Knowledge Distillation via the Target-aware Transformer

Sihao Lin[1,3†‡], Hongwei Xie[2†], Bing Wang[2], Kaicheng Yu[2],
Xiaojun Chang[3§], Xiaodan Liang[4], Gang Wang[2]
[1]RMIT University [2]Alibaba Group [3]ReLER, AAII, UTS [4]Sun Yat-sen University
{linsihao6, hongwei.xie.90, Kaicheng.yu.yt, xdliang328}@gmail.com
{fengquan.wb, wg134231}@alibaba-inc.com, xiaojun.chang@uts.edu.au

Abstract

Knowledge distillation becomes a de facto standard to improve the performance of small neural networks. Most of the previous works propose to regress the representational features from the teacher to the student in a one-to-one spatial matching fashion. However, people tend to overlook the fact that, due to the architecture differences, the semantic information on the same spatial location usually vary. This greatly undermines the underlying assumption of the one-to-one distillation approach. To this end, we propose a novel one-to-all spatial matching knowledge distillation approach. Specifically, we allow each pixel of the teacher feature to be distilled to all spatial locations of the student features given its similarity, which is generated from a target-aware transformer. Our approach surpasses the state-of-the-art methods by a significant margin on various computer vision benchmarks, such as ImageNet, Pascal VOC and COCOStuff10k. Code will be released soon.

1. Introduction

Knowledge distillation [19, 31] refers to a simple technique to improve the performance of any machine learning algorithms. One common scenario is to distill the knowledge from a larger teacher neural network to a smaller student one, such that the performance of student model can be significantly boosted comparing to training the student model alone. Concretely, people formulate an external loss function that guides the student feature map to mimic teacher's. Recently, it has been applied to various downstream applications, such as model compression [42, 48], continual learning [25], and semi-supervised learning [8].

Earlier works only distill the knowledge from the final layer of neural networks, for example, the "logits" in image classification task [1, 19]. Recently, people discover that distilling the intermediate feature maps is a more effective approach to boost the student's performance. This line of works encourage similar patterns to be elicited in the spatial dimensions [36, 50], and is constituted as state-of-the-art knowledge distillation approach [7, 22].

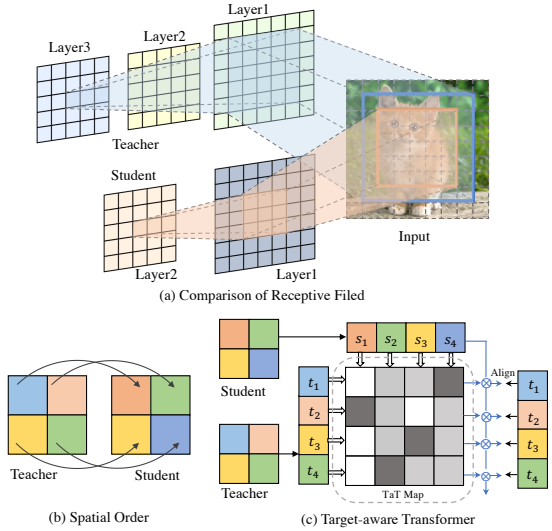


Figure 1. **Illustration of semantic mismatch.** Suppose that teacher and student are the 3-layers and 2-layers convnets with kernel size 3×3 and stride 1×1. (a) shows the receptive field of the middle pixel of the final feature map, where the blue box represents the teacher's receptive field and the orange box is that of the student's. Since teacher model has more convolutional operations, the resulting teacher feature map has a larger receptive field and thus contains richer semantic information. (b) Hence, directly regressing the student's and teacher's feature in a one-to-one spatial matching fashion may be suboptimal. (c) We proposed a one-to-all knowledge distillation via a target-aware transformer that can let the teacher's spatial components be distilled to the entire student feature maps.

To compute the distillation loss of the aforementioned approach, one need to select the source feature map from the teacher and the target feature map from the student, where these two feature maps must have the same spatial dimension. As shown in Figure 1 (b), the loss is computed in a one-to-one spatial matching fashion, that is formulated

§Corresponding Author.
†Equal contribution.
‡Part of the work done when as an intern in DAMO Academy.

as a summation of the distance between the source and the target features at each spatial location. One underlying assumption of this approach is the spatial information of each pixel is the same. In practice, this assumption is commonly not valid due to the fact that student model usually has fewer convolutional layers than the teacher. One example is shown in Figure 1 (a), even at the same spatial location, the receptive field of student feature is often significantly smaller than the teacher's and thus contains less semantic information. In addition, recent works [5, 10, 41, 49] evidences the importance of receptive field's influence on the model representation power. Such discrepancy is a potential reason that the current one-to-one matching distillation leads to sub-optimal results.

To this end, we propose a novel one-to-all spatial matching knowledge distillation approach. In Figure 1 (c), our method distills the teacher's features at each spatial location into all components of the student features through a parametric correlation, *i.e*., the distillation loss is a weighted summation of all student components. To model such correlation, we formulate a transformer structure that reconstructs the corresponding individual component of the student features and produces an alignment with the target teacher feature. We dubbed this target-aware transformer. As such, we use parametric correlations to measure the semantic distance conditioned on the representational components of student feature and teacher feature to control the intensity of feature aggregation, which address the downside of one-to-one matching knowledge distillation.

As our method computes the correlation between feature spatial locations, it might become intractable when feature maps are large. To this end, we extend our pipeline in a two-step hierarchical fashion: 1) instead of computing correlation of all spatial locations, we split the feature maps into several groups of patches, then performs the one-to-all distillation within each group; 2) we further average the features within a patch into a single vector to distill the knowledge. This reduces the complexity of our approach by order of magnitudes.

We evaluate the effectiveness of our method on two popular computer vision tasks, image classification and semantic segmentation. On the ImageNet classification dataset, the tiny ResNet18 student can be boosted from 70.04% to 72.41% in terms of the top-1 accuracy, and surpasses the state-of-the-art knowledge distillation by 0.8%. As for the segmentation task on COCOStuff10k, comparing to the previous approaches, our approach is able to boost the compact MobilenetV2 architecture by 1.75% in terms of the mean intersection of union (mIoU).

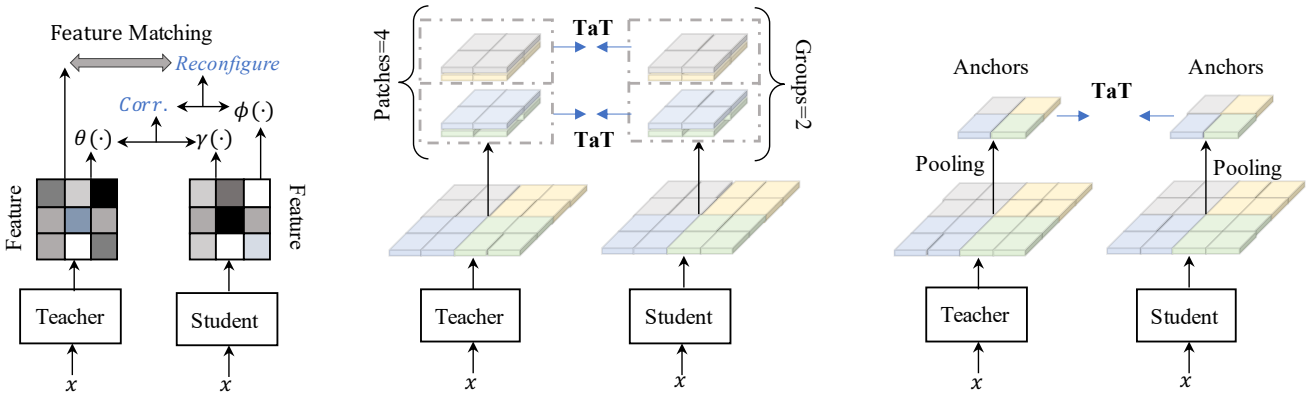
Our contributions can be summarized as follows:

- We propose the knowledge distillation via a target-aware transformer, which enables the whole student to mimic each spatial component of the teacher respectively. In this way, we can increase the matching capability and subsequently improve the knowledge distillation performance.
- We propose the hierarchical distillation to transfer local features along with global dependency instead of the original feature maps. This allows us to apply the proposed method to applications, which are suffered from heavy computational burden because of the large size of feature maps.
- We achieve state-of-the-art performance compared against related alternatives on multiple computer vision tasks by applying our distillation framework.

2. Related Works

The seminal work [19] introduced the idea of knowledge distillation. Specifically, Hinton *et al*. proposed to distill the logits (before sotfmax layer) from teacher to student by minimizing the KL divergence, where a temperature factor is applied to soften the logits. Since feature map contains richer representation, Romero *et al*. [36] introduced the intermediate layer transfer between teacher and student. Lately, AT [50] proposed several statistical methods to highlight the dominating area of the feature map and discarded low-response area as noise. Chen *et al*. [3] proposed the semantic calibration which allowed the student to learn from the most semantic-related teacher layer. In [22], the feature similarities between teacher and student were calculated and then were used as weights to balance the feature matching. These early methods intuitively established the links between knowledge source (teacher) and distillation terminal (student) in the one-to-one manner by spatial order.

However, they overestimated the prior of spatial order while neglected the issues of semantic mismatch, *i.e*., the pixels of teacher feature map often contains richer semantic compared to that of student on the same spatial location. We found that some works [20, 27, 33–35, 43, 48], though unintended, have been proposed to relax the spatial constrain during feature transfer. Typically, they defined the relational graph, and similarity matrix in the feature space of teacher network and transferred it to the student network. For instances, Tung and Mori [43] calculated the similarity matrix where each entry encoded the similarity between two instances. Liu *et al*. [27] measured the correlation between channels by inner-product. They condensed and compressed the entire feature to some properties (often scalar) and thus collapsed the spatial information. On the other hand, such process damaged the original teacher feature and may lead to sub-optimal solution. The spread of KD has also driven some methods designed for specific vision tasks including video captioning [32], action recognition [9, 46], object detection [4, 11, 51] and semantic segmentation [17, 28, 47]. Regarding the semantic segmenta-



(a) **T**arget-**a**ware **T**ransformer (b) Patch-group Distillation (c) Anchor-point Distillation

Figure 2. Illustration of our framework. (a) **Target-aware Transformer**. Conditioned on the teacher feature and the student feature, the transformation map *Corr.* is computed and then applied on the student feature to reconfigure itself, which is then asked to minimize the L_2 loss with the corresponding teacher feature. (b) **Patch-group Distillation**. Both teacher and student features are to be sliced and rearranged as groups for distillation. By concatenating the patches within a group, we explicitly introduce the spatial correlation among the patches beyond the patches themselves. (c) **Anchor-point Distillation**. Each color indicates a region. We use average pooling to extract the *anchor* within a local area of the given feature map, forming the new feature map of a smaller size. The generated anchor-point features will participate in the distillation.

tion, these methods are indeed related to relation knowledge distillation which computes similarity matrix [43]. To investigate the potential of our method, we also adapt the method to semantic segmentation with hierarchical distillation.

The success of Transformer [44] in NLP has attracted lots of attention from the community of computer vision [13, 24, 29, 45]. While the original ViT [13] suffered from computation burden, Liu *et al*. [29] proposed the shifted-window that computes the attention on patch-level. The Pyramid ViT [45] proposed a progressive shrinking pyramid that adjusts the scale of feature map.

3. Method

In this section, we first briefly describe the fundamental elements of feature map knowledge distillation and then introduce the general formulation of our knowledge distillation via a target-aware transformer. As our method computes the point-wise correlation of the given feature maps, the computational complexity becomes intractable on large-scale features, we then introduce the hierarchical distillation approach to address this limitation.

3.1. Formulation

Suppose the teacher and the student are two convolutional neural networks, denoted by T and S. $F^T \in \mathbb{R}^{H\times W\times C}$ and $F^S \in \mathbb{R}^{H\times W\times C'}$ denote the teacher feature and student feature respectively, where H and W are the height and width of the feature map, and C represents the channel numbers. In the pioneer work [19], the distillation loss is formulated by a distance of features that come from the last layer of the networks. For example, in the image classification domain, it refers to the "logits" before going in the softmax layer and cross-entropy loss. Concretely, the vanilla distillation loss is defined as:

$$\mathcal{L}_{\mathrm{KL}} = \mathrm{KLD}(\sigma(\frac{T(x)}{\tau}), \sigma(\frac{S(x)}{\tau})), \tag{1}$$

where $\mathrm{KLD}(\cdot)$ measures the Kullback-Leibler divergence, $\sigma(\cdot)$ is the softmax function, $T(x)$ and $S(x)$ are the output logits given specific input x, and τ is the temperature factor. Without loss of generality, we assume that $C^{'}$ aligns with C and reshape both F^T and F^S into 2D matrices:

$$\begin{aligned} f^s &= \Gamma(F^S) \in \mathbb{R}^{N\times C}, \\ f^t &= \Gamma(F^T) \in \mathbb{R}^{N\times C}. \end{aligned} \tag{2}$$

Here $\Gamma(\cdot)$ is a function that flattens the 3D feature tensor into the 2D matrix where each row of the matrix is associated with a pixel in the feature tensor by spatial order and $N = H \times W$. We can describe f^s and f^t as two sets of the pixels with cardinality N :

$$\begin{aligned} {f^s}^\top &= [f_1^s, f_2^s, f_3^s, \ldots, f_N^s], \\ {f^t}^\top &= [f_1^t, f_2^t, f_3^t, \ldots, f_N^t]. \end{aligned} \tag{3}$$

Previous work [36] simply minimize the discrepancy between two sets f^s and f^t in a one-to-one spatial matching manner, we denote this approach feature matching (FM):

$$\mathcal{L}_{\mathrm{FM}} = ||F^S - F^T||_2 = \sum_{i=1}^{N} ||f_i^s - f_i^t||_2. \tag{4}$$

This formulation assumes that the semantic distributions of the teacher and the student match exactly. However, as mentioned earlier, for the feature maps of the teacher network, which usually encompasses more layers and larger feature channels, the spatial information of the same pixel location contains a richer semantic information compare to the student network. Directly regressing the features in a pixel-wise manner may lead to suboptimal distillation results.

To this end, we propose a one-to-all spatial matching knowledge distillation pipeline that allows the each feature location of the teacher to teach the entire student features in a dynamic manner. To make the whole student mimic a spatial component of the teacher, we propose the **T**arget-**a**ware **T**ransformer (**TaT**) to pixel-wisely reconfigure the semantic of student feature in the certain position. Given a spatial component (alignment target) of the teacher, we use **TaT** to guide the whole student to reconstruct the feature in its corresponding location. Conditioned on the alignment target, **TaT** should reflect the semantic similarity with the components of the student feature. We use a linear operator to avoid changing the distribution of student semantics. The formulation of transformation operator W^i can be defined as:

$$\begin{aligned} W^i &= \sigma(\langle f_1^s, f_i^t \rangle, \langle f_2^s, f_i^t \rangle, \ldots, \langle f_N^s, f_i^t \rangle) \\ &= [w_1^i, w_2^i, \ldots, w_N^i], \end{aligned} \tag{5}$$

where f_i^t and f_i^s denote the corresponding i-th components of teacher and student, $\langle \cdot, \cdot \rangle$ represents the inner-product and $\|W^i\| = 1$. We use inner-product to measure the semantic distance and softmax function for normalization. Each entry of W^i is like the gate and controls the amount of semantic that will be propagated to the i-th reconfigured point. By aggregating the these related semantic across all the components, we have the result:

$$f_i^{s'} = w_1^i \times f_1^s + w_2^i \times f_2^s + \cdots + w_N^i \times f_N^s. \tag{6}$$

The Eq. 5 and Eq. 6 can be combined and rewritten as the form of matrix multiplication: $f_i^{s'} = \sigma(f^s \cdot f_i^t) \cdot f^s$.

Note this is the simple non-parametric method that only depends on the original features. To facilitate the training, we introduce the parametric method with the extra linear transformation applied on the student feature and teacher feature. We observe that parametric version performs better than non-parametric one in ablation study. Guided by the target-aware transformer, the reconfigured student feature can be formulated as:

$$f^{s'} = \sigma(\gamma(f^s) \cdot \theta(f^t)^\top) \cdot \phi(f^s), \tag{7}$$

where $\theta(\cdot)$, $\gamma(\cdot)$ and $\phi(\cdot)$ are the linear functions consisting of 3×3 conv layer plus the BN layer [21]. We compare the parametric **TaT** to non-parametric one to analyse the effectiveness brought by these linear functions in the Section 4.5. In the case that the channel numbers of F^S do not match with that of F^T, $\gamma(\cdot)$ can help with alignment.

After reconfiguration, each component of $f^{s'}$ aggregates the meaningful semantic from the original feature, which enhances the expressivity. We do not require the student to reconstruct the teacher feature in a pixel-to-pixel manner. Indeed, our model allows the student to act as a whole to mimic the teacher. The resulting $f^{s'}$ is lately asked to minimize the L_2 loss with the teacher feature. The objective for **TaT** knowledge distillation can be given by:

$$\mathcal{L}_{\mathrm{TaT}} = ||f^{s'} - f^t||_2. \tag{8}$$

Finally, the total loss of our proposed method can be defined by:

$$\mathcal{L} = \alpha \mathcal{L}_{\mathrm{Task}} + \beta \mathcal{L}_{\mathrm{KL}} + \epsilon \mathcal{L}_{\mathrm{TaT}}, \tag{9}$$

Here $\mathcal{L}_{\mathrm{Task}}$ can be any loss on the generic machine learning tasks. α, β and ϵ are the weight factors to balance the loss. Empirically, we find that our model benefits from $\mathcal{L}_{\mathrm{KL}}$. However, the model can achieve state-of-the-art without the help of $\mathcal{L}_{\mathrm{KL}}$.

3.2. Hierarchical Distillation

The proposed **TaT** lift the limitation of previous one-to-one spatial matching fashion. However, the computation complexity of **TaT** map will become intractable when it comes to a large feature map. Assuming the spatial dimensions of the feature map are H and W, this means the computation complexity will reach $\mathcal{O}(H^2 \cdot W^2)$. Therefore, we propose a hierarchical distillation approach to address this large feature map limitation. It contains two steps: 1) patch-group distillation that splits the entire feature maps into smaller patches, so to distill local information from the teacher to the student; 2) we further summarize the local patches into one vector and distill this for global information.

3.2.1 Patch-group Distillation

As mentioned above, as the spatial dimension of input feature maps increases, distillation becomes more difficult. A straightforward solution [27] is to divide the feature map into patches and perform distillation within patches individually. However, the correlation between patches is completely ignored, resulting in sub-optimal solutions.

In contrast to Liu *et al*. [27], we propose the patch-group distillation (See Figure 2 (b)) that allows the student to learn the local feature from patches and retain the correlation among them to some extent. Given the original student feature F^S and teacher feature F^T, they are partitioned into $n \times m$ patches of size $h \times w$, where $h = H/n$, $w = W/m$. They are further arranged as g groups sequentially where each group contains $p = n \cdot m/g$ patches. Specifically, the patches in a group will be concatenated channel-wisely,

forming a new tensor of size $h \times w \times c \cdot g$ that would be used for distillation lately. In this way, each pixel of the new tensor contains the features from p positions of the original feature, which explicitly includes the spatial pattern. Therefore, during the distillation, the student can learn not only the single pixel but the correlation among them. Intuitively, a larger group will introduce richer correlation but complex correlation will turn to difficult to learn. We study the effectiveness of different group sizes in the experiments.

Similar to the formulation presented in Section 3.1, the patch-group distillation can be given by simply replacing the original input with the reorganized one, and the variant is denoted as $\mathcal{L}^{\mathcal{P}}_{\mathrm{TaT}}$. To relax the strict constraints of the spatial pattern in the patch-group, we set the $\theta(\cdot)$ as linear transformation in our experiments.

3.2.2 Anchor-point Distillation

The patch-group distillation can learn the fined-grained feature on the patch level and retain the spatial correlation among the patches to some extent. However, it is not capable of perceiving the long-range dependency. As will see in the ablation study, the attempt to preserve the global correlation through concatenating all the patches would fail. For complex scenes, long-range dependency is important to capture the relation (*e.g.* layout) of different objects.

We address the conundrum by the proposed anchor-point distillation. As shown in Figure 2 (c), we summarize the local area to compact representation, referred to *anchor*, within a local area that is representative to describe the semantic of the given area, forming the new feature map of smaller size. Since the new feature map consists of the summary of the original feature, it can approximately substitute the original one to obtain the global dependency. We simply use average pooling to extract the anchor points. Then all the anchors are scattered back to the associated position to form a new feature map. The anchor-point feature is used for distillation as described in Section 3.1 and the objective is denoted as $\mathcal{L}^{\mathcal{A}}_{\mathrm{TaT}}$. The patch-group distillation enables the student to mimic the local feature while the anchor-point distillation allows the student to learn the global representation over the coarse anchor-point feature, which are complementary to each other. Therefore, the combination of these two objectives can bring the best of two worlds. Our objective designed for semantic segmentation can be written by:

$$\mathcal{L}_{\mathrm{Seg}} = \alpha\mathcal{L}_{\mathrm{CE}} + \delta\mathcal{L}^{\mathcal{P}}_{\mathrm{TaT}} + \zeta\mathcal{L}^{\mathcal{A}}_{\mathrm{TaT}} \tag{10}$$

4. Experiment

In this section, we empirically evaluate the effectiveness of the proposed method through extensive experiments. On image classification, we leverage the commonly used benchmark in knowledge distillation such as Cifar-100 [23] and ImageNet [12], and show our model can improve the student performance by a significant margin compared to many state-of-the-art baselines. In addition, we extend our method to another popular computer vision task, semantic segmentation to further demonstrate the generalization ability of our method. We nonetheless provide a detailed ablation study in the end of this section.

4.1. Datasets

Cifar-100 [23]. This benchmark contains 100 categories including 600 samples each. For each category, there are 500 images for training while 100 images for testing. We report top-1 accuracy as evaluation metric.

ImageNet [12]. This is a challenging benchmark for image classification including more than one million training samples with 1,000 categories. Similarly, we report the top-1 accuracy to measure the model performances.

Pascal VOC [14]. This benchmark contains 20 foreground classes with a background class. It provides 1,464 training, 1,499 validation, and 1,456 testing samples. Apart from the fine annotated samples, we also use additional coarse annotated images from [15] for training, resulting in 10,582 training samples. We report the mean Intersection over Union (mIoU) on the validation set to measure the proposed method.

COCOStuff10k [2]. The challenging dataset is developed on MSCOCO [26] by adding dense pixel-wise stuff label, resulting in 172 classes: 80 for thing, 91 for stuff, and 1 for unlabeled. It contains 9k training samples and 1k validation samples. We report the mIoU to evaluate our method.

4.2. Implementation Details

Image classification. For the experiments on Cifar-100, we use SGD optimizer [40] and the total running epoch is set to 240. The initial learning rate is 0.05 with a decay rate 0.1 at epoch 150, 180, and 210. In terms of data augmentation, the input images will be randomly cropped and flipped horizontally. We use Bayesian optimization [39] for hyper-parameters (*i.e.* α and ϵ in Eq. 9) searching. We report the exact values in the supplementary materials. For the ImageNet experiments, we use the AdamW optimizer [30] and train all of the models for 100 epochs with a batch size of 2048. The initial learning rate is set to 1.6e-4 and decays by 0.1 at epoch 30, 60, and 90. We apply standard data augmentations including random crop and horizontal flip. We use a simple grid search on the hyper-parameters, and set α=0.5, β=0.5 and ϵ=0.1 in Eq. 9.

Semantic segmentation. We choose the DeepLabV3+ [6] as the base architecture, where it contains a backbone to extract feature and a head to generate the segmentation results. For the teacher, we follow [6] to use the ResNet101

Table 1. **Top-1 accuracy(%) on Cifar-100.** The loss term $\mathcal{L}_{\mathrm{KL}}$ in Eq. 9 is removed in this experiment.

Method	Network Architecture						
	WRN-40-2 WRN-16-2	WRN-40-2 WRN-40-1	ResNet56 ResNet20	ResNet110 ResNet20	ResNet110 ResNet32	ResNet32×4 ResNet8×4	VGG13 VGG8
Teacher	75.61	75.61	72.34	74.31	74.31	79.42	74.64
Vanilla	73.26	71.98	69.06	69.06	71.14	72.50	70.36
KD [19]	74.92	73.54	70.66	70.67	73.08	73.33	72.98
FitNet [36]	73.58	72.24	69.21	68.99	71.06	73.50	71.02
AT [50]	74.08	72.77	70.55	70.22	72.31	73.44	71.43
SP [43]	73.83	72.43	69.67	70.04	72.69	72.94	72.68
CC [35]	73.56	72.21	69.63	69.48	71.48	72.97	70.71
RKD [33]	73.35	72.22	69.61	69.25	71.82	71.90	71.48
PKT [34]	74.54	73.45	70.34	70.25	72.61	73.64	72.88
FSP [48]	72.91	NA	69.95	70.11	71.89	72.62	70.20
NST [20]	73.68	72.24	69.60	69.53	71.96	73.30	71.53
CRD [42]	75.48	74.14	71.16	71.46	73.48	75.51	73.94
ICKD [27]	75.64	74.33	**71.76**	71.68	73.89	75.25	73.42
Ours w/o $\mathcal{L}_{\mathrm{KL}}$	**76.06**	**74.97**	71.59	**71.70**	**74.05**	**75.89**	**74.39**

as the backbone model. For the student, we select two networks, ResNet18 which shares a similar architecture design as the backbone, and MobileNetV2 [37] which is drastically different. We use random flip and Gaussian blur for data augmentation. The samples are randomly cropped and rescaled to 513×513 during the training and are resized to the same resolution during the testing. The student backbone ResNet18 is trained for 100 epochs with an initial learning rate 7e-3 on the Pascal VOC and 1e-2 on COCOStuff10k respectively. For MobileNetV2, the learning rate is set to 7e-3 for all datasets. We incorporate the cosine learning rate scheduler for all experiments. On Pascal VOC, the weight factors of Eq. 10 are α = 1, δ = 0.1, and ζ = 0.05. In terms of COCOStuff10k, the weight factors are α = 1, δ=0.6, ζ=0.6 for ResNet18, and are α=1, δ=0.4, ζ = 0.4 for MobileNetV2.

4.3. Image Classification

Results on Cifar-100. To show the generalization ability of our method, we applied our distillation approach to various network architectures, including ResNet [16], VGG [38] and WideResNet [49]. And in these experiments, we set $\theta(\cdot)$ as an identical function instead of linear transformation, *e.g.*, Conv+BN. As shown in Table 1, our method surpasses all baselines on six out of seven teacher-student settings, often by a significant margin. This evidences the effectiveness and generalization ability of our approach. Compared to the closest baseline, FitNet [36], which directly computes the distillation loss in one-to-one fashion, our approach improves on average 2.72%. The results when distilling to ResNet20 is interesting. In this case, using a less powerful teacher, ResNet56, results a better student performance on average comparing to using ResNet110. In particular, directly distilling the feature in one-to-one fashion deteriorates the student's performance compared to vanilla training. Our distillation approach addressed such mismatch and achieves 71.70% which is 2.64% better than the baseline. We also compare our method to the comparison methods in the setting with extra $\mathcal{L}_{\mathrm{KL}}$ in appendix.

Results on ImageNet. Since Cifar-100 only contains 50,000 training images, we further evaluate our approach on a more challenging dataset. Here, we choose ResNet34 and ResNet18 as teacher and student model respectively. We show the Top-1 accuracy of the student and teacher model in Table 2. Our method outperforms the state-of-the-art methods by a significant margin. Notice that, even without the help of $\mathcal{L}_{\mathrm{KL}}$, our model can reach 72.07% on a tiny ResNet18, comparing to some methods which rely on the $\mathcal{L}_{\mathrm{KL}}$ by more than 1%. When enabling $\mathcal{L}_{\mathrm{KL}}$, the proposed method can further improve the Top-1 accuracy of the student to 72.41%. Compared to SCKD [3] which uses an attention mechanism to re-allocate the most semantic-related teacher layers to the student, our method has a significant improvement. That means even matching two layers of teacher and student with similar semantics, the student may not be able to catch up with the teacher in the pixel-to-pixel manner due to semantic mismatch. In contrast, our method leverages a target-aware transformer to address the semantic mismatch in a more efficient manner.

4.4. Semantic Segmentation

As the feature map size is fairly small when performing distillation on image classification, we plan to further investigate the generalization ability of our method on semantic

Table 2. Top-1 Accuracy(%) on ImageNet validation set. The ResNet34 is employed as the teacher backbone and the ResNet18 is selected as the student backbone. Our method can boost the performance of the tiny ResNet18 beyond 72% and outperforms other methods without $\mathcal{L}_{\mathrm{KL}}$.

Method	Vanilla	AT [50]	CRD [42]	SAD [22]	ICKD [27]	KR [7]	Ours	KD [19]	SCKD [3]	CC [35]	RKD [33]	Ours	Teacher
w/ $\mathcal{L}_{\mathrm{KL}}$								✓	✓	✓	✓	✓	-
Top-1	70.04	70.59	71.17	71.38	71.59	71.61	72.07	70.68	70.87	70.74	71.34	**72.41**	73.31

Table 3. **Comparing the semantic segmentation results (in mIoU%) of different methods on Pascal VOC.** We can observe that our method surpasses all previous baselines by a significant margin. Specifically, on the popular compact architecture MobilenetV2, our method improves the student by 5.39% comparing to the stand-alone training, and by 1.06% comparing to the state-of-the-art method ICKD. † indicates reproducing by training 100 epochs, using the official released code.

	ResNet18	MobilenetV2
Teacher	78.43	78.43
Student	72.07	68.46
KD [19]	73.74	71.73
AT [50]	73.01	71.39
FitNet [36]	73.31	69.23
Overhaul† [18]	73.98	72.30
ICKD [27]	75.01	72.79
Ours	**75.76**	**73.85**

Table 4. Non-parametric vs. parametric implementation of target-aware transformer on ImageNet, where check mark indicates applying linear function.

$\theta(\cdot)$	$\gamma(\cdot)$	Top-1 Acc.
		72.22
	✓	**72.41**
✓	✓	72.35

Table 5. **Comparing the semantic segmentation results (in mIoU%) of different methods on COCOStuff10k.** As most baselines do not provide the code on the COCO dataset except KR, we only compare our method to KR in this case. We reproduce the baseline using the official code with the same training procedure. Our method surpasses the baseline by nearly 2%, and further demonstrates the effectiveness of our approach.

	Sutdent	KR [7]	Our	Teacher
ResNet18	26.33	26.73	**28.75**	33.10
MobilenetV2	26.29	26.63	**28.05**	33.10

segmentation, where the feature size is drastically larger. As in Section 3.2, we adapt our TaT method with the patch-group and anchor-point scheme. We select two popular benchmarks, Pascal VOC and COCOStuff10k, and present the results in Table 3 and Table 5 respectively. Our method clearly surpasses all baselines by a clear margin. For instance, on Pascal VOC, the proposed model can improve the MobileNetV2 by more than 5%, which shows great potential to unlock the hardware limitation. On the challenging benchmark COCOStuff10k, the model can improve the ResNet18 and MobileNetV2 by 2.42% and 1.76%.

Table 6. **Impact of function $\theta(\cdot)$ on a variety of network architectures.** We report the top-1 accuracy on Cifar-100. *id* indicates for identity mapping.

Teacher	Student	Conv+BN	*id*
ResNet56	ResNet20	71.45	**71.59**
ResNet110	ResNet20	71.68	**71.70**
ResNet110	ResNet32	73.75	**74.05**
ResNet32×4	ResNet8×4	75.30	**75.89**
VGG13	VGG8	73.48	**74.39**

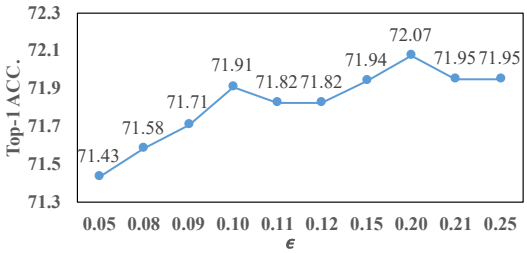


Figure 3. The performance of our model under different ϵ on ImageNet. Here the loss $\mathcal{L}_{\mathrm{KL}}$ is removed and α is set to 0.1.

4.5. Ablation Study

Here, we provide detailed ablation study to validate each component of our approach.

Linear transformation functions. We first study the impact of function $\theta(\cdot)$. We are interested in that if the learning target (*i.e.* teacher feature) is fixed, can the student adapt itself through target-aware transformer better? We compare different settings of $\theta(\cdot)$ including identical mapping against Conv+BN. The result on Cifar100 is presented on Table 6. Surprisingly, the identical mapping for $\theta(\cdot)$ always performs better.

We further investigate the non-parametric implementation by setting both $\theta(\cdot)$ and $\gamma(\cdot)$ as identical mapping on ImageNet (Table 4). The result shows that the semi-parametric version performs best, where the fixed teacher and the linear transformation applying to student feature can facilitate the student to reconfigure itself.

Validating ϵ. To investigate the efficacy brought by the proposed Eq. 8, we then further explore the different settings of the coefficient ϵ used in Eq. 9 (See Figure 3). When increased from 0.05 to 0.25, the objective $\mathcal{L}_{\mathrm{TaT}}$ can bring positive and stable effect.

We also conduct the thorough experiments to understand the contribution brought by the proposed patch-group dis-

Table 7. Contribution of patch-group and anchor-point distillation. We observe that patch-group distillation presents more efficacy.

Anchor-point	Patch-group	mIoU
		72.07
✓		75.37
	✓	75.63
✓	✓	**75.76**

Table 8. Performance (%) and training time (minutes) of anchor-point distillation on Pascal VOC under different kernel sizes.

Pooling kernel	2×2	4×4	8×8	16×16
Training time	423	403	389	374
mIoU	**75.37**	75.27	74.79	74.56

tillation $\mathcal{L}_{\mathrm{TaT}}^{\mathcal{P}}$ and anchor-point distillation $\mathcal{L}_{\mathrm{TaT}}^{\mathcal{A}}$. As discussed previously, $\mathcal{L}_{\mathrm{TaT}}^{\mathcal{A}}$ is proposed to learn the global representation to capture long-range dependency while $\mathcal{L}_{\mathrm{TaT}}^{\mathcal{P}}$ is designed to concentrate on local feature. By covering each one of them, the individual effectiveness of the two components can be examined. As shown in Table. 7, both objectives can improve the vanilla student significantly while $\mathcal{L}_{\mathrm{TaT}}^{\mathcal{P}}$ presents more efficacy. The combination of both components achieves the best performance, demonstrating that the two proposed objectives are complementary.

Validating the anchor-point distillation. Then, we give more insight concerning the proposed objectives' functionality through sensitivity analysis. Specifically, we investigate the hyper-parameters that would influence the behavior of the training process. In terms of the anchor-point distillation, this work utilizes average pooling to extract the anchor in a local area from the original feature, forming the associated anchor-point feature. It is a trade-off between reducing computation overhead and summarizing fine-grained spatial information since a bigger kernel would reduce feature size along with more informative representation, *e.g.*, when feature map size is reduced to 1×1, it degrades to ignoring the spatial information and posing one-to-one fashion distillation. Thus we study the pooling kernel size that directly yields different feature resolutions. The result exhibited in Table 8 shows that the amount of distillation calculation is greatly reduced with the increasing pooling size. On the other hand, excessive pooling range would omit useful and informative representation and damage the performance. We also report the mean training time. All experiments are conducted on a single Nvidia A100 GPU (40GB memory) with Intel Xeon CPU (8 cores) for 3 times.

Validating the patch-group distillation. Next we analyze the two key factors of patch-group distillation, *i.e.* patch size $h \times w$ and groups g. In Table 9, we found that generally, smaller patch size is advantageous to patch-group distillation and overlarge patch size, however, may be unfavourable since it approaches the original feature. Regarding the groups, it merges the patches as a group for joint distillation. In the experiment shown in Table 10, the patch size is set to 8×8, which divides the original feature map into $128/8 * 128/8 = 256$ patches. There are two extreme situations. When only one group is used, it indicates that all of the patches will be distilled jointly. On the contrary, using 256 groups means each patch is distilled individually. In this example, we found that 4 patches as a group can reach the best performance.

Table 9. Performance (%) of patch-group distillation on Pascal VOC under different settings of patch size ($h \times w$). Groups is equal to patches $g = n \times m$.

Patch size	32×32	16×16	8×8	4×4
mIoU	75.33	75.45	**75.50**	75.47

Table 10. Performance (%) of patch-group distillation on Pascal VOC under different settings of groups where patch size is 8×8 and patch numbers is 256.

Groups	1	32	64	128	256
mIoU	75.26	75.57	**75.63**	75.62	75.50

5. Conclusion

This work develops a framework for knowledge distillation through a target-aware transformation that enables the student to aggregate the useful semantic over itself to enhance the expressivity of each pixel, which allows the student to act as a whole to mimic the teacher rather than minimize each partial divergence in parallel. Our method is successfully extended to semantic segmentation by the proposed hierarchical distillation consisting of patch-group and anchor-point distillation, designed to focus on local feature and long-range dependency. We conduct thorough experiments to validate the effectiveness of the method and advance the state-of-the-art.

6. Discussion

Potential negative societal impact. Our method has no ethical risk on dataset usage and privacy violation as all the benchmarks are public and transparent.

Limitations. There are some issues of interest that we would like to explore in the future: (1) Currently, we only select the last layer of the backbone network for distillation. It would be interesting to see the efficacy when multiple layers are get involved with distillation which has been explored by some works [7, 50]. (2) Also, we didn't investigate the effectiveness on other applications like object detection, which may need to design the new objective to fit the nature of specific application.

Acknowledgement

This work was supported in part by National Natural Science Foundation of China (NSFC) under Grant No.61976233, National Key Research and Development Program of China (Grant NO. 2020AAA0108104), Australian Research Council (ARC) Discovery Early Career Researcher Award (DECRA) under DE190100626, and Alibaba Innovative Research (AIR) Program.

References

[1] Lei Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? *arXiv preprint arXiv:1312.6184*, 2013. 1

[2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 5

[3] Defang Chen, Jian-Ping Mei, Yeliang Zhang, Can Wang, Zhe Wang, Yan Feng, and Chun Chen. Cross-layer distillation with semantic calibration. *ArXiv*, abs/2012.03236, 2020. 2, 6, 7

[4] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017. 2

[5] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 5

[7] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5008–5017, 2021. 1, 7, 8

[8] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Everest Hinton. Big self-supervised models are strong semi-supervised learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 22243–22255, 2020. 1

[9] Zijun Cui, Tengfei Song, Yuru Wang, and Qiang Ji. Knowledge augmented deep neural networks for joint facial expression and action unit recognition. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[10] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 2

[11] Xing Dai, Zeren Jiang, Zhao Wu, Yiping Bao, Zhicheng Wang, Si Liu, and Erjin Zhou. General instance distillation for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2021. 2

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 5

[15] Bharath Hariharan, Pablo Arbeláez, Lubomir D. Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. *2011 International Conference on Computer Vision*, pages 991–998, 2011. 5

[16] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 6

[17] Tong He, Chunhua Shen, Zhi Tian, Dong Gong, Changming Sun, and Youliang Yan. Knowledge adaptation for efficient semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 578–587, 2019. 2

[18] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, H. Park, N. Kwak, and J. Choi. A comprehensive overhaul of feature distillation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1921–1930, 2019. 7, 11

[19] Geoffrey E. Hinton, Oriol Vinyals, and J. Dean. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531, 2015. 1, 2, 3, 6, 7

[20] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017. 2, 6

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 4

[22] Mingi Ji, Byeongho Heo, and S. Park. Show, attend and distill: Knowledge distillation via attention-based feature matching. *ArXiv*, abs/2102.02973, 2021. 1, 2, 7

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. *tech report*, 2009. 5

[24] Mingjie Li, Wenjia Cai, Rui Liu, Yuetian Weng, Xiaoyun Zhao, Cong Wang, Xin Chen, Zhong Liu, Caineng Pan, Mengke Li, et al. Ffa-ir: Towards an explainable and reliable medical report generation benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 3

[25] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2018. 1

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5

[27] Li Liu, Qingle Huang, Sihao Lin, Hongwei Xie, Bing Wang, Xiaojun Chang, and Xiaodan Liang. Exploring inter-channel correlation for diversity-preserved knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8271–8280, October 2021. 2, 4, 6, 7, 11

[28] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for

semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019. 2

[29] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 3

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[31] Asit K. Mishra and Debbie Marr. Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. In *ICLR*, 2018. 1

[32] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10870–10879, 2020. 2

[33] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3962–3971, 2019. 2, 6, 7

[34] N. Passalis and A. Tefas. Learning deep representations with probabilistic knowledge transfer. In *ECCV*, 2018. 2, 6

[35] Baoyun Peng, Xiao Jin, Jiaheng Liu, Shunfeng Zhou, Y. Wu, Y. Liu, Dong sheng Li, and Z. Zhang. Correlation congruence for knowledge distillation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5006–5015, 2019. 2, 6, 7

[36] A. Romero, Nicolas Ballas, S. Kahou, Antoine Chassang, C. Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *CoRR*, abs/1412.6550, 2015. 1, 2, 3, 6, 7

[37] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 6

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[39] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012. 5

[40] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147. PMLR, 2013. 5

[41] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[42] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *ICLR*, 2020. 1, 6, 7

[43] F. Tung and G. Mori. Similarity-preserving knowledge distillation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1365–1374, 2019. 2, 3, 6

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[45] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021. 3

[46] Xionghui Wang, Jian-Fang Hu, Jian-Huang Lai, Jianguo Zhang, and Wei-Shi Zheng. Progressive teacher-student learning for early action prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3556–3565, 2019. 2

[47] Yukang Wang, W. Zhou, T. Jiang, X. Bai, and Yongchao Xu. Intra-class feature variation distillation for semantic segmentation. In *ECCV*, 2020. 2

[48] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7130–7138, 2017. 1, 2, 6

[49] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *ArXiv*, abs/1605.07146, 2016. 2, 6

[50] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *ArXiv*, abs/1612.03928, 2017. 1, 2, 6, 7, 8

[51] Linfeng Zhang and Kaisheng Ma. Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors. In *International Conference on Learning Representations*, 2020. 2

Appendix

A.1 Asset Usage

This work is built upon some public dataset and code assets. We appreciate their efforts. The benchmark dataset has been introduced in main paper. Here we list the URL, version, and license of the code assets that we used:

Table 11. Usage of Code assets.

Exp.	URL	Ver.	Licence
ImageNet	`https://github.com/yoshitomo-matsubara/torchdistill`	`7b883ec`	MIT
Cifar100	`https://github.com/HobbitLong/RepDistiller`	`9b56e97`	BSD 2-Clause
Pascal VOC	`https://github.com/jfzhang95/pytorch-deeplab-xception`	`9135e10`	MIT
	`https://github.com/clovaai/overhaul-distillation`	`76344a8`	MIT
COCOStuff10k	`https://github.com/kazuto1011/deeplab-pytorch`	`4219467`	MIT
	`https://github.com/dvlab-research/ReviewKD`	`cede6ea`	N/A

A.2 Additional Experiments

A.2.1 Comparison on COCOStuff10k

For the experiments of semantic segmentation, we have compared our method to a variety of stat-of-the-art methods in the Section 4 of the main paper. In terms of COCOStuff10k, since some methods do not support this dataset, we re-implement them and the result is presented on Table 12. We found that our method is competitive and it outperforms the comparison methods.

Table 12. Comparison (mIoU%) on COCOStuff10k.

	ICKD [27]	Overhaul [18]	Ours
ResNet18	27.22	27.86	28.75
MobileNetV2	26.64	26.96	28.05

A.2.2 Hyperparameters on Cifar-100

We used Bayesian optimization to obtain the weight factors α and ϵ in Eq. 9. Here we show the searching result on different backbones (See Table 13). We found that in most cases (4 out of 6), ϵ is greater than α, which indicates that our proposed objective is more important than the standard Cross-entropy during distillation. For instance, in the distillation VGG13→VGG8, ϵ is 8 and α is only 0.1. We also found that for the similar architectures, the searching result is similar, *e.g.*, when WRN-40-2 and ResNet110 are selected as teacher.

Table 13. Coefficients α and ϵ on different backbones on Cifar-100.

Teacher	WRN-40-2	WRN-40-2	ResNet56	ResNet110	ResNet110	ResNet32×4	VGG13
Student	WRN-16-2	WRN-40-1	ResNet20	ResNet20	ResNet32	ResNet8×4	VGG8
α	0.8	0.7	0.8	1	1	6	0.1
ϵ	4	3.6	0.4	0.75	1	39	8

Table 14. Adding $\mathcal{L}_{\mathrm{KL}}$ on Cifar100.

Teacher Student	WRN-40-2 WRN-16-2	ResNet110 ResNet20	ResNet32×4 ResNet8×4	VGG13 VGG8
KD	74.92	70.67	73.33	72.98
FitNet+KD	75.12	70.67	74.66	73.22
AT+KD	75.32	70.97	74.53	73.48
SP+KD	74.98	71.02	74.02	73.49
CC+KD	75.09	70.88	74.21	73.04
RKD+KD	74.89	70.77	73.79	72.97
PKT+KD	75.33	70.72	74.23	73.25
NST+KD	74.67	71.01	74.28	73.33
CRD+KD	75.64	71.56	75.46	74.29
ICKD+KD	75.57	71.91	75.48	73.88
Ours+KD	**76.08**	**72.16**	**75.54**	**74.35**

A.2.3 Adding KD loss on Cifar-100

We report the result of our method in Table 14 with $\mathcal{L}_{\mathrm{KL}}$ loss to compare with the baselines under the same settings. Our method with KD loss surpasses all the baselines again.

A.2.4 Feature Visualization

We further visualize the feature map and the associated TaT map to intuitively understand the functionality behind the proposed Target-aware Transformer. As exhibited in Figure 4, we visualize the feature maps of student before and after distillation, which are compared to the feature map of teacher. The teacher backbone is ResNet34 and student backbone is ResNet18. The input images are randomly selected from ImageNet validation set. While the 4-th block (*i.e*. distillation layer) of ResNet34 and ResNet18 has 512 channels, we visualize 64 channels for better visualization.

Obviously, the reconfigured student feature (3rd column) has a more similar pattern with teacher feature (4th column), which demonstrates that TaT can effectively adapt the student to mimic the teacher. In terms of the TaT map, which controls the intensity of semantic aggregation, it is close to the identity matrix. Recall that we apply the linear function $\phi(\cdot)$ on student feature f^s. And the TaT map will be further applied on $\phi(f^s)$ to reconfigure the student feature, which is lately asked to minimize the L_2 distance with teacher feature. When the TaT map is an identity matrix, it means that $\phi(f^s)$ can reconstruct the teacher feature on its own. However, since TaT map is not strictly the identity matrix, it indicates that each pixel of $\phi(f^s)$ still needs to *borrow* the semantic from other position (mostly neighborhood) to enhance itself. Indeed, by aggregating the semantic from neighbors, each pixel increases the receptive field and thus semantic capacity. This demonstrates the semantic mismatch between student and teacher due to the variation on network depth and width.

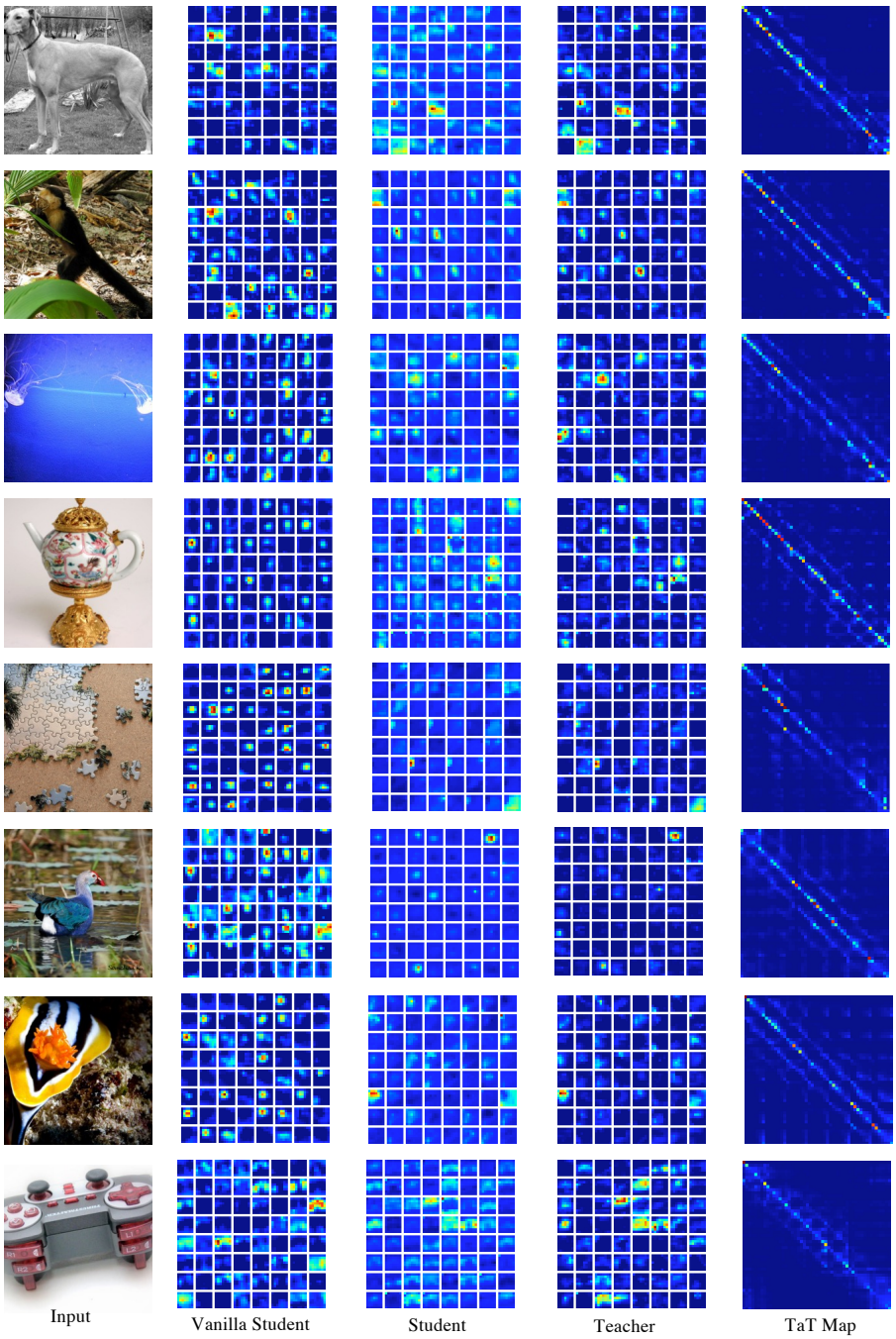


Figure 4. **Visualization of feature map and TaT map.** The input is selected from ImageNet validation set. The teacher backbone is ResNet34 and student backbone is ResNet18. The feature map of the distillation layer (4-th block) has been visualized. While there are 512 feature channels in total, we visualize 64 channels for better visualization. Through the Target-aware transformer, we found that the reconfigured student feature (3rd column) has a similar pattern with teacher feature (4th column). The associated TaT map has also been visualized, which indicates the student would aggregate the semantic mostly from neighbor to enhance its pixels.