Training Student Networks for Acceleration with Conditional Adversarial Networks

Zheng Xu
xuzh@cs.umd.edu
Department of Computer Science, University of Maryland, College Park, MD

Yen-Chang Hsu
yenchang.hsu@gatech.edu
Georgia Institute of Technology, Atlanta, GA

Jiawei Huang
jhuang@honda-ri.com
Honda Research Institute, Mountain View, CA

Abstract

There is an increasing interest on accelerating neural networks for real-time applications. We study the student-teacher strategy, in which a small and fast student network is trained with the auxiliary information learned from a large and accurate teacher network. We propose to use conditional adversarial networks to learn the loss function to transfer knowledge from teacher to student. The proposed method is particularly effective for relatively small student networks. Moreover, experimental results show the effect of network size when the modern networks are used as student. We empirically study the trade-off between inference time and classification accuracy, and provide suggestions on choosing a proper student network.

1 Introduction

Deep neural networks (DNNs) achieve massive success in artificial intelligence by substantially improving the state-of-the-art performance in various applications. The accuracy of DNNs for large-scale image classification has become comparable to humans on several benchmark datasets [28]. The recent progress towards such impressive accomplishment is largely driven by exploring deeper and wider network architectures [10, 41]. However, it is difficult to deploy the trained modern networks on embedded systems for real-time applications because of the heavy computation and memory cost. In the meantime, the demand for low cost networks is increasing for applications on mobile devices and autonomous cars.

Do DNNs really need to be deep and wide? Early theoretical studies suggest that shallow networks are powerful and can approximate arbitrary functions [7, 12]. More recent theoretical results show depth is indeed beneficial for the expressive capacity of networks [8, 22, 29, 34]. Moreover, the overparameterized and redundant networks, which can easily memorize and overfit the training data, surprisingly generalize well in practice [43]. Various explanations have been investigated, but the secret of deep and wide networks remains an open problem.

Empirical studies suggest that the performance of shallow networks can be improved by learning from large networks following the student-teacher strategy [2, 35]. In these

approaches, the student networks are forced to mimic the output probability distribution of the teacher networks to transfer the knowledge embedded in the soft targets. The intuition is that the *dark knowledge* [10], which contains the relative probabilities of "incorrect" answers , is informative and representative. For example, we want to classify an image over the label set (dog, cat, car). Given an image of a dog, a good teacher network may mistakenly recognize it as cat with small probability, but should seldom recognize it as car; the soft target of output distribution over categories for this image, $(0.7, 0.3, 0)$, contains more information such as categorical correlation than the hard target of one-hot vector, $(1, 0, 0)$. The student is trained by minimizing a predetermined loss which measures similarity between student and teacher output, such as Kullback-Leibler (KL) divergence.

In previous studies, knowledge transfer has been used to train shallow but wide student networks, which potentially have more parameters than the teacher networks [2, 35]; ensemble of networks are used as teacher, and a student network with similar architecture and capacity can be trained [10]; particularly, a small deep and thin network is trained to replace a shallow and wide network for acceleration [27], given the best teacher at that time is the shallow and wide VGGNet [31]. Since then, the design of network architecture has advanced. ResNet [10] has significantly deepened the networks by introducing residual connections, and wide residual networks (WRNs) [41] suggest widening the networks leads to better performance. It is unclear whether the dark knowledge from the state-of-the-art networks based on residual connections, which are both deep and wide, can help train a shallow and/or thin network (also with residual connections) for acceleration.

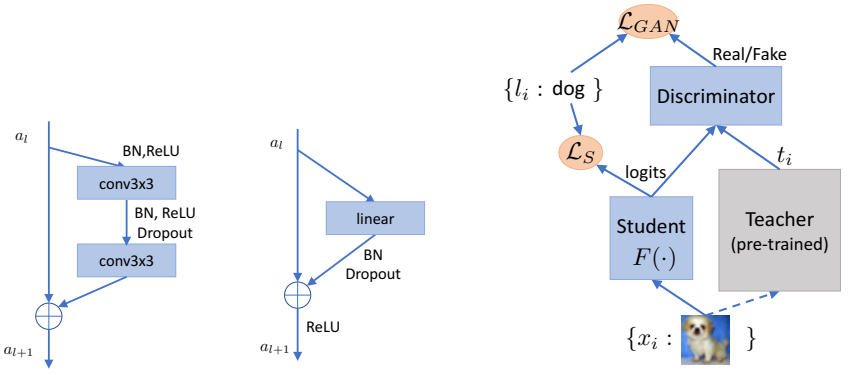
In this paper, we focus on improving the performance of a shallow and thin modern network (student) by learning from the dark knowledge of a deep and wide network (teacher). Both the student and teacher networks are convolutional neural networks (CNNs) with residual connections, and the student network is shallow and thin so that it can run much faster than the teacher network during inference. Instead of adopting the classic student-teacher strategy of forcing the output of a student network to exactly mimic the soft targets produced by a teacher network, we introduce conditional adversarial networks to transfer knowledge from teacher to student. We empirically show that the loss learned by the adversarial training has the advantage over the predetermined loss in the student-teacher strategy, especially when the student network has relatively small capacity.

Our learning loss approach is inspired by the recent success of conditional adversarial networks for various image-to-image translation applications [16]. We show that adversarial nets can benefit a task that is very different from image generation. In the student-teacher strategy, forcing a student network to exactly mimic one of the soft targets (or the average/ensemble of several teacher networks) is not only unnecessary (because of the multi-modality [1]), but also difficult (because the student has smaller capacity). Our approach preserves the multi-modality by introducing an auxiliary network for learning the loss to transfer the knowledge.

1.1 Related work

Network acceleration techniques can be roughly divided into three categories: low precision, sparse parameter pruning, and knowledge distillation. Low precision methods use limited number of bits to store and operate the network weights [20, 26], which often achieve

[1] For the previous example, the output distribution for a dog image can also be (0.8, 0.2, 0). In fact, there are infinite number of soft targets that can correctly predict the label.



(a) Residual blocks for convolutional neural networks [10] (left) and multi-layer perceptron (right). Blocks are equipped with batch normalization (BN), activation ReLU, and dropout. a_l is the output of the lth block.

(b) Proposed adversarial training. The deep and wide teacher is pre-trained offline. The student network and discriminator are updated alternatively. Additional supervised loss is added for both student and discriminator.

Figure 1: Network architectures.

conceptual acceleration because mainstream GPUs have limited support for low precision computation. Networks can be directly modified by pruning and factorizing the redundant weights [13], which aim to construct networks of similar architecture with reduced number of weights by assuming sparsity. Moreover, network pruning papers mostly report indirect speedup measured in the number of basic operations, rather than by inference time.

Knowledge distillation is a principled approach to train small neural networks for acceleration. We slightly generalize the term *knowledge distillation* to represent all methods that train student networks by transferring knowledge from teacher networks. Bucilua et al. [2] pioneered this approach for model compression. Ba and Caruana [2] and Urban et al. [35] trained shallow but wide student by learning from a deep teacher, which were not primarily designed for acceleration. Hinton et al. [11] generalized the previous methods by introducing a new metric between the output distribution of teacher and student, as well as a tuning parameter. Variants of knowledge distillation has also been applied to tasks in other domains [5, 23, 30, 33] A recent preprint [17] presented promising preliminary results on CIFAR-10 by learning a small ResNet from a large ResNet. Another line of research focuses on transferring intermediate features instead of soft targets from teacher to student [14, 27, 36, 39, 40, 42]. Our approach is complementary to those methods by using adversarial networks to learn a new metric between the output distribution of teacher and student.

Generative adversarial networks (GAN) has been extensively studied over recent years since [9]. GAN trains two neural networks, the generator and the discriminator, in an adversarial learning process that alternatively updates the two networks. We use adversarial networks conditioned on input images [16, 25, 37]. Unlike previous works that focused on image generation, we aim at learning a loss function for knowledge distillation, which requires quite different architectural choices for our generator and discriminator. A recent preprint [3] appears a few months later than ours has a similar approach for network compression. We are the first to apply adversarial training for knowledge distillation. Moreover, we provide systematical study on choosing the student.

2 Learning loss for knowledge distillation

In this section, we introduce the learning loss approach based on conditional adversarial networks. We start from a recap of modern network architectures (section 2.1), and then describe the dark knowledge that can be transferred from teacher to student networks (section 2.2). Our approach with adversarial networks for learning loss is detailed in section 2.3.

2.1 Neural networks with residual connection

Residual blocks are shown to be effective for training deep CNNs to achieve state-of-the-art performance [10, 21, 41]. We build both student and teacher networks by stacking the residual convolutional blocks shown in Figure 1a (left). The first layer contains 16 filters of 3×3 convolution, followed by a stack of $6n$ layers, which is 3 groups of n residual blocks, and each block contains two convolution layers equipped with batch normalization [15], ReLU [19] and dropout [32]. The output feature map is subsampled twice, and the number of filters are doubled when subsampling . After the last residual block is the global average pooling, and then fully-connected layer and softmax. In the following sections, the architecture of wide residual networks (WRNs) is denoted as WRN-d-m following [41], where the total depth is $d = 6n+4$, and m is the widen factor that increases the number of filters by m times in each residual block. Our teacher network is deep and wide WRN with large d and m, while student network is shallow and thin WRN with small d and m.

2.2 Knowledge distillation

The output of neural networks for image classification is a probability distribution over categories, which is generated by applying a softmax function over the output of the last fully connected layer (known as *logits*). Rich information is embedded in the output of a teacher network, and we can use logits to transfer the knowledge to student network [2, 4, 11, 35]. We review [11] that generalized previous methods, which provides a metric between student and teacher logits for *knowledge distillation (KD)*.

The logits vector generated by pre-trained teacher network for an input image $x_i, i = 1,\ldots,N$ is represented by t_i, where the dimension of vector $t_i = (t_i^1,\ldots,t_i^C)$ is the number of categories C. We now consider training a student network F to generate student logits $F(x_i)$. By introducing a parameter called temperature T, the generalized softmax layer can convert logits vector t_i to probability distribution q_i,

$$M_T(t_i) = q_i, \text{ where } q_i^j = \exp(t_i^j/T)/\textstyle\sum_k \exp(t_i^k/T). \tag{1}$$

where higher temperature T produces softer probability over categories. The regular softmax for classification is a special case of the generalized softmax with $T = 1$.

Hinton et al. [11] proposed to minimize the KL divergence between teacher and student,

$$\mathcal{L}_{KD}(F,T) = 1/N \sum_{i=1}^{N} \mathrm{KL}(M_T(t_i) \| M_T(F(x_i))), \tag{2}$$

and show that when T is very large, $\mathcal{L}_{KD}$ becomes the Euclidean distance between teacher and student logits. Given the image-label pairs $\{x_i, l_i\}$, the cross-entropy loss for supervised training of a neural network is

$$\mathcal{L}_S(F) = 1/N \sum_{i=1}^{N} \mathcal{H}(l_i, M_1(F(x_i))), \tag{3}$$

which is widely used for standard supervised learning. Finally, Hinton et al. [11] proposed to minimize the weighted sum of $\mathcal{L}_{KD}$ and $\mathcal{L}_S$ to train a student network,

$$\mathcal{L}_1(F,T) = 1/2\mathcal{L}_S(F) + T^2\mathcal{L}_{KD}(F,T). \tag{4}$$

2.3 Learning loss with adversarial networks

Overview. The main idea of learning the loss for transferring knowledge from teacher to student is depicted in Figure 1b. Instead of forcing the student to exactly mimic the teacher by minimizing KL-divergence in $\mathcal{L}_1(F,T)$ of Equation (4), the knowledge is transferred from teacher to student through a discriminator in our approach. This discriminator is trained to distinguish whether the output logits is from teacher or student network, while the student is adversarially trained to fool the discriminator, i.e., output logits that are indistinguishable to the teacher logits.

There are several benefits of the proposed method. First, the learned loss is often effective, as has already been demonstrated for several image to image translation tasks [16]. Moreover, our approach relieves the pain for hand-engineering the loss. Though the parameter tuning and hand-engineering of the loss is replaced by hand-engineering the discriminator networks in some sense, our empirical study shows that the performance is less sensitive to the discriminator architecture than the temperature parameter in knowledge distillation. The second benefit is closely related to the multi-modality of network output. As discussed before, it is unnecessary and difficult to exactly mimic the output of teacher networks. The trained discriminator can capture the relative similarities between the categories from the multi-modal logits of teacher, and directs the student to produce correct but not necessarily same outputs as the teacher.

Discriminator update. We now describe the proposed method in a more rigorous way. The student and discriminator in Figure 1b are alternatively updated in the proposed approach. Let us first look at the update of the discriminator, which is trained to distinguish teacher and student logits. We use multi-layer perceptron (MLP) as discriminator. Its building block — residual block is shown in Figure 1a (right). The number of nodes in each layer is the same as the dimension of logits, i.e., the number of categories C. We denote the discriminator that predicts binary value "Real/Fake" as $D(\cdot)$. To train D, we fix the student network $F(\cdot)$ and seek to maximize the log-likelihood, which is known as binary cross-entropy loss,

$$\mathcal{L}_A(D,F) = 1/N \sum_{i=1}^{N} \Big(\log P(\text{Real}|D(t_i)) + \log P(\text{Fake}|D(F(x_i))) \Big). \tag{5}$$

The plain adversarial loss $\mathcal{L}_A$ for knowledge distillation, which follows the original GAN [9], faces two major challenges. First, the adversarial training process is difficult [38]. Even if we replace the log-likelihood with advanced techniques such as Wasserstein GAN [1] or Least Squares GAN [24], the training is still slow and unstable in our experiments. Second, the discriminator captures the high-level statistics of teacher and student outputs, but the low-level alignment is missing. The student outputs $F(x_i)$ for x_i can be aligned to a completely unrelated teacher sample t_j by optimizing $\mathcal{L}_A$, which means a dog image can generate a logits vector that predicts cat. One extreme example is that the student always mispredicts dog as cat and cat as dog, but the overall output distribution may still be close to the teacher's.

To tackle these problems, we modify the discriminator objective to also predict the class labels, where the output of discriminator $D(\cdot)$ is a $C+1$ dimensional vector with C *Label* predictions and a *Real/Fake* prediction. We now maximize

$$\mathcal{L}_{\text{Discriminator}}(D,F) = 1/2(\mathcal{L}_A(D,F) + \mathcal{L}_{DS}(D,F)), \tag{6}$$

where $\mathcal{L}_A$ is the previously defined adversarial loss over *Real/Fake*, $\mathcal{L}_{DS}$ is the supervised log-likelihood of discriminator over *Labels*, written as

$$\mathcal{L}_{DS}(D,F) = 1/N \sum_{i=1}^{N} \Big(\log P(l_i|D(t_i)) + \log P(l_i|D(F(x_i))) \Big). \tag{7}$$

We assume *Label* and *Real/Fake* are conditionally independent in Equation (6). To avoid using this assumption, we can maximize the log-likelihood of discriminator to predict the tuple { *Label, Real/Fake* }, which requires $D(\cdot)$ to predict a $2C$ dimensional vector. In our experiments, optimizing the proposed method with or without the independent assumption achieves almost identical results. Hence we will always use the independent assumption for a more compact discriminator. Note that equation (6) has the same form as the auxiliary classifier GANs [25, 37].

The adversarial training becomes much more stable when the proposed discriminator also predicts category *Labels* besides *Real/Fake*. Moreover, the discriminator can provide category-level alignment between outputs of student and teacher. The student outputs of a dog image are more likely to learn from the teacher outputs that predict dogs. However, the proposed method still lacks instance-level knowledge. To further boost the performance, we start with investigating conditional discriminators, in which the input of discriminators are logits concatenated with a conditional vector. We tried the following conditional vectors: image with convolutional embedding; label one-hot vector with embedding; and the extracted teacher logits. However, it turns out the conditional vectors are easily ignored during the training of the discriminator and does not help in practice. We will introduce a direct instance-level knowledge for training student network later.

Student update. We update the student network after updating the discriminator in each iteration. When updating the student network $F(\cdot)$, we aim to fool the discriminator by fixing discriminator $D(\cdot)$ and minimizing the adversarial loss $\mathcal{L}_A$. In the meantime, the student network is also trained to satisfy the auxiliary classifier of discriminator $\mathcal{L}_{DS}$. Besides the category-level knowledge in $\mathcal{L}_{DS}$, we introduce instance-level knowledge by aligning outputs of teacher and student,

$$\mathcal{L}_{L_1}(F) = {}^1\!/_N \sum_{i=1}^{N} \|F(x_i) - t_i\|_1. \tag{8}$$

The L_1 norm has been found helpful in the GAN-based image to image translation [16].

Finally, we combine the learned loss with the supervised loss $\mathcal{L}_S$ in (3), and minimize the following objective for the student network $F(\cdot)$,

$$\mathcal{L}_{\text{Student}}(D,F) = \mathcal{L}_S(F) + \mathcal{L}_{L_1}(F) + \mathcal{L}_{GAN}(D,F), \text{where } \mathcal{L}_{GAN}(D,F) = \frac{1}{2}(\mathcal{L}_A(D,F) - \mathcal{L}_{DS}(D,F)). \tag{9}$$

The sign of $\mathcal{L}_{DS}$ is flipped in (6) and (9) because both the discriminator and student are trained to preserve the category-level knowledge.

Our final loss $\mathcal{L}_{\text{Student}}(D,F)$ in (9) is a combination of the learned loss for knowledge distillation and the supervised loss for neural network, and may look complicated at the first glance. However, each component of the loss is relatively simple. Moreover, since both student F and discriminator D are learned, there is no explicit parameters to be tuned in the loss function. Our experiments suggest the performance of the proposed method is reasonably insensitive to the discriminator architecture and the learned loss can outperform the hand-engineered loss for knowledge distillation.

3 Experiments

After presenting experimental settings, we show the benefits of our proposed method in section 3.1 and perform ablation study in section 3.2. We present the effect of depth and width of the student network in section 3.3, followed by the discussion of trade-off between classification accuracy and inference time in section 3.4.

We consider three image classification datasets: ImageNet32 [6], CIFAR-10 and CIFAR-100 [18], and use wide residual networks (WRNs) [41] for both student and teacher networks. The teacher network is a fixed WRN-40-10, while the student network has varying depth and width in different experiments. We use multi-layer preceptron (MLP) as the discriminator in our approach. 3-layer MLP is used for most of the experiments except for section 3.2, in which we study the effect of discriminator depth. To speed up the experiments, the logits of teacher network are generated offline and stored in memory. We use stochastic gradient descent (SGD) as optimizer and follow standard training scheduler, and set dropout ratio to 0.3 for both discriminator and student networks. The results below are the median of five random runs.

	CIFAR-10	CIFAR-100	ImageNet32
Student	7.46	28.52	48.2
Teacher	4.19	20.62	38.41
KD (T=1)	7.27	28.62	49.37
KD (T=2)	7.3	28.33	49.48
KD (T=5)	7.02	27.06	49.63
KD (T=10)	6.94	27.07	51.12
Ours	**6.09**	**25.75**	**47.39**

Table 1: Error rate achieved on benchmark datasets.

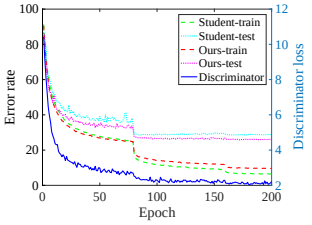
3.1 Benefits of learning loss

We first show the proposed method is effective for transferring knowledge from teacher to student. Table 1 shows the error rate of classification on the three benchmark datasets. The teacher is the deep and wide WRN-40-10. The student is much shallower and thinner, WRN-10-4 for CIFARs, and WRN-22-4 for ImageNet32. We choose a larger student network for ImageNet32 because it contains more samples and categories. We will have more discussion on wisely choosing the student architecture in sections 3.3 and 3.4 . The first two rows of Table 1 show the performance of standard supervised learning for student and teacher networks, without knowledge transfer. We then compare our approach with knowledge distillation (KD) in [11]. We choose the temperature parameter $T \in \{1, 2, 5, 10\}$ following the original work. No parameter is tuned for our method.

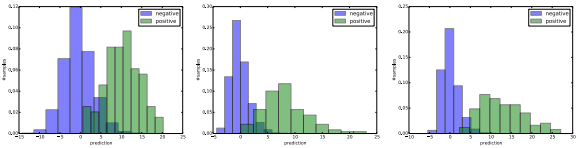
In Table 1, the deep and wide teacher performs much better than the shallow and thin student with standard supervised learning, and lower bounds the error rate of the small network trained with student-teacher strategy. Baseline method KD helps the training of small networks for the two CIFARs, but does not help for ImageNet32. We conjecture the reason to be that the capacity of the student is too small to learn from knowledge distillation for larger dataset such as ImageNet32. The temperature parameter T introduced in KD is useful. For CIFARs, KD performs better when T is large, and $T = 5$ and $T = 10$ performs similarly. The proposed method improves the performance of small network for all three datasets, and outperforms KD by a margin.

3.2 Analysis of the proposed method

We discuss the proposed method in more details. Figure 2a presents the training curve of the small student network, WRN-10-4, on CIFAR-100 dataset. The loss of the discriminator (blue solid line) is gradually decreasing, which suggests the adversarial training steadily makes progress. The error rates of the proposed method for both training and testing data are



(a) The training curve on CIFAR-100.



(b) The distribution of prediction for category 85 in CIFAR-100..

Figure 2: Analysis of the proposed method.

Loss composition	CIFAR-10	CIFAR-100
$\mathcal{L}_S$	7.46	28.52
$\mathcal{L}_{GAN}$	14.82	47.04
$\mathcal{L}_S + \mathcal{L}_{GAN}$	6.56	27.27
$\mathcal{L}_S + \mathcal{L}_{L_1}$	6.44	26.66
$\mathcal{L}_S + \mathcal{L}_{L_1} + \mathcal{L}_{GAN}$	**6.09**	**25.75**

Table 2: The effect of different components of the loss in the proposed method.

decreasing. The testing error rate of the proposed method is consistently better than the pure supervised training of the student model, and looks more stable between epoch 50-100. The training error rate of the proposed method is slightly worse than pure supervised learning, which suggests knowledge transfer can benefit generalization.

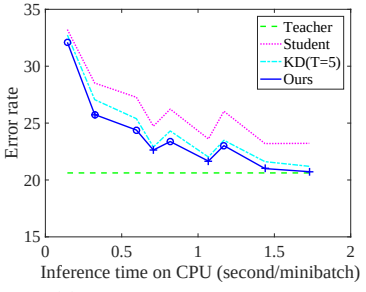
Next, we performing ablation study on components of the proposed approach, as shown in Table 2. By combining the adversarial loss and the category-level knowledge transfer (Equation (6)), the learned loss $\mathcal{L}_{GAN}$ performs reasonably well. However, the indirect knowledge provided by $\mathcal{L}_{GAN}$ alone is not as good as standard supervised learning $\mathcal{L}_S$. Both category-level knowledge transferred by $\mathcal{L}_{GAN}$ and instance-level knowledge transferred by $\mathcal{L}_{L_1}$ can improve the performance of training student network. Our final approach combines these components and performs the best without parameter tuning.

We present the effect of the depth of MLP as discriminator in Table 3. The error rate is relatively insensitive to the depth of discriminator. The error rate slightly decreases as the depth increases when the discriminator is generally shallow. When the discriminator becomes deeper, the error rate increases as the adversarial training becomes unstable. Decreasing the learning rate of discriminator sometimes helps, but it may introduce parameter tuning. The 3-layer MLP works reasonably well and is used for all our experiments to keep the proposed method simple.

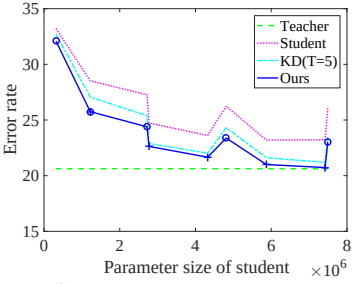
Finally, we present qualitative visualization for the proposed approach. Figure 2b shows the scaled histogram for the prediction of category 85 in CIFAR-100. The histogram is calculated on the 10K testing samples, in which 100 samples are from category 85 and labeled as positive (green in figure), and the other 9.9K are labeled as negative (blue in the figure). The histogram is normalized to sum up to one for positives and negatives, respectively. The three plots represent the distribution predicted by student network trained by standard supervised learning, the student network trained by the proposed approach, and the teacher network. The histogram in the middle is similar to the histogram on the right, which suggests the proposed approach effectively transfers knowledge from teacher to student.

Depth	1	2	3	4
Error rate	26.13	25.88	**25.75**	27.42

Table 3: The effect of discriminator depth on CIFAR-100.



(a) Inference time versus error rate.



(b) Network size versus error rate.

Figure 3: Trade-off of error rate to inference time and parameter size. The figure is generated from Table 4. Networks WRN-10-m are labeled as circles, and WRN-d-4 are labeled as crosses for the proposed approach. The largest student is 7x smaller and 5x faster than the teacher WRN-40-10.

3.3 Does WRN need to be deep and wide?

Urban et al. [35] asked the question for convolutional neural networks and claimed the network should at least has a few layers of convolutions. In this section, we study the modern architecture WRN of residual blocks, and show that even for the modern architecture WRN, the network has to be deep and wide to some extent. Table 4 presents the results of standard supervised learning, knowledge distillation [11] and the proposed approach for different student networks trained on CIFAR-100. We first fix the depth of WRN as 10, and change the widen factor from 2 to 10. We then fix the width as 4, and increase depth from 10 to 34. The parameter size is in millions, and the inference time is measured in seconds per minibatch of 100 samples on CPU.

When the student is very small, such as WRN-10-2, it is difficult to transfer knowledge from teacher to student because the student is limited by its capacity. When the student is large, such as WRN-34-4, both KD and the proposed approach can improve the performance to approximate the teacher. The advantage of the proposed method is observed at all depths and widths but is most pronounced for relatively small students such as WRN-10-4. Increasing depth is more effective than width. For example, WRN-34-4 has less parameter than WRN-10-10, but achieves lower error rate.

WRN	Size (M)	Time (s)	Student	KD (T=5)	**Ours**
10-2	0.32	0.14	33.22	32.74	**32.1**
10-4	1.22	0.32	28.52	27.16	**25.75**
10-6	2.72	0.60	27.27	25.39	**24.39**
10-8	4.81	0.82	26.23	24.31	**23.38**
10-10	7.49	1.17	26.04	23.49	**23.02**
16-4	2.77	0.71	24.73	22.9	**22.73**
22-4	4.32	1.07	23.61	22.02	**21.66**
28-4	5.87	1.44	23.2	21.61	**21.00**
34-4	7.42	1.73	23.22	21.2	**20.73**
40-10	55.9	8.73	20.62	-	**-**

Table 4: The effect of depth and width in student network; the parameter size, inference time and error rate on CIFAR-100.

3.4 Training student for acceleration

The shallow and thin network is much easier to deploy in practice. We present the trade-off between error rate, inference time and parameter size in Figure 3. The figure is generated by changing the architecture of the student network. Larger student network is more accurate but also slower. For network with similar size, such as WRN-10-10 and WRN-34-4, deeper network achieves lower error rate, while wider network runs slightly faster. When the student network is relatively large, such as WRN-34-4, the student network trained by the proposed approach can achieve competitive error rate as the teacher WRN-40-10, while being 7x smaller and 5x faster. The proposed approach also decreases the absolute error rate by 2.5% compared to the standard training without knowledge transfer.

4 Conclusion and discussion

We study the student-teacher strategy for network acceleration in this paper. We propose to use adversarial networks to learn the loss for transferring knowledge from teacher to student. We show that the proposed approach can improve the training of student network, especially when the student network is shallow and thin. Moreover, we empirically study the effect of network capacity when adopting modern network as student and provide guidelines for wisely choosing a student to balance error rate and inference time. We can train a student that is 7x smaller and 5x faster than teacher without loss of accuracy.

The proposed approach is stable and easy to implement after applying several advanced techniques in the GAN literature. The current implementation uses the stored logtis from teacher network to save GPU memory and computation. Generating teacher logits on the fly can be more reliable for the adversarial training. Moreover, the proposed approach can be naturally extended to use ensemble of networks as teacher. The logits of multiple teacher networks can be fed into the discriminator for better performance. We will investigate these ideas for future work.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *ICML*, 2017.

[2] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *NIPS*, 2014.

[3] Vasileios Belagiannis, Azade Farshad, and Fabio Galasso. Adversarial network compression. *arXiv preprint arXiv:1803.10750*, 2018.

[4] Cristian Bucilua, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *KDD*, 2006.

[5] Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Darkrank: Accelerating deep metric learning via cross sample similarities transfer. *arXiv preprint*, 2017.

[6] Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint*, 2017.

[7] George Cybenko. Approximation by superpositions of a sigmoidal function. *MCSS*, 1989.

[8] Ronen Eldan and Ohad Shamir. The power of depth for feedforward neural networks. In *COLT*, 2016.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint*, 2015.

[12] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 1989.

[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 preprint*, 2017.

[14] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint*, 2017.

[15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

[17] Seung Wook Kim and Hyo-Eun Kim. Transferring knowledge to smaller network with class-distance loss. *ICLR Workshop*, 2017.

[18] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[20] Hao Li, Soham De, Zheng Xu, Christoph Studer, Hanan Samet, and Tom Goldstein. Training quantized nets: A deeper understanding. *arXiv preprint*, 2017.

[21] Hao Li, Zheng Xu, Gavin Taylor, and Tom Goldstein. Visualizing the loss landscape of neural nets. *arXiv preprint arXiv:1712.09913*, 2017.

[22] Shiyu Liang and R Srikant. Why deep neural networks for function approximation? *ICLR*, 2017.

[23] Ping Luo, Zhenyao Zhu, Ziwei Liu, Xiaogang Wang, Xiaoou Tang, et al. Face model compression by distilling knowledge from neurons. In *AAAI*, 2016.

[24] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. *arXiv preprint*, 2016.

[25] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. *ICML*, 2017.

[26] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. XNOR-net: Imagenet classification using binary convolutional neural networks. In *ECCV*, 2016.

[27] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *ICLR*, 2015.

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[29] Itay Safran and Ohad Shamir. Depth-width tradeoffs in approximating natural functions with neural networks. In *ICML*, 2017.

[30] Jonathan Shen, Noranart Vesdapunt, Vishnu N Boddeti, and Kris M Kitani. In teacher we trust: Learning compressed models for pedestrian detection. *arXiv preprint*, 2016.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint*, 2014.

[32] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[33] Yee Whye Teh, Victor Bapst, Wojciech Marian Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. *arXiv preprint*, 2017.

[34] Matus Telgarsky. Benefits of depth in neural networks. *COLT*, 2016.

[35] Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, and Matt Richardson. Do deep convolutional nets really need to be deep and convolutional? *ICLR*, 2017.

[36] Jingdong Wang, Zhen Wei, Ting Zhang, and Wenjun Zeng. Deeply-fused nets. *arXiv preprint*, 2016.

[37] Zheng Xu, Michael Wilber, Chen Fang, Aaron Hertzmann, and Hailin Jin. Beyond textures: Learning from multi-domain artistic images for arbitrary style transfer. *arXiv preprint arXiv:1805.09987*, 2018.

[38] Abhay Yadav, Sohil Shah, Zheng Xu, David Jacobs, and Tom Goldstein. Stabilizing adversarial nets with prediction methods. *ICLR*, 2018.

[39] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. *CVPR*, 2017.

[40] Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *KDD*, 2017.

[41] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint*, 2016.

[42] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *ICLR*, 2017.

[43] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *ICLR*, 2017.