multi-branch network architecture. The teacher prediction is a weighted average of the branch predictions.

Knowledge distillation was first introduced as a technique for neural network compression. Resource efficiency considerations have led to a recent increase in interest in efficient neural architectures [13, 14, 23, 32, 44], as well as in algorithms for compressing trained deep networks. Weight pruning methods [11, 20, 22, 24, 33, 35, 39] remove unimportant weights from the network, sparsifying the network connectivity structure. The induced sparsity is unstructured when individual connections are pruned, or structured when entire channels or filters are pruned. Unstructured sparsity usually results in better accuracy but requires specialized sparse matrix multiplication libraries [26] or hardware engines [10] in practice. Quantized networks [8, 15, 17, 30, 43, 46], such as fixed-point, binary, ternary, and arbitrary-bit networks, encode weights and/or activations using a small number of bits, or at lower precision. Fractional or arbitrary-bit quantization [9, 17] encodes individual weights at different precisions, allowing multiple precisions to be used within a single network layer. Low-rank factorization methods [6, 7, 16, 27, 45] produce compact low-rank approximations of filter matrices. Techniques from different categories have also been optimized jointly or combined sequentially to achieve higher compression rates [7, 11, 33].

State-of-the-art network compression methods can achieve significant reductions in network size, in some cases by an order of magnitude, but often require specialized software or hardware support. For example, unstructured pruning requires optimized sparse matrix multiplication routines to realize practical acceleration [26], platform constraint-aware compression [2, 36, 37] requires hardware simulators or empirical measurements, and arbitrary-bit quantization [9, 17] requires specialized hardware. One of the advantages of knowledge distillation is that it is easily implemented in any off-the-shelf deep learning framework without the need for extra software or hardware. Moreover, distillation can be integrated with other network compression techniques for further gains in performance [28].

5. Conclusion

We proposed similarity-preserving knowledge distillation: a novel form of knowledge distillation that aims to preserve pairwise similarities in the student's representation space, instead of mimicking the teacher's representation space. Our experiments demonstrate the potential of similarity-preserving distillation in improving the training outcomes of student networks compared to training with only data supervision (e.g. ground truth labels). Moreover, in a transfer learning setting, when traditional class score based distillation is not directly applicable, we have shown that similarity-preserving distillation provides a robust solution to the challenging domain shift problem. We have also shown that similarity-preserving distillation complements the state-of-the-art attention transfer method and captures teacher knowledge that is not fully encoded in spatial attention maps. We believe that similarity-preserving distillation can provide a simple yet effective drop-in replacement for (or complement to) traditional forms of distillation in a variety of application areas, including model compression [28], privileged learning [21], adversarial defense [25], and learning with noisy data [19].

Future directions. As future work, we plan to explore similarity-preserving knowledge distillation in semi-supervised and omni-supervised [29] learning settings. Since similarity-preserving distillation does not require labels, it is possible to distill further knowledge from the teacher using auxiliary images without annotations. For example, the supervised loss (e.g. cross-entropy) can be computed using the usual annotated training set, while the distillation loss can be computed using an auxiliary set of unlabelled web images. In this setting, the distillation loss is analogous to the reconstruction or unsupervised loss in semi-supervised learning.

References

[1] Anubhav Ashok, Nicholas Rhinehart, Fares Beainy, and Kris M. Kitani. N2N learning: Network to network compression via policy gradient reinforcement learning. In *International Conference on Learning Representations*, 2018.

[2] Changan Chen, Frederick Tung, Naveen Vedula, and Greg Mori. Constraint-aware deep neural network compression. In *European Conference on Computer Vision*, 2018.

[3] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[4] Elliot J. Crowley, Gavin Gray, and Amos Storkey. Moonshine: Distilling with cheap convolutions. In *Advances in Neural Information Processing Systems*, 2018.

[5] Luke N. Darlow, Elliot J. Crowley, Antreas Antoniou, and Amos J. Storkey. CINIC-10 is not ImageNet or CIFAR-10. arXiv:1810.03505, 2018.

[6] Emily Denton, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, 2014.

[7] Abhimanyu Dubey, Moitreya Chatterjee, and Narendra Ahuja. Coreset-based neural network compression. In *European Conference on Computer Vision*, 2018.

[8] Julian Faraone, Nicholas Fraser, Michaela Blott, and Philip H. W. Leong. SYQ: Learning symmetric quantization for efficient deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[9] Josh Fromm, Shwetak Patel, and Matthai Philipose. Heterogeneous bitwidth binarization in convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[10] Song Han, Xingyu Liu, Huizi Mao, Jing Pu, Ardavan Pedram, Mark A. Horowitz, and William J. Dally. EIE: Efficient inference engine on compressed deep neural network. In *ACM/IEEE International Symposium on Computer Architecture*, 2016.

[11] Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. In *International Conference on Learning Representations*, 2016.

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv:1503.02531, 2015.

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

[14] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5mb model size. arXiv:1602.07360, 2016.

[15] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[16] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Speeding up convolutional neural networks with low rank expansions. In *British Machine Vision Conference*, 2014.

[17] Soroosh Khoram and Jing Li. Adaptive quantization of neural networks. In *International Conference on Learning Representations*, 2018.

[18] Xu Lan, Xiatian Zhu, and Shaogang Gong. Knowledge distillation by on-the-fly native ensemble. In *Advances in Neural Information Processing Systems*, 2018.

[19] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *IEEE International Conference on Computer Vision*, 2017.

[20] Zhenhua Liu, Jizheng Xu, Xiulian Peng, and Ruiqin Xiong. Frequency-domain dynamic pruning for convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[21] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations*, 2016.

[22] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. ThiNet: A filter level pruning method for deep neural network compression. In *IEEE International Conference on Computer Vision*, 2017.

[23] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *European Conference on Computer Vision*, 2018.

[24] Sharan Narang, Greg Diamos, Shubho Sengupta, and Erich Elsen. Exploring sparsity in recurrent neural networks. In *International Conference on Learning Representations*, 2017.

[25] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *IEEE Symposium on Security and Privacy*, 2017.

[26] Jongsoo Park, Sheng Li, Wei Wen, Ping Tak Peter Tang, Hai Li, Yiran Chen, and Pradeep Dubey. Faster CNNs with direct sparse convolutions and guided pruning. In *International Conference on Learning Representations*, 2017.

[27] Bo Peng, Wenming Tan, Zheyang Li, Shun Zhang, Di Xie, and Shiliang Pu. Extreme network compression via filter group approximation. In *European Conference on Computer Vision*, 2018.

[28] Antonio Polino, Razvan Pascanu, and Dan Alistarh. Model compression via distillation and quantization. In *International Conference on Learning Representations*, 2018.

[29] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: towards omni-supervised learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[30] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. XNOR-Net: ImageNet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, 2016.

[31] Adriana Romero, Nicolas Ballas, Samira E. Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: hints for thin deep nets. In *International Conference on Learning Representations*, 2015.

[32] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[33] Frederick Tung and Greg Mori. CLIP-Q: Deep network compression learning by in-parallel pruning-quantization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[34] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. KDGAN: Knowledge distillation with generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2018.

[35] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, 2016.

[36] Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze. Designing energy-efficient convolutional neural networks using energy-aware pruning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[37] Tien-Ju Yang, Andrew Howard, Bo Chen, Xiao Zhang, Alec Go, Mark Sandler, Vivienne Sze, and Hartwig Adam. NetAdapt: Platform-aware neural network adaptation for mobile applications. In *European Conference on Computer Vision*, 2018.

[38] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[39] Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I. Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S. Davis. NISP: Pruning networks using neuron importance score propagation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[40] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, 2016.

[41] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *International Conference on Learning Representations*, 2017.

[42] Mengyao Zhai, Lei Chen, Frederick Tung, Jiawei He, Megha Nawhal, and Greg Mori. Lifelong GAN: Continual learning for conditional image generation. In *International Conference on Computer Vision*, 2019.

[43] Dongqing Zhang, Jiaolong Yang, Dongqiangzi Ye, and Gang Hua. LQ-Nets: Learned quantization for highly accurate and compact deep neural networks. In *European Conference on Computer Vision*, 2018.

[44] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[45] Xiangyu Zhang, Jianhua Zou, Xiang Ming, Kaiming He, and Jian Sun. Efficient and accurate approximations of nonlinear convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[46] Aojun Zhou, Anbang Yao, Kuan Wang, and Yurong Chen. Explicit loss-error-aware quantization for low-bit deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.