Robust Knowledge Transfer via Hybrid Forward on the Teacher-Student Model

Liangchen Song, [1] **Jialian Wu,** [1] **Ming Yang,** [2] **Qian Zhang,** [2] **Yuan Li,** [3] **Junsong Yuan** [1]
[1]University at Buffalo [2]Horizon Robotics [3]Google
{lsong8,jialianw,jsyuan}@buffalo.edu, m-yang4@u.northwestern.edu, qian01.zhang@horizon.ai, liyu@google.com

Abstract

When adopting deep neural networks for a new vision task, a common practice is to start with fine-tuning some off-the-shelf well-trained network models from the community. Since a new task may require training a different network architecture with new domain data, taking advantage of off-the-shelf models is not trivial and generally requires considerable try-and-error and parameter tuning. In this paper, we denote a well-trained model as a teacher network and a model for the new task as a student network. We aim to ease the efforts of transferring knowledge from the teacher to the student network, robust to the gaps between their network architectures, domain data, and task definitions. Specifically, we propose a hybrid forward scheme in training the teacher-student models, alternately updating layer weights of the student model. The key merit of our hybrid forward scheme is on the dynamical balance between the knowledge transfer loss and task specific loss in training. We demonstrate the effectiveness of our method on a variety of tasks, *e.g*, model compression, segmentation, and detection, under a variety of knowledge transfer settings.

Introduction

The flourish and success of deep learning community attribute greatly to researchers who released their well-trained neural network models for generic vision tasks, such as object recognition (Simonyan and Zisserman 2014), detection (Ren et al. 2015), and image segmentation (Chen et al. 2017). These off-the-shelf models, such as VGG (Simonyan and Zisserman 2014) and ResNet (He et al. 2016), are often employed as the backbone networks and a wise starting point for new vision tasks. Leveraging the knowledge encoded in a backbone model and transferring them to a new model generally involves considerable efforts on tuning network parameters and try-and-error, due to the wide variety of vision tasks and training data distributions. In the paper, we borrow the terms in knowledge distillation (Hinton, Vinyals, and Dean 2015) and refer an off-the-shelf well-trained model as a teacher network and a model dedicated for the new task as a student network. This is in the sense that we strive to transfer the knowledge, *i.e.*, the strong feature extraction capability, from *teacher* networks to *student* networks for new vision tasks.

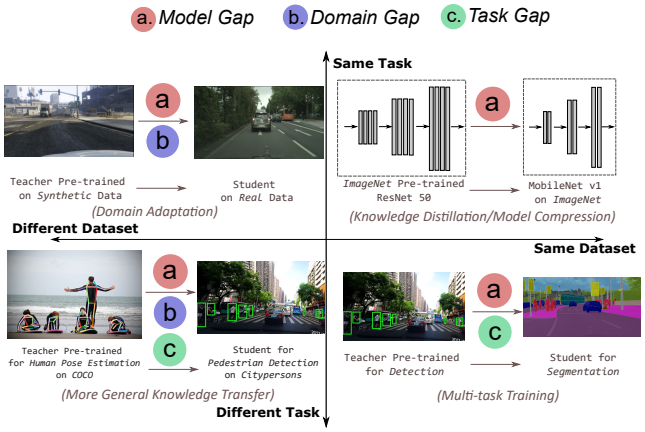

Figure 1: Illustration of different settings for transferring a well-trained teacher network to a student network, with the model, domain, and task gaps defined in this paper. We validate the generality of the proposed hybrid forward extensively on diverse knowledge transferring settings in the experiments.

Fine-tuning a well-trained network tends to be more time efficient and effective than training from scratch. Nevertheless, there are many issues to handle in practice. The diverse vision tasks require different network architectures, for example, object recognition on a mobile phone requires a lighter network than those trained for ImageNet recognition on a server, or image segmentation requires a network to output per pixel labeling than per image classification in scene recognition. Moreover, data in different domains generally present different distributions or bias, *e.g.*, synthetic street-view data generated by a simulator *vs.* real-world scenes collected by multiple cameras. These challenges originate from the disparate training settings for well-trained teacher and new student networks. We abbreviate the differences of network architectures, domain data distributions, and the task definitions as the *model gap*, *domain gap*, and *task gap* between the teacher and student networks, as illustrated in Fig. 1. Last but not the least, in transferring from a well-trained teacher network to a student, it is critical yet tricky to balance the transfer loss and the task loss, *i.e.*, how much to tune the backbone towards the new data and task.

In this paper, we investigate how to leverage knowledge from a well-trained teacher network to obtain a student network for new vision tasks. We propose a robust knowledge transfer method embedded in the core neural network training procedure – feedforward pass and back-propagation. In training a neural network, the forward pass processes data samples through the network and the back-propagation updates the network weights guided by a loss function. The knowledge a network learns is encoded and memorized in the connection weights and the parameters of neurons. Therefore, in another word, the forward pass applies the knowledge in the network to data samples and the back-propagation updates the knowledge by the task specific loss. In our method, we integrate the teacher and student networks and train the combined network on the new task. Specifically, the forward pass processes data samples alternately through a sub-network of layers of one network and then some other layers of the other network as shown in Fig. 2, which we coin it as the *hybrid forward*. The pre-requisites of this method are that 1) the spatial resolutions of feature maps are scaled down accordingly for both teacher and student networks; 2) the separation between the backbone network for feature extraction and the output head network for specific tasks. These pre-requisites are readily satisfied in practice with some manipulation of network layers.

In the proposed hybrid forward scheme, the knowledge transfer from trained teacher networks to new student networks is supervised by the loss function and training samples in the new task. That is, the training loss of knowledge transfer is evaluated by the same loss function in the target task, which resolves a critical and tricky tradeoff in training a teacher-student pair. More precisely, after the sample batches going through the teacher network, they are sent into the task head of the student network to compute the loss with the ground truth labels. Thus, we can measure the teacher network's relative impact on student network training. Therefore, we are relieved from tuning how to balance the hybrid forward, by treating the knowledge transfer loss and the task loss equally important. This robust knowledge transfer deals with the aforementioned gaps in Fig. 1 between the training settings of a teacher and student network. The technical contributions are as follows:

- We propose a novel hybrid forward scheme to pass the knowledge from a teacher network to a student network, enabling efficient transfer feature extraction capability for a new task.
- We design a dynamic balancing strategy by employing the loss function in a target task for knowledge transfer, thus avoid manually tuning the tradeoff in our hybrid forward scheme.

We validate the generality of the proposed robust knowledge transfer on diverse vision tasks under a variety of settings, *e.g.*, different network architectures, different domain data, or different target tasks. To list a few: model compression from ResNet to MobileNetV1 on ImageNet, domain adaptation from synthetic to real-world image segmentation on CityScapes, and utilizing pose estimation models for pedestrian detection on CityPersons. The proposed method effectively leverages well-trained networks on new tasks and demonstrates competitive results with the SOTA.

Related Work

Leveraging a well-trained network for a new task has been widely studied under many different settings. The practices to transfer feature extraction capability from an off-the-shelf network to a new network can be roughly categorized into two classes: initialization based and teacher-student based. In contrast, the proposed hybrid forward scheme constructs a joint teacher-student network and update the sub-networks in an alternating way.

Pre-trained model as an initialization

The straightforward practice to utilize a pre-trained model is to use it to initialize the backbone network, which is fine-tuned with additional output layers by new training data. For example, reusing the classification models trained on ImageNet often transfers abundant feature extraction knowledge to other vision tasks. In these cases, usually models with the same network architecture are fine-tuned on different data and tasks, which are referred as domain and task gaps in this paper. Later on, Net2Net (Chen, Goodfellow, and Shlens 2016) explored the possibility of knowledge transfer in presence of model gap. Similarly, (Wei et al. 2016; Wei, Wang, and Chen 2019; Fang et al. 2019, 2020) proposed to migrate the weights of a well-trained network to anther network with a different network architecture.

Knowledge distillation and feature mimicking

In (Hinton, Vinyals, and Dean 2015), Hinton et al. first proposed the knowledge distillation, *i.e.*, using a cumbersome network as a teacher to generate soft labels to supervise the training of a compact student network. Later the teacher-student training scheme has been extended in (Romero et al. 2015; Zagoruyko and Komodakis 2017; Yim et al. 2017), trying to match the intermediate features between the teacher and student network. More recent methods (Srinivas and Fleuret 2018; Kim, Park, and Kwak 2018; Heo et al. 2019b,a; Peng et al. 2019; Tung and Mori 2019; Wu et al. 2020c,b,a) designed delicate transformations to ensure only useful knowledge in a teacher network is transferred to a student. All of these above methods focus on obtaining a light-weight student model from the teacher, for the same task and on the same dataset, which fall to the *model gap* cases as only the network architectures are different. Further, (Kundu, Lakkakula, and Babu 2019; Gong et al. 2021) transferred knowledge among tasks for unsupervised domain adaptation. (Ye et al. 2019; Shen et al. 2019) studied the problem of learning from multiple teachers, which is referred to as knowledge amalgamation. In our work, we are concerned with how to effectively exploit the knowledge in a well-trained network for a new task, with no access to previous training data.

Sub-network training

In the proposed hybrid transfer, we first construct a large network joining the teacher and student networks, then for

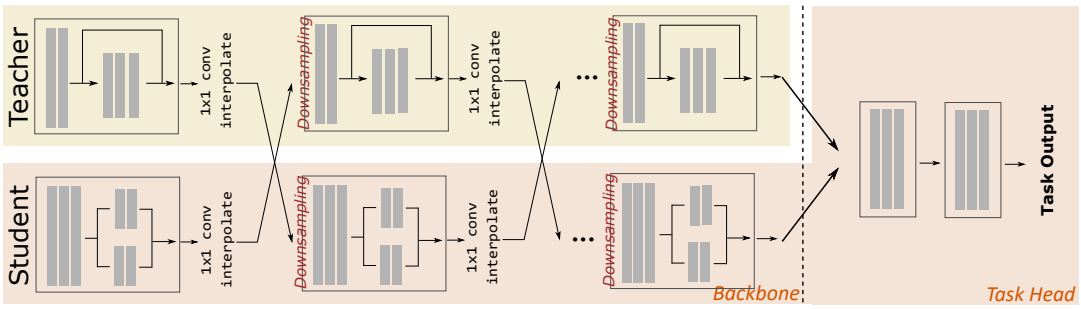

Figure 2: We propose a hybrid forward scheme to transfer the knowledge encoded in a teacher network to a student network. During the training process of a student network, the feedforward passes the training data through a sub-network of layers of the student network and then a sub-network of a teacher network alternately, while the back-propagation updates the student network only and leaves the teacher network fixed.

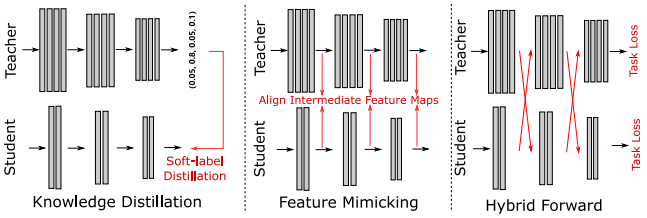

Figure 3: Existing knowledge transfer schemes and our proposed hybrid forward scheme. (Left) Knowledge distillation supervises the training of a compact student network by the soft labels generated by a teacher. (Middle) Feature mimicking aligns the intermediate feature maps. (Right) Our hybrid forward scheme passes the sub-networks of a teacher and student network alternately for transferring the knowledge.

each data batch, sample a sub-network to train in a particular way. The idea of training only a sub-network in a large network has been adopted for optimizing the network architecture (Yu et al. 2019; Yu and Huang 2019; Bender et al. 2018). Besides, Gao et al. (Gao et al. 2019) proposed to fuse the intermediate feature maps for multi-task training. A more related recent work (Xu et al. 2020) used the hybrid forwarding scheme for compressing NLP models. Our method works in a different way from the above methods, by alternating the sub-networks from the teacher and student networks in feedforward, while updating the student network alone in back-propagation.

Proposed Method

We strive to develop a general method that enables transferring knowledge from a well-trained teacher network to a student network for new vision tasks, robust to the model, domain and task gaps. We propose a hybrid forward scheme, which integrates a teacher network into the forward pass of a student network and process training samples through sub-networks in a particular manner. This is a novel approach taking advantage of off-the-shelf well-trained models, as illustrated in Fig. 3, in contrast to knowledge distillation (Hinton, Vinyals, and Dean 2015) deriving soft-labels from a teacher for the student network training, or feature mimicking (Srinivas and Fleuret 2018; Kim, Park, and Kwak 2018; Heo et al. 2019b) aligning feature maps between a teacher and student network.

To train the network that integrates both the teacher and student on a new task, we need to properly scale the outputs of loss functions so that they are on a common ground to compare. Thus, we propose to balance the losses of teacher and student networks dynamically by scaling the loss of a teacher network *w.r.t* its output of some training data batches.

Hybrid Forward

We design a novel and effective hybrid forward scheme, in which the forward pass is computed on the sub-networks of a student and a teacher network alternately. As demonstrated in Fig. 2, there are three components in our methods: the teacher backbone, the student backbone and the student head for the current task. During the training process of of the integrated teacher and student network, the input data will flow through one sub-network of the student and then one sub-network of the teacher in an alternating fashion. Specifically, the forward pass will switch to another network before entering the downsampling layer. The reason why we switch before the downsampling layer is due to the convention that the feature maps before the downsampling layer are commonly used for task specific predictions. That is, the feature map extracted right before the downsampling layer is a good representation at that image scale. Many structures are proposed to fully exploit the knowledge contained in these intermediate outputs, such as FPN (Lin et al. 2017).

However, the network width, the channels of the feature maps of the two networks may be different, which is an issue preventing us from directly using the feature maps from the teacher as inputs to the student. To solve this issue, we add 1×1 convolutions before switching in the forward path. In other words, each feature map is processed by a 1×1 convolution to match the required shape of the next stage. Although the scales between the two stages are the same, the size of the feature map may not be the same probably due to different padding settings. So, in some cases, we also need an interpolation step after the 1×1 convolution layer.

Finally, after we obtain the feature maps generated by for-

warding with sub-networks from both the teacher and the student, we send the feature maps to the student head network, which generates outputs for the current task. Note that during the above training process, all sub-networks from the teacher network are frozen and not updated by back-propagation, to preserve the feature extraction knowledge in the teacher network unchanged.

Dynamic Balancing

Through the proposed hybrid forward scheme, training samples may start with the sub-network of the teacher or the sub-network of the student, thus this results in two paths to the student task head and two loss function outputs. Both of these two losses are evaluated by the samples and loss function in the current task, which is the major difference between our method and previous knowledge distillation methods. In other words, transferring knowledge is achieved by minimizing the loss related to the current task, *i.e.*, the task that the student network aims to solve.

Given using the same task related loss in knowledge transfer, we are able to quantitatively evaluate the teacher network's impact to training the student network. The ability of measuring the teacher and student networks on a common ground is the vital factor towards bridging those gaps between a well-trained teacher network and the student network. Since we employ the same loss function in the current task, we can compare the losses of two knowledge transferring paths with the normal loss of forwarding a data sample batch through the student network. As illustrated in Fig. 4, Loss_1 and Loss_2 are the losses for passing through either staring from the teacher's sub-network or the student's sub-network, while Loss_3 is the loss of a normal forward for supervised training the student network. Let us denote the three losses as L_1, L_2 and L_3 respectively. L_1, L_2 and L_3 are all computed by a same loss function, such as Cross-Entropy loss if the student network is for classification. Since L_1, L_2 and L_3 are comparable, we can simply evaluate the values of these losses and then quantitatively find out the gap between the two networks.

Generally, if both L_1 and L_2 are constantly larger than L_3, the student network itself may underfit the current task when training with the sum of the three losses. That is, the learning task is dominated by the knowledge transfer loss. To avoid the risk, we just need to multiply a balancing parameter to L_1 and L_2. Fortunately, the balancing parameter is quite straightforward to determine in our formulations.

As our ultimate goal is to obtain a good student network, the main loss is the supervised task loss, i.e., L_3. Thus, we propose to adjust the scale of L_1 and L_2 according to the scale of L_3, to ensure the student network itself properly fitting the data of current task. More precisely, for each training mini-batch, after three losses are computed, a balancing parameter α is calculated by

$$\alpha = \frac{2 * L_3}{L_1 + L_2}. \tag{1}$$

Then the final loss for this mini-batch is $L = \alpha(L_1 + L_2) + L_3$. By multiplying the weight on the loss values, we are in fact balancing the gradients introduced by each loss to back-propagate through the student network, that is, if W is the weights of the student network, then we have

$$\frac{\partial L}{\partial W} = \alpha \left(\frac{\partial L_1}{\partial W} + \frac{\partial L_2}{\partial W} \right) + \frac{\partial L_3}{\partial W}, \tag{2}$$

where W means the parameters of the network.

Table 1: Results of model compression on ImageNet. Only the model gap exists between the teacher network and the student network.

Network	Method	Top-1	Top-5
ResNet-50	Teacher	76.16	92.86
MobileNet-v1	Baseline	68.87	88.76
	KD (Hinton, Vinyals, and Dean 2015)	68.58	88.98
	AT (Zagoruyko and Komodakis 2017)	69.56	89.33
	FT (Kim, Park, and Kwak 2018)	69.88	89.5
	AB (Heo et al. 2019b)	68.89	88.71
	Ours	**71.39**	**90.47**

Experiments

To validate the effectiveness of our proposed method, we conduct experiments on four different settings, as introduced in the introduction of the paper (Fig. 1).

Challenge: Only Model Gap

We begin our experiments with the gap between the teacher and the student: the model gap. Such a setting is also known as the distillation based model compression. In the experiments, a teacher network that is well-trained on ImageNet (Deng et al. 2009) will be used to guide a shallower student network. To test how the model gap affects the transfer process, we select the teacher network to be ResNet-50 (He et al. 2016) and the student network to be MobileNet-v1 (Howard et al. 2017), which is also tested in (Heo et al. 2019a). The model gap between ResNet-50 and MobileNet-v1 is in two aspects: First is the depth of the two networks, i.e., 50 layers for ResNet and 28 layers for MobileNet-v1; The second is the convolution kernel types, i.e., the normal convolution and the depth-wise convolution.

Comparison with SOTA. Since there is only the model gap, during training, we simply add the supervised task loss with the loss from the hybrid forward path. For training hyper-parameters, we use the same parameters as (Heo et al. 2019a): Batch size is set to 256; Learning rate is initialized with 0.1 and decay by 0.1 every 30 epochs. Also, for a fair comparison, the teacher network adopts the ResNet-50 trained by Torchvision. The final model compression results are shown in Tab. 1. We can see that our method outperforms 4 very recent methods.

Curves of training loss and training accuracy. Apart from the final results of the student network, we demonstrate the curves of training loss and training accuracy in Fig. 5.

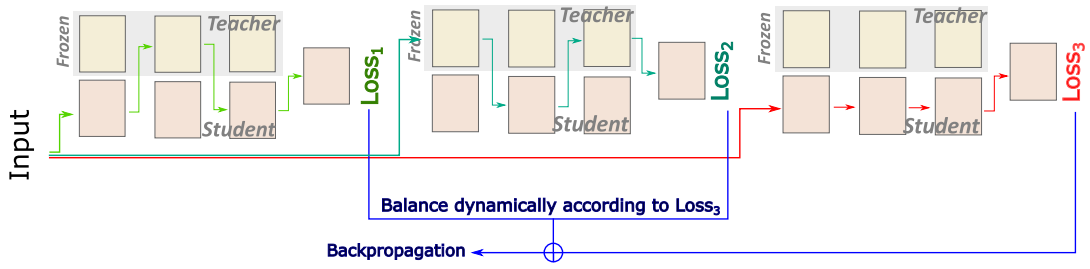

Figure 4: Losses computed from 3 different forwarding paths are balanced in a dynamic manner. $Loss_1$ and $Loss_2$ are the loss of knowledge transfer, passing a mini-batch starting with the sub-network of student (teacher) network then to the teacher (student) network. $Loss_3$ is the conventional training loss of the student network in the current task.

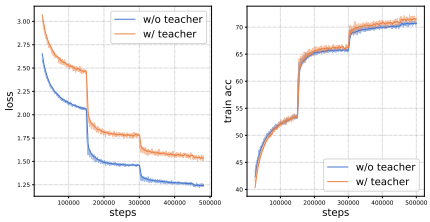

Figure 5: Training loss and training accuracy with and without the teacher network on ImageNet with ResNet-50 as the teacher and MobileNet-v1 as the student.

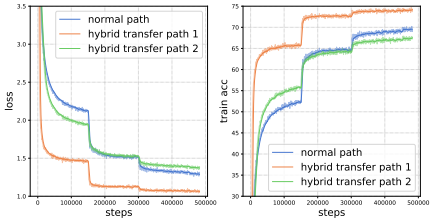

Figure 6: Training loss and training accuracy of the three forward paths on ImageNet with ResNet-50 as the teacher and MobileNet-v1 as the student.

We can see that the training loss with the teacher is always higher than the loss without the teacher, but the training accuracy with the teacher becomes higher than those without the teacher after the first time learning rate decay. The higher training loss is in accordance with our designed knowledge transferring rule, in which the training loss is the sum of the loss from the items L_1, L_2, and L_3.

Another thing we find interesting is the training loss and training accuracy of the three forward paths (in Fig. 4) shown in Fig. 6. The first thing we observe is that the curve of the normal path, i.e., the student network, is in the middle of the hybrid transfer paths. We check the detailed configs of the three paths and find that hybrid path 1 consists of 46 layers and hybrid path 2 consists of 32 layers. Recall that the normal training path of MobileNet v1, consists of 28 layers, the number of layers among the paths demonstrates that a deeper depth does not always have a lower training loss. The type of convolution is also an important factor in our case since the hybrid path involves the computation of a hybrid of depth-wise convolution and normal convolution. Another interesting observation is that before the first decay of the learning rate, the training loss of the hybrid paths is lower than the normal path, and the training accuracy is higher. This phenomenon shows that the knowledge embedded in the teacher network is helpful. Moreover, it indicates that forwarding with some sub-networks from the teacher network is able to utilize the pre-trained knowledge in the teacher network.

Table 2: Results of the student network on CityScapes with different teachers. The CS det teacher means the Faster RCNN network trained on the CityScapes dataset. The GTA5 seg teacher means the DeepLab v2 network trained on the GTA5 dataset.

Teacher	Transfer Method	mIoU
No Teacher	-	59.7
CS det (task gap)	FitNet (Romero et al. 2015)	59.9
	AT (Zagoruyko and Komodakis 2017)	60.0
	Ours	**60.2**
GTA5 seg (domain gap)	FitNet (Romero et al. 2015)	60.0
	AT (Zagoruyko and Komodakis 2017)	60.4
	Ours	**60.7**

Challenge: Model + Domain/Task Gap

Now we consider the other two gaps: the domain gap and the task gap. For the domain gap, the teacher network is trained on the same task with the same labels, but on a different dataset. For the task gap, the teacher network is trained on the same dataset but for a different task. As mentioned before, some knowledge of the teacher network may be be not relevant to the current task, therefore we need to balance the 3 training losses. For experimental settings, we follow the setup in recent progress on domain adaptation (Tsai et al. 2018; Song et al. 2020b,a). It is worth noting that in unsupervised domain adaptation the target domain, which is the dataset for our student network, is unlabeled. Since we hope to study how the teacher network from another domain is going to help, we assume the target domain is labeled.

Table 3: The results of the student network on GTA5 with different teachers. The performance of the student on GTA5 improves more when using the teacher from GTA5.

Model	mIoU
DeepLab v2 (Teacher)	65.1
w/o Teacher	33.5
w/ CS det	33.6
w/ GTA5 seg	34.2

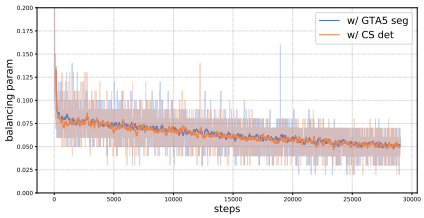

Figure 7: The dynamically computed balancing parameter of the hybrid loss and the training loss of the student network.

Datasets. Following (Tsai et al. 2018), the datasets used in this section are GTA5 (Richter et al. 2016) and CityScapes (Cordts et al. 2016). The GTA5 dataset has 24966 images and we randomly select 500 images out as the validation set for training the teacher network. Apart from the above, to better investigate on which gap is more challenging, we employ a multi-task setting on the CityScapes dataset. We first train a teacher network on CityScapes for detection and then use the detection teacher network for helping the segmentation task, which is the new task for the student network.

Network and training details. For the teacher networks, we choose the DeepLab v2 (Chen et al. 2017) with ResNet-101 as the backbone for domain adaptation, and the Faster RCNN (Ren et al. 2015) with ResNet-50 as the backbone for multi-task testing. Besides, DeepLab v2 with VGG 16 as backbone is chosen to be the student network. In this way, when running the transfer process with our method, we are also dealing with the model gap at the same time. When training the student network with the above two teachers, we use the same training hyper-parameters: batch size of 8, 40000 iterations and learning rate starting from 1e-3 with polynomial decay.

The student network performance. Since we are interested in whether the knowledge from the teacher can help the student, we first present results on the CityScapes validation set with the above two teachers. For comparison, we report the results with two intermediate feature alignment based knowledge transfer methods: FitNet (Romero et al. 2015) and AT (Zagoruyko and Komodakis 2017). To get a fair comparison, we tune the balancing parameter of the mimic loss for the above two methods under our settings. The results are presented in Tab. 2, from which we can come to the conclusion that the teacher trained on GTA5 is better than the teacher trained on CityScapes detection. We think the reason why the teacher from GTA5 is better is that GTA5 is a much larger dataset compared to CityScapes.

Performance on teacher's domain. When we use the teacher from another dataset, which is in fact another domain, the student network may also benefit from the teacher's knowledge about the other domain. Therefore, we also test the performance of the student on the other domain, which is the GTA5 dataset. In Tab. 3, we show the results on the GTA5 validation set, which is split previously to acquire a teacher network on the GTA5. The student taught by the teacher from GTA5 performs better than the one without the teacher. This result implies that by our hybrid forward scheme, the student is able to learn domain specific knowledge, even without seeing any data from the other domain.

Balancing parameter along with training. The balancing parameter reflects the ratio of the loss from the normal training of the student network and the loss from the the two hybrid forward paths. That is, the balancing parameter is able to reflect how much the teacher is helping the student. In Fig. 7, we draw the value of balancing parameters with respect to the training steps. We can observe: 1) The balancing parameter becomes smaller along with the training, which means that the student network learns more from the teacher network. 2) In our setting, it is hard to tell which teacher is more helpful, since the balancing parameters of the two teachers are about the same. However, the initial value of the detection teacher is much larger than the GTA5 teacher, maybe because they are trained on the same dataset and less noisy at the beginning.

Challenge: Model + Domain + Task Gap

Now, we move to the settings with all the gaps, i.e., the general knowledge transfer setting shown in Fig. 1. The teacher network and the student network will be trained on different datasets and for different tasks. The teacher network and the student network will are still semantically relevant. In our experiments, the teacher network is a human pose estimation while the student network is a pedestrian detection network.

Datasets and networks. For human pose estimation, we directly use the well-trained network from (Kreiss, Bertoni, and Alahi 2019) which is public available. The teacher network uses ResNet-50 as backbone. For the student network, we use Faster RCNN with MobileNet v1 as backbone. The dataset for the student network is CityPersons (Zhang, Benenson, and Schiele 2017), which uses the images from CityScapes and the pedestrian are manually relabeled. Again, all the training hyper-parameters are kept the same when training with and without the teacher.

Table 4: Pedestrian detection performance of the student network on CityPersons validation set. The metric is miss-rate and a smaller value is better. R means reasonable and performance on the R subset is usually more important (Zhang, Benenson, and Schiele 2017). The teacher is a human pose estimation network, so we are dealing with all the three gaps.

Model	R	R small	R occ heavy	All
w/o Teacher	17.20	21.87	61.76	47.09
FitNet (Romero et al. 2015)	16.70	22.20	61.13	46.89
AT (Zagoruyko and Komodakis 2017)	16.69	21.42	65.50	46.86
Ours	**16.19**	21.12	64.11	46.85

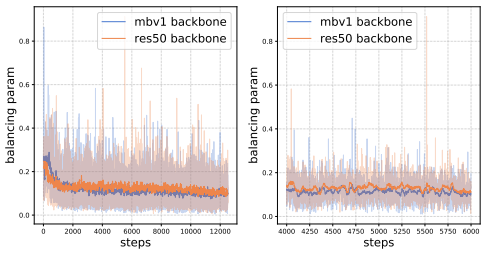

Figure 8: The dynamically computed balancing parameter of two student networks. The only difference between the two student network is their backbone networks, ResNet-50 and MobileNet-v1.

Results of knowledge transfer. Firstly, we evaluate how much the teacher network will help the student network. The results on CityPersons validation set are presented in Tab. 4. Comparison methods are chosen with the same settings as in Tab. 2. From the table, we can see that with the help from a human pose estimation teacher, most of the metrics are improved. Among all the metrics, the improvements on the reasonable subset is the most significant, which is usually more practical important. For the heavily occluded subset, we presume that it is caused by the knowledge passed by the teacher, which is a pose estimation network and unable to handle the heavily occluded pedestrian.

How the model gap affects knowledge transfer? In our previous analysis, we claim that the model gap is the easiest gap among the three gaps. The intuition is that the model gap will only affect how the knowledge can be passed to the student. However, when we face all the three gaps, we can no longer analyze the three gaps separately. So the model gap can be also hard to deal with as the other two gaps when all the there gaps present. To answer the question, we train another Faster RCNN network on CityPersons with the same teacher, but the backbone is ResNet-50, which is the same as the teacher. Therefore, by looking into the balancing parameter, we can have a sense of whether the model gap is a serious issue or not. In Fig. 8, the curves of the dynamically computed balancing parameter concerning two different backbone settings are drawn. The first conclusion we can

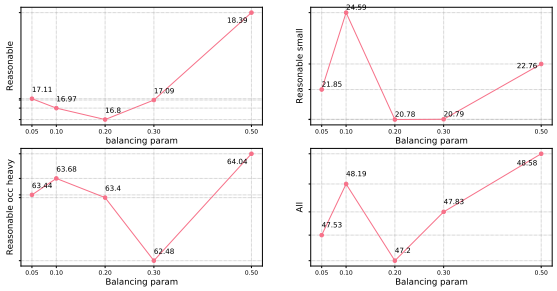

Figure 9: Results on general knowledge transfer with different fixed balancing parameters.

have is that the model gap is indeed not a serious issue. The reason is similar to our previous analysis: The balancing parameter reflects how knowledge transfer loss is contributing to the current task. Another interesting result from the curves is that the model gap has some impact in the middle of the learning process. We zoom in some middle steps (4000 - 6000) of the training and get the curves on the right. We can see that generally the ResNet-50 based student has a larger balancing parameter than the MobileNet v1 based student. This indicates that the knowledge from the teacher is more helpful if the model gap is smaller.

Comparison with fixed balancing parameter. One of the main contributions in our method is that we adjust the balancing parameter in a dynamic manner. In Fig. 9, we manually tune the balancing parameter for the hybrid forward loss and the task loss, with the same teacher-student setting as in Tab. 4. From the figure, we can see that if we manually tune the balancing parameter, the best result does not achieve the accuracy of our dynamic balancing strategy. However, after carefully tuning the balancing parameter, the pedestrian detection network works better than the one without the teacher, thereby verifying the effectiveness of our hybrid forward scheme.

Conclusion

In this paper, to achieve the goal of robust knowledge transfer, we propose a hybrid forward scheme to pass the knowledge from a teacher network to a student network. Since employing the task specific loss functions to measure the losses of hybrid forward paths, we design a dynamic balancing strategy to enable the knowledge being transferred more robust. To validate that our proposed hybrid forward scheme can deal with different gaps between the teacher and student, we first test our method on the conventional model compression setting, which means only the model gap exists. Next, we study the settings that two gaps are presented. Finally, we use a human pose estimation network as the teacher and use a pedestrian detection network as the student, which deals with three gaps between the teacher-student pair. All of the experiments demonstrate that our method effectively enables knowledge transfer from the teacher to the student, despite the gaps between them.

Acknowledgement

This work is supported in part by a gift grant from Horizon Robotics and National Science Foundation Grant CNS-1951952. We thank the four anonymous reviewers for their constructive comments and thank Jiemin Fang, Helong Zhou and Yuzhu Sun for the discussion and assistance.

References

Bender, G.; Kindermans, P.-J.; Zoph, B.; Vasudevan, V.; and Le, Q. 2018. Understanding and Simplifying One-Shot Architecture Search. In Dy, J.; and Krause, A., eds., *International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 550–559. Stockholmsmässan, Stockholm Sweden: PMLR.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(4): 834–848.

Chen, T.; Goodfellow, I. J.; and Shlens, J. 2016. Net2Net: Accelerating Learning via Knowledge Transfer. In *International Conference on Learning Representations*.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3213–3223.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Fang, J.; Sun, Y.; Peng, K.; Zhang, Q.; Li, Y.; Liu, W.; and Wang, X. 2019. Fast Neural Network Adaptation via Parameter Remapping and Architecture Search. In *International Conference on Learning Representations*.

Fang, J.; Sun, Y.; Zhang, Q.; Peng, K.; Li, Y.; Liu, W.; and Wang, X. 2020. FNA++: Fast Network Adaptation via Parameter Remapping and Architecture Search. *arXiv preprint arXiv:2006.12986* .

Gao, Y.; Ma, J.; Zhao, M.; Liu, W.; and Yuille, A. L. 2019. NDDR-CNN: Layerwise Feature Fusing in Multi-Task CNNs by Neural Discriminative Dimensionality Reduction. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Gong, X.; Chen, S.; Zhang, B.; and Doermann, D. 2021. Style Consistent Image Generation for Nuclei Instance Segmentation. In *Winter Conference on Applications of Computer Vision*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Heo, B.; Kim, J.; Yun, S.; Park, H.; Kwak, N.; and Choi, J. Y. 2019a. A Comprehensive Overhaul of Feature Distillation. In *IEEE International Conference on Computer Vision*.

Heo, B.; Lee, M.; Yun, S.; and Choi, J. Y. 2019b. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3779–3787.

Hinton, G. E.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* .

Kim, J.; Park, S.; and Kwak, N. 2018. Paraphrasing complex network: Network compression via factor transfer. In *Advances in Neural Information Processing Systems*, 2760–2769.

Kreiss, S.; Bertoni, L.; and Alahi, A. 2019. PifPaf: Composite Fields for Human Pose Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 11977–11986.

Kundu, J. N.; Lakkakula, N.; and Babu, R. V. 2019. UM-Adapt: Unsupervised Multi-Task Adaptation Using Adversarial Cross-Task Distillation. In *IEEE International Conference on Computer Vision*.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2117–2125.

Peng, B.; Jin, X.; Liu, J.; Li, D.; Wu, Y.; Liu, Y.; Zhou, S.; and Zhang, Z. 2019. Correlation Congruence for Knowledge Distillation. In *IEEE International Conference on Computer Vision*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 91–99.

Richter, S. R.; Vineet, V.; Roth, S.; and Koltun, V. 2016. Playing for data: Ground truth from computer games. In *European conference on computer vision*, 102–118. Springer.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2015. FitNets: Hints for Thin Deep Nets. In *International Conference on Learning Representations*.

Shen, C.; Xue, M.; Wang, X.; Song, J.; Sun, L.; and Song, M. 2019. Customizing Student Networks From Heterogeneous Teachers via Adaptive Knowledge Amalgamation. In *IEEE International Conference on Computer Vision*, 3504–3513.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .

Song, L.; Wang, C.; Zhang, L.; Du, B.; Zhang, Q.; Huang, C.; and Wang, X. 2020a. Unsupervised domain adaptive re-identification: Theory and practice. *Pattern Recognition* 102: 107173.

Song, L.; Xu, Y.; Zhang, L.; Du, B.; Zhang, Q.; and Wang, X. 2020b. Learning from Synthetic Images via Active Pseudo-Labeling. *IEEE Transactions on Image Processing* .

Srinivas, S.; and Fleuret, F. 2018. Knowledge Transfer with Jacobian Matching. In *International Conference on Machine Learning*, 4730–4738.

Tsai, Y.-H.; Hung, W.-C.; Schulter, S.; Sohn, K.; Yang, M.-H.; and Chandraker, M. 2018. Learning to Adapt Structured Output Space for Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Tung, F.; and Mori, G. 2019. Similarity-Preserving Knowledge Distillation. In *IEEE International Conference on Computer Vision*.

Wei, T.; Wang, C.; and Chen, C. W. 2019. Stable Network Morphism. In *International Joint Conference on Neural Networks*, 1–8.

Wei, T.; Wang, C.; Rui, Y.; and Chen, C. W. 2016. Network Morphism. In *International Conference on Machine Learning*, 564–572.

Wu, J.; Song, L.; Wang, T.; Zhang, Q.; and Yuan, J. 2020a. Forest R-CNN: Large-vocabulary long-tailed object detection and instance segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1570–1578.

Wu, J.; Zhou, C.; Yang, M.; Zhang, Q.; Li, Y.; and Yuan, J. 2020b. Temporal-Context Enhanced Detection of Heavily Occluded Pedestrians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13430–13439.

Wu, J.; Zhou, C.; Zhang, Q.; Yang, M.; and Yuan, J. 2020c. Self-mimic learning for small-scale pedestrian detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2012–2020.

Xu, C.; Zhou, W.; Ge, T.; Wei, F.; and Zhou, M. 2020. BERT-of-Theseus: Compressing BERT by Progressive Module Replacing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7859–7869. Association for Computational Linguistics.

Ye, J.; Ji, Y.; Wang, X.; Ou, K.; Tao, D.; and Song, M. 2019. Student Becoming the Master: Knowledge Amalgamation for Joint Scene Parsing, Depth Estimation, and More. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2829–2838.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning. *IEEE Conference on Computer Vision and Pattern Recognition* 7130–7138.

Yu, J.; and Huang, T. S. 2019. Universally Slimmable Networks and Improved Training Techniques. In *IEEE International Conference on Computer Vision*.

Yu, J.; Yang, L.; Xu, N.; Yang, J.; and Huang, T. 2019. Slimmable Neural Networks. In *International Conference on Learning Representations*.

Zagoruyko, S.; and Komodakis, N. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *International Conference on Learning Representations*.

Zhang, S.; Benenson, R.; and Schiele, B. 2017. CityPersons: A Diverse Dataset for Pedestrian Detection. In *IEEE Conference on Computer Vision and Pattern Recognition*.