Adaptive Perspective Distillation for Semantic Segmentation

Zhuotao Tian, Pengguang Chen, Xin Lai, Li Jiang, Shu Liu, *Member, IEEE*, Hengshuang Zhao, *Member, IEEE*, Bei Yu, *Member, IEEE*, Ming-Chang Yang, *Member, IEEE*, and Jiaya Jia, *Fellow, IEEE*

Abstract—Strong semantic segmentation models require large backbones to achieve promising performance, making it hard to adapt to real applications where effective real-time algorithms are needed. Knowledge distillation tackles this issue by letting the smaller model (student) produce similar pixel-wise predictions to that of a larger model (teacher). However, the classifier, which can be deemed as the perspective by which models perceive the encoded features for yielding observations (i.e., predictions), is shared by all training samples, fitting a universal feature distribution. Since good generalization to the entire distribution may bring the inferior specification to individual samples with a certain capacity, the shared universal perspective often overlooks details existing in each sample, causing degradation of knowledge distillation. In this paper, we propose Adaptive Perspective Distillation (APD) that creates an adaptive local perspective for each individual training sample. It extracts detailed contextual information from each training sample specifically, mining more details from the teacher and thus achieving better knowledge distillation results on the student. APD has no structural constraints to both teacher and student models, thus generalizing well to different semantic segmentation models. Extensive experiments on Cityscapes, ADE20K, and PASCAL-Context manifest the effectiveness of our proposed APD. Besides, APD can yield favorable performance gain to the models in both object detection and instance segmentation without bells and whistles.

Index Terms—Knowledge distillation, scene understanding, semantic segmentation

1 Introduction

Deep learning has significantly boosted the performance of semantic segmentation. Powerful segmentation models [3], [54] require strong feature extractors [9], [36], [41] to reach high performance. While real-time algorithms are more preferred in practice. Designing efficient segmentation models [21], [44], [53] is thus important.

Compared to hand-crafted efficient model design, knowledge distillation (KD) [12] is a more general technique for achieving high efficiency since KD can be applied to any existing models without structural constraints. Specifically, "knowledge" is distilled from a large model (teacher) to a smaller one (student) by minimizing the Kullback-Leibler divergence (KLD) between student output and soft target yielded by the teacher.

KD has been shown effective in classification [12], [28], [34], [42], while in segmentation, models are required to maintain the encoded features in certain resolutions and accomplish pixel-wise labeling by up-sampling to the original size. Contextual information is essential in segmentation because models cannot make predictions merely based on the RGB value of every single pixel. Design for contextual information enrichment (i.e., global pooling [19], pyramid pooling [54], dilated convolution [4] and attention [38]) can significantly improve the baselines. Previous methods [20], [39] propose distillation schemes to extract and transfer structured information on features, while it is notable that one important factor "perspective" in semantic segmentation is seldom studied.

Perspective works by representing the light that passes from a scene through a plane to the viewer's eye. In fact, *deep models perceive the encoded semantic features and make final predictions from the essential "perspective"*. We can consider the final classifier as a form of perspective for a model. Put differently, the inference of a segmentation model can be deemed as a process that *the perspective (classifier) projects the encoded high-level semantic information to yield observations (predictions) for the viewer*, as illustrated in Fig. 1. Compared to the student, the teacher usually has a better perspective because of the large feature encoder that can produce high-quality features to learn a good perspective, providing more accurate observations (predictions) used as soft targets in normal KD loss [12].

During KD, the teacher's feature encoder and perspective are fixed. Both of them generally fit the universal distribution given that they have been sufficiently trained on the entire training set. The fixed "universal perspective" of teacher achieves high-quality evaluation results by generalizing to all testing samples. However, the soft targets exploited with such a good generalization might not be the

- *Zhuotao Tian, Pengguang Chen, Xin Lai, Li Jiang, Bei Yu, Ming-Chang Yang, and Jiaya Jia are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. E-mail: {tianzhuotao, jlnemuru}@gmail.com, {pgchen, byu, mcyang, leojia}@cse.cuhk.edu.hk, laixin1998@outlook.com.*
- *Shu Liu is with Smartmore, Hong Kong. E-mail: liushuhust@gmail.com.*
- *Hengshuang Zhao is with the University of Oxford, OX1 2JD Oxford, U.K. E-mail: hengshuangzhao@gmail.com.*

Manuscript received 15 June 2021; revised 10 Jan. 2022; accepted 1 Mar. 2022. Date of publication 16 Mar. 2022; date of current version 6 Jan. 2023. (Corresponding author: Zhuotao Tian.) Recommended for acceptance by Z. Tu. Digital Object Identifier no. 10.1109/TPAMI.2022.3159581

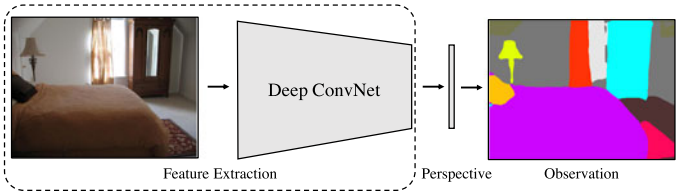


Fig. 1. Deep semantic segmentation framework is abstracted as the process that the final pixel-wise observation (prediction) is obtained from the perspective (classifier) based on encoded features produced by the deep neural networks.

optimal choice for transferring knowledge from the teacher to student, because, with a certain capacity, high generalization might cause poor specification that can reveal more useful information of the encoded features for decent knowledge distillation. To maintain good specification, the feature maps of different training samples should be projected by different perspectives to yield predictions, because even the same object may occur with varying co-occurrence information in different training samples, and a fixed universal perspective might not be able to well handle all the individual cases.

To address this key issue, we propose a new knowledge distillation method based on the concept of perspective for semantic segmentation. Our method enables models to form the adaptive perspective for every input image, i.e., different images are processed by different perspectives, based on their contextual contents. As illustrated in Figs. 2 and 3, the adaptive perspective is generated for each image and it can better describe the encoded feature distribution, which reveals more contextual details that are conducive to knowledge distillation. As teacher always learns a better

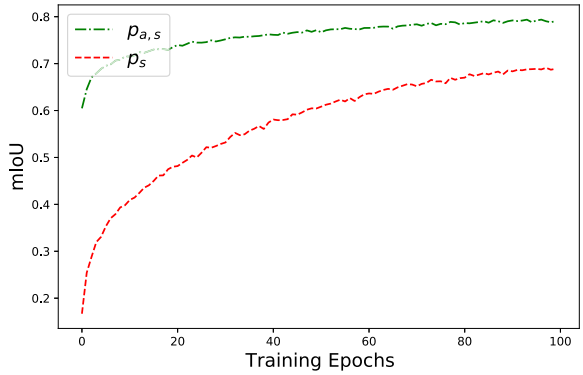


Fig. 3. Training mIoU curves of the auxiliary prediction $p_{a,s}$ and the main prediction p_s of the student model on PASCAL-Context. $p_{a,s}$ and p_s are obtained from the adaptive perspective and fixed universal perspective respectively. The auxiliary prediction $p_{a,s}$ achieves much higher mIoU on the training set because $p_{a,s}$ is generated by the adaptive perspective $\mathcal{A}_s$ that is with high specification to each image, mining more details for knowledge distillation and forming better decision boundaries as depicted by the bottom examples in Fig. 2. The comparison on the validation set is presented in Fig. 5.

universal perspective, we also align the adaptive perspectives of teacher and student. It makes the student learn to form better adaptive perspectives under the teacher's guidance. Besides, the auxiliary observations (predictions) are obtained from the adaptive perspectives of the teacher and student. They are then used for distillation from the adaptive perspectives, further boosting performance.

We name our method Adaptive Perspective Distillation (APD) since it offers an adaptive perspective to reveal more contextual cues for semantic segmentation. Our method is effective in boosting different models on various benchmark

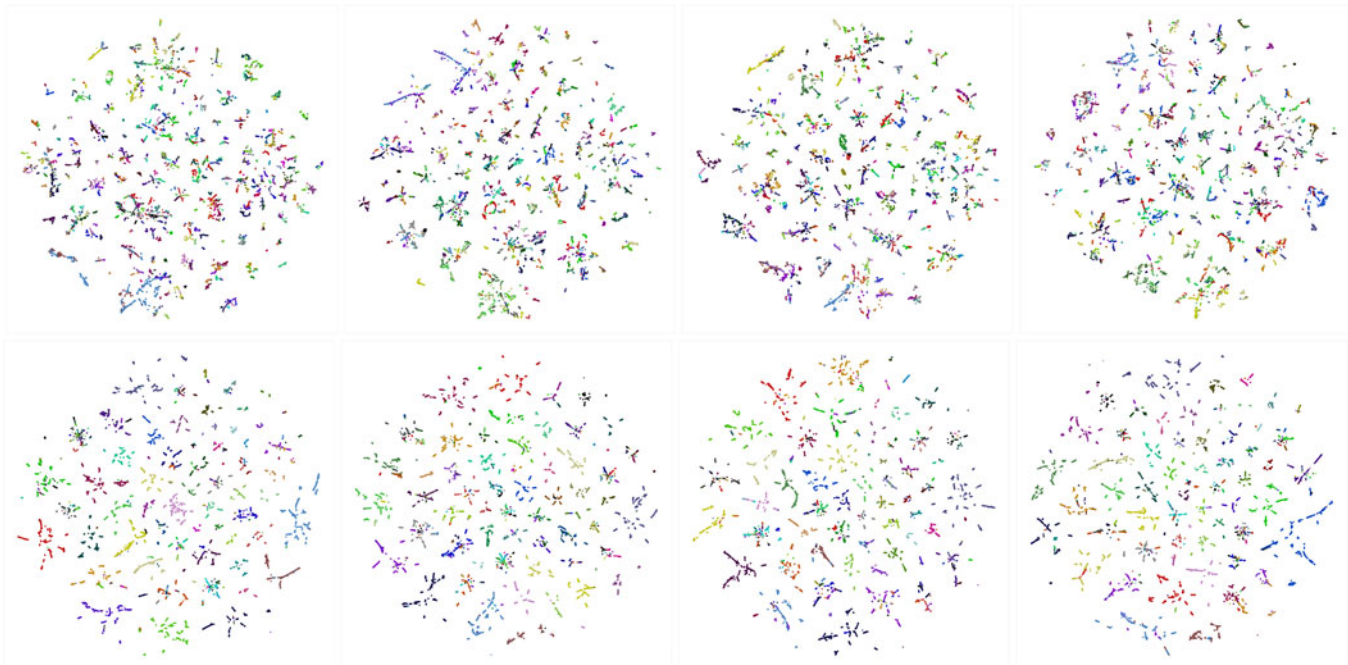


Fig. 2. Qualitative t-SNE [35] results of the difference between the fixed universal perspective (top) and adaptive perspective (bottom). Categories are represented by different colors. Top figures show that generally correct observations can be obtained by the fixed universal perspective, while the lack of specification to individual samples causes erroneous observations/knowledge for distillation. On the other hand, with our proposed APD, models learn to form adaptive perspectives that are clearer decision boundaries as demonstrated in the bottom figures where the adaptive perspective, conditioned on the content of each image, decently describes the feature distribution. Therefore, APD reveals additional detailed co-occurring semantic cues conditioned on individual training samples so as to better accomplish the knowledge distillation.

datasets, achieving advanced performance compared with state-of-the-art algorithms. Note only two light-weight projectors are introduced for knowledge distillation, and, after training, they are simply discarded without causing any structural modification to the original model during evaluation, manifesting the substantial practical merit. In summary, our contribution is threefold.

- Different from the common practice in KD, we exam individual images and generate adaptive perspectives and observations to improve knowledge distillation.
- The proposed APD is model-agnostic and achieves great success by significantly improving different semantic segmentation models on popular datasets without structural constraints.
- Our method is also effective for knowledge distillation on the tasks of object detection and instance segmentation, further demonstrating the generalization ability.

2 Related Work

Semantic Segmentation. Semantic segmentation is a fundamental and challenging task that requires accurate pixel-wise predictions for each image. FCN [31] is the first to adopt the convolution layers instead of the fully-connected layer to accomplish the semantic segmentation task. Encoder-decoder is developed [1], [23], [29] to let the encoded latent features refined by the decoder in steps. Dilated convolution [3], [46] enlarges the receptive field that is important for per-pixel predictions based on the contextual information. Pooling is another way for providing more contextual cues, such as global pooling [19], pyramid pooling [3], [43], [54], and strip pooling [13]. Note attention mechanism further boosts the performance by leveraging the long-range relationship across features [7], [8], [14], [17], [47], [48], [50], [55].

Recently, in order to perform pixel-wise semantic segmentation in real-time on mobile devices, efficient segmentation models are developed [21], [26], [45], [53]. E-Net [26] incorporates early down-sampling, filter factorization, and pooling in parallel with strided convolution to reduce the computation overhead without compromising accuracy. ESPNet [21] builds the efficient spatial pyramid (ESP) module with factorized convolutions to accelerate the model. ICNet [53] leverages the multi-resolution branches with label guidance to accomplish real-time inference effectively. BiSeNet [45] proposes the spatial- and context-path to obtain sufficient contextual cues efficiently.

Knowledge Distillation. Knowledge distillation was proposed by Hinton in [12]. It supervises a compact model by a larger pre-trained teacher in classification. The teacher provides soft labels, which contain useful "dark knowledge" for the student. The student could learn better results from the soft labels. Later, FitNet [28] distills knowledge from the features instead of the final prediction, which opened a new door in knowledge distillation. Following work [11], [25], [49] studied how to extract useful information from the features to better transfer to the student.

The study of knowledge distillation in semantic segmentation tasks commences in recent years. SKD [20] extracts structured information from the features. It also leverages a GAN network on top of the prediction of teacher and student to distill the holistic knowledge. Similar to SKD, the structural knowledge is also used in KA [10] by distilling the spatial correlation from the element-to-element similarity matrix, but differently, KA optimizes the feature similarity in a transferred latent domain formulated by an auto-encoder, instead of the original features used by SKD, to alleviate the issues brought by the inconsistency between teacher and the student. After that, IFVD [39] extracts the intra-class feature variation on the features. SKD replaces the transformation in SKD with an IFV transformation. Besides, CSCACE [24] makes use of both channel and spatial correlation (CSC) with an adaptive cross-entropy (ACE) loss that tries to combine the merits of the ground truth labels and predictions of the teacher network. More recently, [15] applies domain adaptive distillation to tackle the unsupervised domain adaptation problem and yields decent improvement. However, the study of knowledge distillation in semantic segmentation is still far from satisfactory.

Alternatively, in this paper, we analyze the knowledge distillation problem from a new view, and propose the Adaptive Perspective Distillation that achieves advanced performance on different baselines and datasets.

3 Preliminary - Knowledge Distillation

Large models always achieve better performance than the small ones because of the large capacity. As suggested by Hinton *et al.* [12], knowledge of a large model (teacher) can be transferred to the smaller one (students) via soft labels that are more informative than the one-hot hard labels. This process is called knowledge distillation (KD). By mimicking the soft labels predicted by the teacher, the student gradually obtains the "dark knowledge" contained in the teacher model, such as correlation between different entities, which is conducive to the representation learning and cannot be expressed by the hard labels.

Liu *et al.* [20] apply KD to semantic segmentation where the Kullback-Leibler divergence (KLD) is calculated in a pixel-wise manner. Formally, let H and W denote the height and width of the prediction, and the knowledge distillation loss $\mathcal{L}_{kd}$ is the average KLD of all pixels as

$$\mathcal{L}_{kd} = \frac{1}{H \times W} \sum_{x=1}^{H \times W} KLD(\boldsymbol{p}_s^x || \boldsymbol{p}_t^x), \tag{1}$$

where x is the pixel index, thus $\boldsymbol{p}_t^x$ and $\boldsymbol{p}_s^x$ represent the class probabilities of x-th pixel predicted by teacher and student models respectively.

It is worth noting that, normally, the teacher model is fixed during training to provide consistent soft targets p_t^x to student, and $\mathcal{L}_{kd}$ is used as an auxiliary loss that is optimized together with the main loss $\mathcal{L}_{ce}$ produced by p_s^x and one-hot hard labels. Therefore, the overall training objective $\mathcal{L}$ is

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda_{kd} \mathcal{L}_{kd}, \tag{2}$$

where λ_{kd} is set to 10 following [20], [39].

4 Adaptive Perspective Distillation

Overview. All semantic segmentation models can be decomposed into two components: 1) feature generator $\mathcal{G}$ and 2)

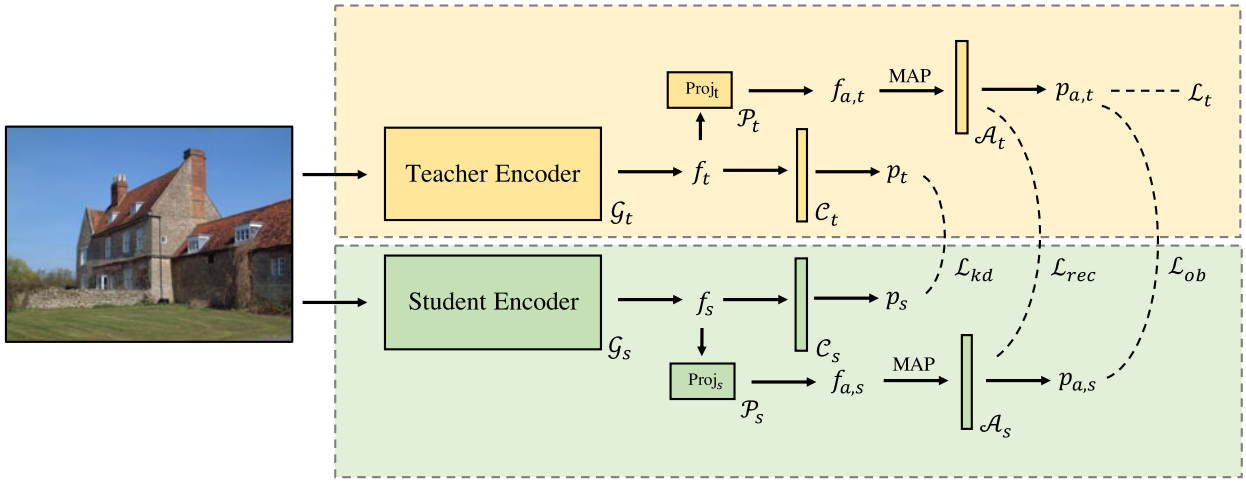


Fig. 4. Illustration of our method. The input image is first processed by teacher and student encoders ($\mathcal{G}_t$ and $\mathcal{G}_s$) respectively to get the encoded feature maps $\boldsymbol{f}_t$ and $\boldsymbol{f}_s$. To accomplish normal KD, $\mathcal{L}_{kd}$ [12] is applied to the predictions obtained from the main classifiers $\boldsymbol{C}_t$ and $\boldsymbol{C}_s$, offering a global perspective. $\boldsymbol{f}_t$ and $\boldsymbol{f}_s$ are also transformed by projectors ($\mathcal{P}_t$ and $\mathcal{P}_s$) to form adaptive classifiers $\mathcal{A}_t$ and $\mathcal{A}_s$, serving as local perspectives that reveal useful details by better describing the feature distributions as shown in Figs. 2 and 3. We note that the projected features $\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$ are *l*-2 normalized. Then, the distillation from the adaptive perspectives is accomplished by the proposed $\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$ that rectifies adaptive classifiers and aligns auxiliary predictions ($\boldsymbol{p}_{a,t}$ and $\boldsymbol{p}_{a,s}$) respectively. $\mathcal{L}_t$ only updates teacher's projector $\mathcal{P}_t$, and the gradients yielded by $\mathcal{L}_{kd}$, $\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$ will not be back-propagated to $\boldsymbol{p}_t$, $\mathcal{A}_t$ and $\boldsymbol{p}_{a,t}$. The normal cross entropy loss $\mathcal{L}_{ce}$ applied to $\boldsymbol{p}_s$ is omitted in this figure for simplicity.

classifier $\mathcal{C}$. Both $\mathcal{G}$ and $\mathcal{C}$ are fixed in the teacher model during distillation. Teacher's classifier $\mathcal{C}_t$ takes the features $\boldsymbol{f}_t$ extracted from $\mathcal{G}_t$ and produces soft targets for $\mathcal{L}_{kd}$. However, $\mathcal{C}_t$ fits the entire training set, and thus it provides a fixed universal perspective for mining knowledge from each feature map extracted by $\mathcal{G}_t$ of the teacher.

To further investigate the "dark knowledge" inside the teacher, we take a closer look at each training sample by forming individual adaptive perspectives $\mathcal{A}_t$ that are composed of semantic anchors (i.e., representative vectors for individual semantic classes) obtained from the encoded features $\boldsymbol{f}_t$, which serves as another auxiliary task providing local perspectives for distilling knowledge. Auxiliary observations $\boldsymbol{p}_{a,t}$ are then generated by adaptive perspectives $\mathcal{A}_t$ and encoded features $\boldsymbol{f}_t$ for transferring the knowledge from teacher to student. The student feature generator $\mathcal{G}_s$ is required to mimic $\mathcal{G}_t$ to yield similar adaptive perspectives $\mathcal{A}_s$, as well as the auxiliary observations $\boldsymbol{p}_{a,s}$ obtained from $\mathcal{A}_s$. Since both the adaptive perspective and auxiliary observations are generated specifically for each training sample, they provide more informative cues for KD. Our method is abstracted in Fig. 4.

Adaptive Perspective. In the following, we introduce the way to generate adaptive perspectives to better distill the knowledge between the teacher and student models. First, two light-weight projectors, i.e., two 2-layer Multi-layer Perceptrons (MLPs) with an intermediate ReLU activation layer, are used to produce the adapted features for constructing new perspectives with the same channel numbers, making our method model-agnostic because the teacher and student models usually have different output channels. We can formalize this procedure as

$$\boldsymbol{f}_{a,t} = \mathcal{P}_t(\boldsymbol{f}_t), \quad \boldsymbol{f}_{a,s} = \mathcal{P}_s(\boldsymbol{f}_s). \tag{3}$$

Masked average pooling (MAP) is then applied to $\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$ to generate the C-dimensional semantic anchors $\mathcal{A}_t^i$ and $\mathcal{A}_s^i \in \mathcal{R}^{[1\times C]} (i \in \{1, \ldots, N\})$ as shown in Eq. (4), where $\boldsymbol{M}_i \in \mathcal{R}^{[H\times W\times 1]}$ is the binary mask obtained from the ground truth label, indicating whether the features belong to class c_i, and x denotes the feature position. N represents the number of classes contained in the current image, and different images may have different values of N. For simplicity, we only discuss the case with one single image.

$$\mathcal{A}_t^i = \frac{\sum_{x=1}^{HW} \boldsymbol{f}_{a,t}^x \cdot \boldsymbol{M}_i^x}{\sum_{x=1}^{HW} \boldsymbol{M}_i^x}, \quad \mathcal{A}_s^i = \frac{\sum_{x=1}^{HW} \boldsymbol{f}_{a,s}^x \cdot \boldsymbol{M}_i^x}{\sum_{x=1}^{HW} \boldsymbol{M}_i^x}. \tag{4}$$

We name the collection of these features, i.e., semantic anchors, as "adaptive perspective" because they are then put together to form a classifier whose semantic information varies on different images, i.e., being "adaptive" to different contexts, for yielding auxiliary predictions during distillation. With the semantic information provided by the ground-truth labels, the adaptive perspective can better describe the encoded semantic intra- and inter-class distributions, as shown in Figs. 2 and 3 where more accurate predictions can be obtained from the adaptive perspective. Thus, though it cannot be used for the final prediction due to the use of the ground-truth label, it is suitable to distill knowledge between the student and teacher with deeper insight, i.e., how the model interprets the encoded features for different images. Note it is normal to add extra modules during distillation in literature. The proposed two projectors are not used during inference, so the model efficiency is not adversely affected.

After we get the adaptive perspectives, additional explicit observations can be obtained by calculating the cosine similarity between the adapted features ($\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$) and adaptive perspectives ($\mathcal{A}_t$ and $\mathcal{A}_s$) as Eqs. (5)-(6) where x is the pixel index, i and j are the indexes among N adaptive perspectives. Therefore, $\boldsymbol{p}_{a,t}^{x,i}$ and $\boldsymbol{p}_{a,s}^{x,i}$ tell how likely the x-th pixels belong to the corresponding i-th semantic anchors of teacher and student respectively.

$$\boldsymbol{p}_{a,t}^{x,i} = \frac{\exp(\cos(\boldsymbol{f}_{a,t}^x, \mathcal{A}_t^i)/\tau)}{\sum_{j=1}^{N} \exp(\cos(\boldsymbol{f}_{a,t}^x, \mathcal{A}_t^j)/\tau)}, \tag{5}$$

$$\boldsymbol{p}_{a,s}^{x,i} = \frac{\exp(\cos(\boldsymbol{f}_{a,s}^{x}, \mathcal{A}_s^i)/\tau)}{\sum_{j=1}^{N}\exp(\cos(\boldsymbol{f}_{a,s}^{x}, \mathcal{A}_s^j)/\tau)}. \tag{6}$$

A new hyper-parameter τ is introduced for yielding predictions via cosine similarity because the value of cosine similarity ranges from -1 to 1 thus the results of the Softmax operation performed in Eqs. (5)-(6) are constrained within a rather small scale. τ is adopted to enlarge the output scale for facilitating the optimization performed with Eqs. (5)-(6), and we empirically set τ to 0.1 in all experiments.

*Learning Objective for Teacher's Adaptive Perspective.*Teacher's projector $\mathcal{P}_t$ is randomly initialized by the default setting of PyTorch, thus it will collapse with meaningless interpretation without optimization. To ensure that $\mathcal{P}_t$ can provide representative perspectives $\mathcal{A}_t \in \mathcal{R}^{[N\times C]}$ that reveal more contextual details for each image, an explicit regularization is indispensable – features belonging to class c_i should get closer to $\mathcal{A}_t^i$ and are far from the semantic anchors of the other co-occurring categories. Therefore, we introduce the learning objective for teacher's projector $\mathcal{P}_t$ as

$$\mathcal{L}_t = \frac{1}{H\times W}\sum_{x=1}^{H\times W} -\log\frac{\exp(\cos(\boldsymbol{f}_{a,t}^{x}, \mathcal{A}_t^{c(x)})/\tau)}{\sum_{i=1}^{N}\exp(\cos(\boldsymbol{f}_{a,t}^{x}, \mathcal{A}_t^{i})/\tau)} \tag{7}$$

where $c(x)$ indicates the class that $\boldsymbol{f}_{a,t}^{x}$ belongs to. We note that the teacher model is fixed during KD, and $\mathcal{L}_t$ only optimizes the teacher's projector $\mathcal{P}_t$.

Learning Objective for the Student. Misaligned perspectives may result in different observations. Therefore, student's feature generator $\mathcal{G}_s$ and projector $\mathcal{P}_s$ are first required to mimic teacher by producing similar perspectives. To realize this objective, we apply $\mathcal{L}_{rec}$ to accomplish the rectification on the adaptive perspectives of teacher and student. $\mathcal{L}_{rec}$ directly encourages the similarity between $\mathcal{A}_t$ and $\mathcal{A}_s$ as

$$\mathcal{L}_{rec} = 1 - \frac{1}{N}\sum_{i=1}^{N}\cos(\mathcal{A}_s^i, \mathcal{A}_t^i). \tag{8}$$

Furthermore, the observation obtained from the student's perspective also needs to imitate the teacher's observation, which can be achieved by minimizing KLD between their observations $\boldsymbol{p}_{a,t}$ and $\boldsymbol{p}_{a,s}$ as

$$\mathcal{L}_{ob} = \frac{1}{H\times W}\sum_{x=1}^{H\times W} KLD(\boldsymbol{p}_{a,s}^{x} || \boldsymbol{p}_{a,t}^{x}). \tag{9}$$

The overall Adaptive Perspective Distillation objective for student extends the loss in Eq. (2) with $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$ providing extra informative cues for distillation as

$$\mathcal{L}_s = \mathcal{L}_{ce} + \lambda_{kd}(\mathcal{L}_{kd} + \mathcal{L}_{ob}) + \lambda_{rec}\mathcal{L}_{rec}, \tag{10}$$

where λ_{kd} for $\mathcal{L}_{kd}$ is set to 10, the same as those in SKD and IFVD for fair comparison. As for $\mathcal{L}_{ob}$ that minimizes the Kullback-Leibler divergence from the adaptive observations, its loss weight is empirically set to λ_{kd}. The weighting factor λ_{rec} is set to 10. τ for scaling the cosine similarity is 0.1 in $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$. The sensitivity analysis of λ_{rec} and τ is given in Section 5.4. They both work well on all datasets with different backbones without further tuning.

Optimization. $\mathcal{L}_t$ only optimizes the teacher's projector $\mathcal{P}_t$ because the gradients yielded by $\mathcal{L}_{kd}$, $\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$ will not be back-propagated to $\boldsymbol{p}_t$, $\mathcal{A}_t$ and $\boldsymbol{p}_{a,t}$, as shown in Fig. 4. On the other hand, $\mathcal{L}_s$ optimizes the entire student model, i.e., feature generator $\mathcal{G}_s$ and classifier $\mathcal{C}_s$, as well as the projector $\mathcal{P}_s$. Therefore, $\mathcal{L}_t$ and $\mathcal{L}_s$ work independently on each training batch. As shown in Algorithm 1, θ_t represents the parameters of teacher's projector $\mathcal{P}_t$, and θ_s denotes all trainable parameters of the student model. Specifically, given a training batch, the teacher's projector $\mathcal{P}_t$ is first updated in lines 4-5. Then, $\mathcal{P}_t$ is detached to accomplish the update of student's parameters θ_s in lines 6-7 without back-propagating the gradients to update θ_t.

Algorithm 1. Optimization of APD

Require: $p(\mathcal{B})$: distribution over the training set.
Require: α, β: step size hyper-parameters.
1: Randomly initialize θ_t and θ_s.
2: **while** not done **do**
3: Sample a batch of samples $\mathcal{B}_i \sim p(\mathcal{B})$
4: Yield $\nabla_{\theta_t}\mathcal{L}_t$ w.r.t. $|\mathcal{B}_i|$ training samples
5: Update $\theta_t \leftarrow \theta_t - \alpha\nabla_{\theta_t}\mathcal{L}_t$
6: Yield $\nabla_{\theta_s}\mathcal{L}_s$ w.r.t. $|\mathcal{B}_i|$ training samples
7: Update $\theta_s \leftarrow \theta_s - \beta\nabla_{\theta_s}\mathcal{L}_s$
8: **end while**

5 Experiments

5.1 Dataset Description

Cityscapes [6] focuses on semantic understanding of urban street scenes. It contains 5000 finely annotated images. Specifically, 2975, 500 and 1525 images for training, validation and testing respectively. 19 classes are required in prediction for evaluation.

ADE20K [56] is a rather challenging dataset that spans diverse annotations of scenes, objects, parts of objects, and in some cases even parts of parts. ADE20K contains up to 150 classes and diverse scenes for semantic segmentation. 20000, 2000 and 3000 images are used for training, validation and testing.

PASCAL-Context [22] extends the original PASCAL VOC semantic segmentation task with more detailed annotations for the whole scene. 4998 and 5105 images are used for training and validation, and 9637 images are used for testing. We evaluate all models on 60 categories (59 + background), following the practice of MMSegmentation [5].

COCO [18] is the most popular and challenging dataset for object detection and instance segmentation. In this paper, we use "COCO" to represent COCO 2017 dataset. It contains more than 200,000 images and 80 object categories for train, validation, and test sets. We use the COCO 2017 *train* set for training and report the validation results on the COCO 2017 *val* set. The results are reported in COCO-style mAP.

5.2 Implementation Details

We adopt three popular scene parsing benchmark datasets (Cityscapes [6], ADE20K [56] and PASCAL-Context [22]) in experiments. Models are trained and evaluated on the training and validation sets of these datasets respectively by default.

TABLE 1
Training Configurations on Different Datasets

Dataset	Epoch	BS	InitLR	PS
Cityscapes [6]	200	8	5e-3	713
ADE20K [56]	100	8	5e-3	473
PASCAL-Context [22]	100	12	7.5e-4	473

Epoch: Training Epoch Number. BS: Batch Size. InitLR: Initial Training Learning Rate. PS: Patch Size for Training.

Both projectors $\mathcal{P}_t$ and $\mathcal{P}_s$ are composed of two 1×1 convolutional layers (denoted as $d_{in} \times d_{out}$) with an intermediate ReLU activation layer, while the difference lies in the input & output dimensions of the convolutional layers. Let d_t and d_s represent the dimensions of features $\boldsymbol{f}_t$ and $\boldsymbol{f}_s$ yielded by teacher and student feature generators $\mathcal{G}_t$ and $\mathcal{G}_s$, respectively. Usually, because the teacher is with a larger capacity and $d_t \geq d_s$, teacher's projector $\mathcal{P}_t$ is required to compress the dimension of $\boldsymbol{f}_t$ from d_t to d_s, matching that of the student feature $\boldsymbol{f}_s$. Therefore, the structure of $\mathcal{P}_t$ is as: $[d_t \times d_s \rightarrow \text{ReLU} \rightarrow d_s \times d_s]$, and the structure of $\mathcal{P}_s$ is: $[d_s \times d_s \rightarrow \text{ReLU} \rightarrow d_s \times d_s]$. Then, the projected features are l-2 normalized for calculating the cosine similarity.

The semantic segmentation models are built upon Semseg [52]. Student models are trained following the default configuration of PSPNet [54] except for the initial learning rate and batch size because PSPNet uses 8 GPUs by default while we use 4 GPUs for training. Specific epoch numbers, initial learning rates and training patch sizes used for different datasets are summarized in Table 1. SGD is used for optimization. Weight decay and momentum are set to 0.0001 and 0.9 respectively. The "poly" learning rate decay [3] is used by multiplying the initial learning rate with $(1 - current\_itermax\_iter)^{power}$, where $power$ is set to 0.9. All models are optimized without OHEM. As for the teacher, since the feature generator and classifier are fixed during training, only the projector $\mathcal{P}_t$ requires gradients. $\mathcal{P}_t$ is optimized by Adam optimizer with initial learning rate 1e-5 and beta (0.9, 0.99), which generalize well on all datasets without additional tuning. Both SKD [20] and IFVD [39] incorporate a GAN loss to accomplish holistic distillation, while the proposed APD does not adopt this strategy during training.

Data augmentation includes mirroring, re-scaling from 0.5 and 2.0, and random rotation from -10 to 10 degrees. Finally, image patches are cropped from the original images as training samples. During evaluation, following the official implementation of PSPNet [54], the sliding window inference strategy with the training crop size is adopted for experiments in semantic segmentation, and we output the prediction without additional post-processing (e.g., fully connected conditional random field (CRF) [16] and multi-scale testing). All experiments are conducted on PyTorch with four NVIDIA GTX 2080Ti GPUs, and results are obtained without altering the original labels. We will make our code publicly available for reproducing all experimental results in this paper.

5.3 Comparison With State-of-the-Art

In this section, we show quantitative and qualitative comparison with recently proposed methods SKD [20], CSCACE [24], KA [10] and IFVD [39]. For a fair comparison, we reproduce these methods in the same training and testing settings as our method.

TABLE 2
Performance Comparison With State-of-the-Art Methods on Cityscapes *val* With PSPNet [54] and DeepLab-V3 [3]

Methods	Backbone	PSPNet	DeepLab-V3
Teacher	RN-101	78.15	78.47
Student-I	RN-18	74.15	74.47
+ KD	RN-18	74.81	73.67
+ SKD	RN-18	74.56	74.03
+ IFVD	RN-18	74.10	74.99
+ CSCACE	RN-18	74.50	74.81
+ KA	RN-18	74.59	74.87
+ Ours	RN-18	**75.68**	**75.45**
Student-II	RN-18*	73.20	74.19
+ KD	RN-18*	73.33	74.53
+ SKD	RN-18*	73.40	74.00
+ IFVD	RN-18*	73.63	74.47
+ CSCACE	RN-18*	72.98	74.46
+ KA	RN-18*	74.18	73.82
+ Ours	RN-18*	**74.77**	**75.14**
Student-III	MN2-1.0	71.34	71.40
+ KD	MN2-1.0	71.91	71.94
+ SKD	MN2-1.0	72.40	71.34
+ IFVD	MN2-1.0	72.94	70.79
+ CSCACE	MN2-1.0	72.56	71.92
+ KA	MN2-1.0	71.01	71.89
+ Ours	MN2-1.0	**73.66**	**74.47**
Student-IV	EN-B0	72.30	71.54
+ KD	EN-B0	73.32	72.55
+ SKD	EN-B0	73.45	69.47
+ IFVD	EN-B0	74.43	72.93
+ CSCACE	EN-B0	74.15	73.25
+ KA	EN-B0	73.83	72.61
+ Ours	EN-B0	**75.79**	**74.92**
Teacher	MN2-1.0	71.34	71.40
Student	MN2-0.5	63.34	63.89
+ KD	MN2-0.5	64.60	66.03
+ SKD	MN2-0.5	65.06	65.84
+ IFVD	MN2-0.5	65.31	66.78
+ CSCACE	MN2-0.5	65.31	66.82
+ KA	MN2-0.5	64.64	66.06
+ Ours	MN2-0.5	**67.28**	**67.58**

RN, MN2 and EN Represent ResNet [9], MobileNet-V2 [30] and EfficientNet [33] Respectively. Models With RN-18 are Trained With 512×512 Crop Size, and the Others are Trained With 713 × 713 Crops Following PSPNet [54].*

Statistical Comparisons. As shown in Table 2, we make comparison between the teacher PSPNet-R101 and student models on different backbones, i.e., ResNet-18 [9], MobileNet-V2 [30] and EfficientNet [33]. Since our method enables models to form new local perspectives that mine extra useful information, the proposed Adaptive Perspective Distillation achieves better performance compared to other methods when different student backbones are adopted.

We note SKD and IFVD only distill knowledge from an unchanged global view with a fixed classifier of the teacher. It is via $\mathcal{L}_{kd}$ [12] without new perspectives, causing limited knowledge that can be transferred. Contrarily, the proposed method mines extra cues for distillation by creating a new perspective for every single image specifically, and thus our method consistently yields significant performance gain to all student models. Besides, in Section 5.4, we show that our proposed APD is complementary to SKD and IFVD.

TABLE 3
Efficiency Comparison on Cityscapes *Test*

Methods	*test* mIoU	Params (M)	FLOPS (G)
ENet [26]	58.3	0.3580	3.612
ESPNet [21]	60.3	0.3635	4.422
FCN [31]	65.3	134.5	333.9
ERFNet	68.0	2.067	25.60
ICNet [53]	69.5	26.50	28.30
RefineNet	73.6	118.1	525.7
PSPNet [54]	78.4	70.43	574.9
RN-18 + SKD	72.9	16.31	148.2
RN-18 + IFVD	73.2	16.31	148.2
RN-18 + CSCACE	73.0	16.31	148.2
RN-18 + KA	72.8	16.31	148.2
RN-18 + Ours	**74.9**	16.31	148.2
MN2-1.0 + SKD	72.1	4.840	39.44
MN2-1.0 + IFVD	72.0	4.840	39.44
MN2-1.0 + CSCACE	71.6	4.840	39.44
MN2-1.0 + KA	71.1	4.840	39.44
MN2-1.0 + Ours	**73.5**	4.840	39.44
EN-B0 + SKD	73.0	13.44	95.86
EN-B0 + IFVD	73.6	13.44	95.86
EN-B0 + CSCACE	73.5	13.44	95.86
EN-B0 + KA	72.9	13.44	95.86
EN-B0 + Ours	**75.2**	13.44	95.86

Teacher Model is PSPNet [54] With ResNet-101. RN, MN2 and EN Represent ResNet [9], MobileNet-V2 [30] and EfficientNet [33] Respectively.

The efficiency comparison is illustrated in Table 3 with the test mIoU results on Cityscapes. We also conduct experiments with PSPNet on ADE20K and PASCAL-Context to show the superiority of our method on different datasets. Results are shown in Table 4.

Cross-Model Distillation. To further manifest the generalization ability of the prosed method, we conduct experiments across different models, i.e., PSPNet → DeepLab-V3 and DeepLab-V3 → PSPNet. The cross-model distillation Results are shown in Table 5. It can be observed that IFVD and SKD may adversely affect the performance for cross-model distillation as sometimes they may cause performance degradation compared to the results of KD proposed by Hinton *et al.* [12]. On the contrary, the proposed method still consistently brings decent performance gain in the practical cross-model setting.

Comparison With Validation Curves. Qualitative comparison with validation curves is presented in Fig. 5. We note that these validation results are obtained from the center regions cropped with the training patch sizes (i.e., 473×473 for ADE20K [56] and PASCAL-Context [22], and 713×713 for Cityscapes [6]), which is different from the formal evaluation phase when the sliding windows inference strategy is adopted. The center cropping for the intermediate validation and the sliding window inference for the final evaluation are both implemented according to the official PyTorch implementation of PSPNet.

From Fig. 5, we can observe that APD consistently outperforms other methods by a large margin on both three benchmark datasets throughout the entire training process, which manifests the robustness of our method.

Discussion. The proposed Adaptive Perspective Distillation (APD) aims at: 1) letting the student mimic teacher to form local perspectives that can well describe temporary feature distributions; 2) learning to form similar observations (predictions) based on the local perspectives (classifiers). In other words, both inter- and intra-class distributions are leveraged by APD to probe more cues from individual training samples, and APD attempts to find a better distribution descriptor for them.

Both SKD [20] and KA [10] exploit the structured information without explicitly modelling the feature distribution, and the difference is that KA adopts an auto-encoder to accomplish knowledge transfer on the compressed

TABLE 4
Performance Comparison With State-of-the-Art Methods Using PSPNet on the Validation Sets of Three Popular Benchmarks: Cityscapes [6], ADE20K [56] and PASCAL-Context [22]

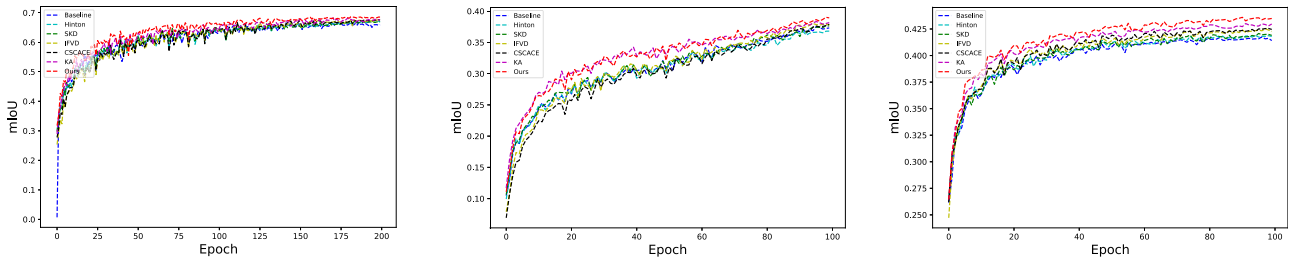
Methods	Cityscapes	ADE20K	PASCAL-Context
Teacher	78.15	43.44	48.50
Student	74.15	37.19	42.29
+ KD [12]	74.81	37.69	42.45
+ SKD [20]	74.56	37.61	42.53
+ IFVD [39]	74.10	37.89	42.74
+ CSCACE [24]	74.50	37.50	42.86
+ KA [10]	74.59	38.26	43.13
+ Ours	**75.68**	**39.25**	**43.96**

Teacher and Student Models Adopt ResNet-101 and ResNet-18 as Their Backbones.

TABLE 5
Cross-Model Distillation Results on Cityscapes *Val* With PSPNet [54] and DeepLab-V3 [3]

Method	Backbone	PSPNet →DL-V3	DL-V3 → PSPNet
Teacher	RN-101	78.15	78.47
Student-I	RN-18	74.15	74.47
+ KD	RN-18	75.13	73.50
+ SKD	RN-18	75.65	73.67
+ IFVD	RN-18	75.42	74.29
+ CSCACE	RN-18	74.93	74.33
+ KA	RN-18	75.64	74.58
+ Ours	RN-18	**76.01**	**75.90**
Student-III	MN2-1.0	71.34	71.40
+ KD	MN2-1.0	71.81	71.57
+ SKD	MN2-1.0	72.45	71.74
+ IFVD	MN2-1.0	70.97	72.54
+ CSCACE	MN2-1.0	71.54	71.80
+ KA	MN2-1.0	70.82	70.61
+ Ours	MN2-1.0	**73.22**	**73.66**
Student-IV	EN-B0	72.30	71.54
+ KD	EN-B0	72.66	73.73
+ SKD	EN-B0	72.29	73.69
+ IFVD	EN-B0	72.87	74.06
+ CSCACE	EN-B0	73.28	74.28
+ KA	EN-B0	72.46	73.62
+ Ours	EN-B0	**75.03**	**75.51**
Teacher	MN2-1.0	71.34	71.40
Student	MN2-0.5	63.34	63.89
+ KD	MN2-0.5	64.42	64.88
+ SKD	MN2-0.5	64.11	64.47
+ IFVD	MN2-0.5	64.27	64.36
+ CSCACE	MN2-0.5	65.13	65.04
+ KA	MN2-0.5	65.46	64.68
+ Ours	MN2-0.5	**67.14**	**66.90**

RN, MN2 and EN Represent ResNet [9], MobileNet-V2 [30] and EfficientNet [33] Respectively. PSPNet → DL-V3 Means the Teacher Network is PSPNet and the Student is DeepLab-V3, and Vise Versa.



(a) Cityscapes (b) ADE20K (c) PASCAL-Context

Fig. 5. Validation mIoU curves on Cityscapes, ADE20K and PASCAL-Context. Our proposed APD (colored in red) consistently outperforms other methods throughout the training process. The teacher is PSPNet with ResNet-101 and the student is PSPNet with ResNet-18.

features, while SKD directly let the student mimic the raw correlation matrix without the feature transformation.

It is worth noting that the "adaptation" of KA is different from our proposed "adaptive perspective". Specifically, the "adaptation" of KA denotes the use of an auto-encoder that adapts the teacher's features to a compressed feature space to extract essential information for distillation. However, the term "adaptive perspective" in the proposed APD refers to the local classifier that is conditioned on the semantic information varying in individual images, serving as a better feature distribution descriptor for mining additional details during the distillation process.

Differently, CSCACE [24] leverages the channel-wise correlation and a pseudo-label based adaptive cross-entropy loss, while the experiments show that the results of CSCACE are also less-satisfying that ours. Besides, IFVD [39] only makes use of the intra-class distribution. Though IFVD also adopts cosine similarity calculation for capturing the intra-class relation, without the inter-class reasoning in Eqs. (5)-(6) and $\mathcal{L}_{ob}$, IFVD achieves inferior results compared to APD. Moreover, with an eye towards a better distribution descriptor, APD applies $\mathcal{L}_t$ and $\mathcal{L}_{rec}$ to regularize the teacher and student models respectively. Therefore, the proposed learning objectives introduces further improvement to IFVD as shown in Table 8. We note that the hyper-parameter τ of APD is used for scaling the output of cosine similarity, and it is helpful by making the temporary inter-class distribution more discriminative.

Visual Comparison. We present the qualitative comparison between SKD and IFVD on Cityscapes, ADE20K and PASCAL-Context in Fig. 6 where it is observed that our predictions are generally better than the others by capturing more local contextual information for distillation.

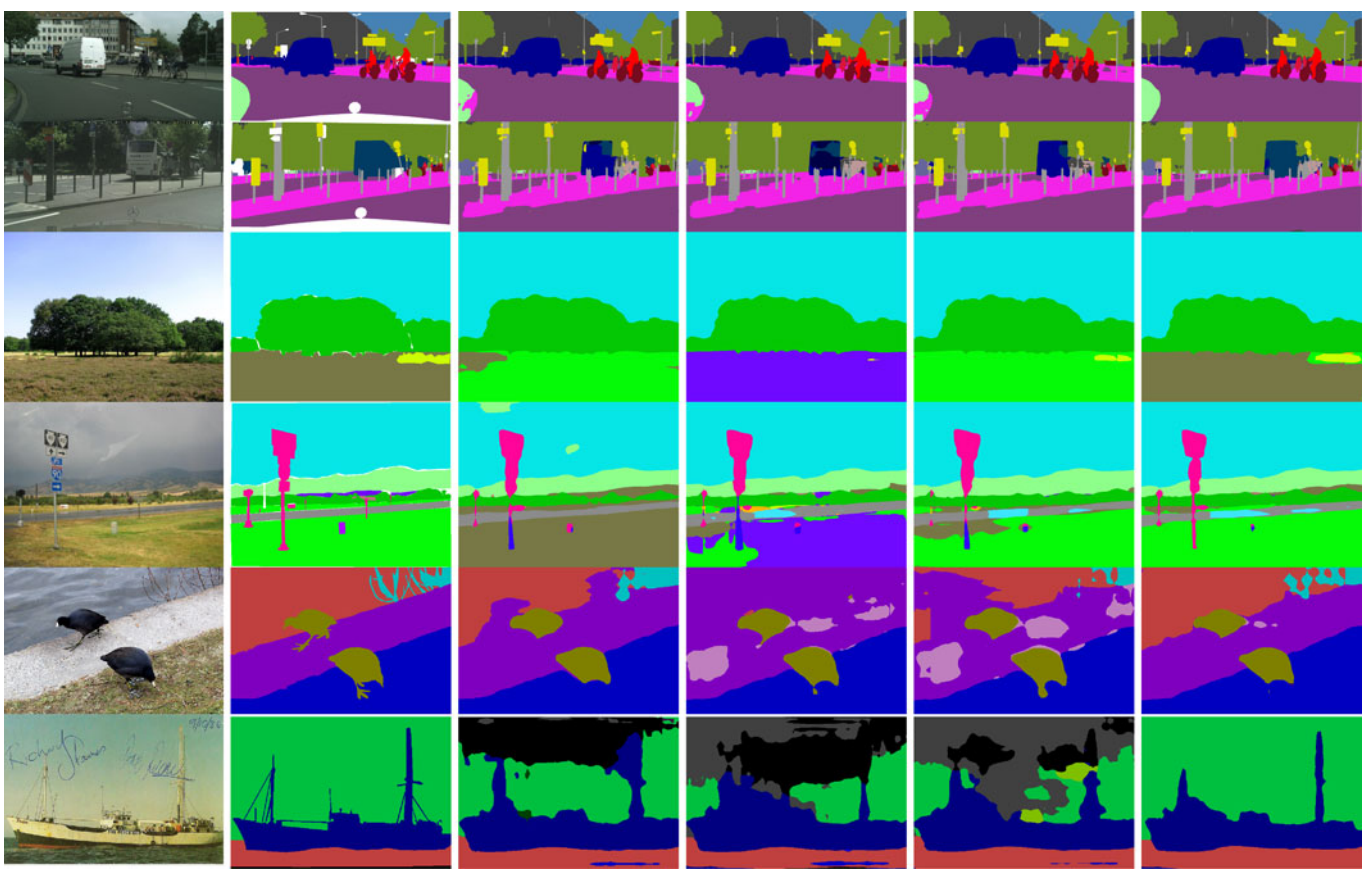


Fig. 6. Visual comparison on Cityscapes, ADE20K and PASCAL-Context. White regions in GT are ignored during evaluation.

TABLE 6
Ablation Study on the Validation Sets Of PASCAL-Context and Cityscapes

Exp.	$\mathcal{L}_{kd}$	$\mathcal{L}_{ob}$	$\mathcal{L}_{ce}^{Local}$	$\mathcal{L}_{rec}$	Context	City
I	-	-	-	-	42.29	74.15
II	✓	-	-	-	42.48	74.81
III	✓	✓	-	-	43.32	75.27
IV	✓	-	✓	-	42.92	74.47
V	✓	-	-	✓	42.88	74.62
VI	✓	✓	-	✓	**43.96**	75.68
VII	✓	-	✓	✓	43.38	75.32
VIII	✓	✓	✓	✓	43.87	**75.72**
IX	-	✓	-	✓	43.81	75.70
X	-	-	✓	-	42.50	74.41

Teacher is PSPNet With ResNet-101 and Student is PSPNet With ResNet-18. The First Column Denotes the Experiment IDs. $\mathcal{L}_{ob}$ Uses With Soft Targets $\boldsymbol{p}_{a,t}$ as Shown in Eq. (9), While $\mathcal{L}_{ce}^{Local}$ Means Directly Applying Cross Entropy Loss on $\boldsymbol{p}_{a,s}$ With One-Hot Hard Targets.

TABLE 7
Ablation Study of Different Methods for Yielding $\mathcal{L}_{rec}$ on the Validation Sets of PASCAL-Context and Cityscapes

Datasets	Center	Pixel	Center & Pixel
Context	43.96	43.40	44.06
Cityscapes	75.68	75.21	75.72

'center' and 'pixel' Adopt the Alignment Between Class Centers and Individual Pixels Respectively. 'center & Pixel' Combines Both.

5.4 Ablation Study

In this Section, we first verify that $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$ are important to align teacher's observations and perspectives respectively. Then, as two projectors $\mathcal{P}_t$ and $\mathcal{P}_s$ are introduced during student training, we show that the improvement brought by $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$ is not originated from these additional learnable modules. Besides, we provide a sensitivity analysis of λ_{rec} and τ to show the robustness of our method.

Effectiveness of $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$. The proposed Adaptive Perspective Distillation (APD) has two components $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$. $\mathcal{L}_{ob}$ accomplishes the alignment between auxiliary predictions $\boldsymbol{p}_{a,t}$ and $\boldsymbol{p}_{a,s}$ (i.e., observations) obtained from the adaptive perspectives, while $\mathcal{L}_{rec}$ rectifies student view $\mathcal{A}_s$, making it similar to $\mathcal{A}_t$ of teacher. Because the adaptive $\mathcal{A}_s$ encodes more specific semantic details for each image than the fixed $\mathcal{C}_s$, the produced $\boldsymbol{p}_{a,s}$ are generally more accurate than $\boldsymbol{p}_s$ obtained from $\mathcal{C}_s$, as demonstrated in Fig. 3. Results in Table 6 show that the observation alignment and perspective rectification are both indispensable.

Different perspectives result in varying observations. Thus perspective rectification is helpful for the observation alignment as proved by Exp.III & Exp.VI and Exp.IV & Exp. VII. However, without observation alignment $\mathcal{L}_{ob}$, implementing $\mathcal{L}_{rec}$ alone with $\mathcal{L}_{kd}$ in Exp.V only slightly improves the performance of Exp.II. On the other hand, merely applying observation alignment via $\mathcal{L}_{ob}$ achieves decent improvement as shown by Exp.II & Exp.III. When perspectives are rectified by $\mathcal{L}_{rec}$, $\mathcal{L}_{ob}$ boosts performance from 42.88 from 43.96 as shown in Exp.V & Exp.VI.

In Eq. (9), $\boldsymbol{p}_{a,t}$ is used as soft targets to distill knowledge from teacher to student in the proposed APD. An alternative is to replace the soft targets with one-hot labels, denoted as $\mathcal{L}_{ce}^{Local}$ in Table 6, thus Kullback-Leibler divergence in Eq. (9) equals to the standard Cross Entropy Loss. We note that the difference between $\mathcal{L}_{ce}^{Local}$ and $\mathcal{L}_{ce}$ is that the former is applied to local predictions $\boldsymbol{p}_{a,s}$ while the latter is applied to $\boldsymbol{p}_s$.

Soft targets encode the "dark knowledge" of teacher and are more informative than one-hot hard labels. Therefore, superior performance has been achieved by $\mathcal{L}_{ob}$ (Exp.III & Exp.VI) compared to $\mathcal{L}_{ce}^{Local}$ (Exp.IV & Exp.VII) in Table 6. While bringing $\mathcal{L}_{ob}$ and $\mathcal{L}_{ce}^{Local}$ together in Exp.VIII is comparable to Exp.VI, implying that the benefits of $\mathcal{L}_{ce}^{Local}$ do not outweigh that of $\mathcal{L}_{ob}$. Also, by comparing Exp.VI and Exp. VIII, we can conclude that the hard one-hot label used by $\mathcal{L}_{ce}^{Local}$ might adversely affect the knowledge transfer that is accomplished by $\mathcal{L}_{ob}$ with the soft labels that are more informative [12]. Besides, Exp.IX shows that even without the normal KD loss $\mathcal{L}_{kd}$, the proposed $\mathcal{L}_{ob}$ and $\mathcal{L}_{rec}$ still achieve decent improvement compared to the baseline results in Exp.I. However, by comparing the results of Exp.II, Exp.IV and Exp.X, $\mathcal{L}_{ce}^{Local}$ alone does not outperforms $\mathcal{L}_{kd}$.

In Eq. (8), student's perspective $\mathcal{A}_s$ is encouraged to be similar to $\mathcal{A}_t$ of teacher by minimizing $\mathcal{L}_{rec}$ between class centers. An alternative way is to apply the pixel-wise alignment between $\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$ in Eq. (8) instead of the rectification on class centers. We believe that mimicking local perspectives is conducive in distilling knowledge from teacher to student since the local observations must be obtained by adaptive perspectives that vary according to the content of individual images, thus the alignment between the adaptive perspectives of teacher and student models may help optimize $\mathcal{L}_{ob}$. The experimental comparison is shown in Table. 7 where it is observed that $\mathcal{L}_{rec}$ yielded with class centers generally leads to a better performance than the pixel-wise counterpart, because the former directly optimizes the adaptive perspectives that are later used in $\mathcal{L}_{ob}$, while the pixel-wise alignment accomplishes the perspective rectification indirectly. Moreover, the combination of 'Center' and 'Pixel' does not bring considerable improvement, manifesting the necessity of rectification on class centers.

Effect of Projectors $\mathcal{P}_t$ and $\mathcal{P}_s$. To generalize our method to different teacher & student models whose output features are with different channels, we use projectors $\mathcal{P}_t$ and $\mathcal{P}_s$ to process the feature maps of teacher and student to the same channels, satisfying the requirement of the similarity calculation in Eq. (8). Otherwise, the perspectives cannot be rectified. Two projectors are only used for training and are simply discarded during inference, boosting student models without structural change.

To show that the improvement of $\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$ is not caused by the two additional projectors, we implement SKD and IFVD on the projected features ($\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$) to compare with the performance obtained from the features without projection ($\boldsymbol{f}_t$ and $\boldsymbol{f}_s$). We note that $\mathcal{P}_t$ is still optimized by $\mathcal{L}_t$ in the following experiments for a fair comparison.

Experimental results are presented in Table 8 where the results of IFVD and SKD implemented on the projected features are comparable to that without projectors as shown in Exp.II & Exp.III and Exp.V & Exp.VI. Besides, the proposed $\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$ are still complementary to the models implemented with IFVD and SKD, proved by Exp.IV and Exp.VII in Table 8.

TABLE 8
Ablation Study on the Validation Sets Of PASCAL-Context and Cityscapes With PSPNet

Exp.	$\mathcal{L}_{kd}$	$\mathcal{L}_{ifv}$	$\mathcal{L}_{skd}$	$\mathcal{P}$	$\mathcal{L}_{ob}$	$\mathcal{L}_{rec}$	Context	City
I	✓	-	-	-	-	-	42.48	74.81
II	✓	✓	-	-	-	-	42.74	74.10
III	✓	✓	-	✓	-	-	43.02	75.21
IV	✓	✓	-	✓	✓	✓	44.05	76.50
V	✓	-	✓	-	-	-	42.53	74.56
VI	✓	-	✓	✓	-	-	42.39	74.13
VII	✓	-	✓	✓	✓	✓	43.98	75.30

Teacher is Built Upon ResNet-101 and Student is With ResNet-18. $\mathcal{L}_{ifv}$ and $\mathcal{L}_{skd}$ are the Intra-Class Feature Variation Distillation and Pair-Wise Distillation of IFVD and SKD. We Reproduce Them According to Their Official Implementations. $\mathcal{P}$ Means $\mathcal{L}_{ifv}$ and $\mathcal{L}_{skd}$ are Applied to the Projected Features $\boldsymbol{f}_{a,t}$ and $\boldsymbol{f}_{a,s}$.

Layer Number of Projector. It is mentioned in Section 5.2 that the projectors for teacher and student models are both implemented by a 2-layers MLP with an intermediate ReLU activation layer. To investigate the influence brought by different layer numbers of MLP, the experimental results are shown in Table 9 from which we can observe that the performance is not sensitive to different layer numbers, and the projectors implemented with 2 fully-connected layers can achieve satisfying results on both two benchmarks.

Necessity of Feature Selection. In Eq. (4), with the ground truth mask, we directly average the features of teacher and student models to yield the semantic anchors $\mathcal{A}_t$ and $\mathcal{A}_s$ respectively, without considering the correctness of predictions of individual feature vectors. Intuitively, pixels with wrong predicted labels might impair the compactness of class centers since their features might be far away from that of the correct ones, thus only incorporating those feature vectors with correct predictions may be helpful to the final performance. To probe the effects of the feature selection mechanism, the results are shown in Table 10 where three additional feature selection schemes are implemented for comparison.

We find that the feature completeness is more important than the correctness. Specifically, merely considering the correctness of the teacher's predictions (Tea-FS) does not significantly undermine the performance since the teacher network has been well trained and thus the auxiliary predictions are generally correct during training, retaining the majority. However, when the correctness of the student is leveraged (i.e., Stu-FS and Tea-Stu-FS), the results are clearly lower than that of the baseline (w/o FS) and Tea-FS, showing the fact that *the feature completeness outweighs the feature correctness* for constructing semantic anchors in our proposed distillation method. The information lost caused by the feature selection should take responsibility for performance deduction of Stu-FS and Tea-Stu-FS, especially on PASCAL-Context where the student model is more likely to make wrong predictions than that on Cityscapes, as manifested by the discrepancy between the mIoU results of their baselines (42.29 and 74.15).

Sensitivity Analysis. Different hyper-parameters may cause performance variation. Thus we conduct sensitivity analysis in Table 11 where the best performance is robust to different values of λ_{rec} and $1/\tau$ within the range of 5-20.

TABLE 9
Ablation Study of Different Layer Numbers for Constructing the Projectors for Teacher and Student Models on the Validation Sets of PASCAL-Context and Cityscapes

Datasets	1	2	3	4
Context	43.66	43.96	43.76	43.62
Cityscapes	75.69	75.68	75.58	75.00

The Intermediate ReLU Layers are Adopted If the Layer Number is Larger Than 1

TABLE 10
Ablation Study of the Effects of Feature Selection (FS) Mechanisms on the Validation Sets of PASCAL-Context and Cityscapes

Datasets	w/o FS	Tea-FS	Stu-FS	Tea-Stu-FS
Context	43.96	43.79	43.01	42.23
Cityscapes	75.68	75.48	75.28	75.01

'Tea-FS': Only Teacher's Local Observations (predictions) $\boldsymbol{p}_{a,t}$ are Used for Selecting the Valid Feature Vectors for Semantic Anchor Generation. 'Stu-FS': Only Student's Local Observations (predictions) $\boldsymbol{p}_{a,s}$ are Used for Determining the Valid Feature Vectors. 'Tea-Stu-FS': Both Teacher's and Student's Local Observations are Adopted, Thus the Valid Feature Vectors for Computing $\mathcal{A}_t$ and $\mathcal{A}_s$ are Filtered by the Correct $\boldsymbol{p}_{a,t}$ and $\boldsymbol{p}_{a,s}$ Respectively.

5.5 Cosine Similarity in APD

In segmentation models, the universal perspective $\mathcal{C}$ applies dot product on the features $\boldsymbol{f}$ yielded by the feature generator $\mathcal{G}$ to produce the observation $\boldsymbol{p} = \text{Softmax}\,(\boldsymbol{f} \cdot \mathcal{C}) = \text{Softmax}\,(|\boldsymbol{f}||\mathcal{C}|\cos(\boldsymbol{f}, \mathcal{C}))$. While, in the proposed APD, the adaptive perspective $\mathcal{A}$ generates observations $\boldsymbol{p}_a$ via cosine similarity: $\boldsymbol{p}_a = \text{Softmax}\,(\cos(\boldsymbol{f}_a, \mathcal{A})/\tau)$. The difference between cosine similarity and dot product is that the former measures the angle between two vectors and the latter takes both the angle and magnitudes into account.

Experimental Results. Both cosine similarity and dot product seem to be feasible for yielding observation, while we find that cosine similarity is more suitable for optimizing the objectives of APD ($\mathcal{L}_{rec}$ and $\mathcal{L}_{ob}$). Results are shown in Table 12. Specifically, by comparing models of "Baseline" and "KD," it can be found that applying cosine similarity to the main universal perspective $\mathcal{C}$ (i.e., Main-Cos) to yield the main predictions $\boldsymbol{p}_s$ and $\boldsymbol{p}_t$ is detrimental to the overall performance. On the other hand, "Main-Cos" also causes performance deduction on the proposed APD, shown by comparing "APD-II" and "APD-III". As for "Adapt-Cos" that can only be adopted by the proposed APD, it is necessary for APD since the performance drops from 43.96 ("APD-II") to 38.04 ("APD-I") if the adaptive perspective

TABLE 11
Sensitivity Analysis With Different Values of λ_{rec} and τ

Values	0.1	1	5	10	20	50	100
λ_{rec}	43.32	43.54	43.70	43.96	43.95	43.71	43.48
$1/\tau$	42.05	42.92	43.79	43.96	43.62	43.36	43.09

Experimental Results are Obtained on PASCAL-Context Val.

TABLE 12
Comparison on the Validation Sets of PASCAL-Context and Cityscapes Between Cosine Similarity and Dot Product for Observation Generation

Method	Main-Cos	Adapt-Cos	Context	City
Baseline-I (Default)		N/A	42.29	74.15
Baseline-II	✓	N/A	41.90	74.25
KD-I (Default)		N/A	42.48	74.81
KD-II	✓	N/A	42.03	73.77
APD-I			38.04	73.29
APD-II (Default)		✓	**43.96**	**75.68**
APD-III	✓	✓	43.24	75.31

The Teacher is PSPNet With ResNet-101 and the Student is PSPNet With ResNet-18. "main-Cos" Means the Main Perspective (classifier) Adopts Cosine Similarity for Prediction and "adapt-Cos" Means the Adaptive One Uses Cosine Similarity. Thus "adapt-Cos" Can Only Be Adopted by APD.

does not exploit the cosine similarity but dot product for yielding the auxiliary observations $\boldsymbol{p}_{a,t}$ and $\boldsymbol{p}_{a,s}$.

In summary, through the experiments in Table 12, we empirically find that the dot-product is more suitable for the universal perspective (i.e., normal classifier) and the cosine similarity is better for the proposed adaptive perspective.

Analysis. The performance discrepancy between "Main-Cos" and "Adapt-Cos" might be related to the formation processes of the universal perspective that is shared by all training images and the adaptive perspective that is created individually. The shared universal perspective approaches to an optimal magnitude by well-fitting the entire training set. The magnitude values of features serve as additional descriptors, revealing more information for individual feature vectors. Therefore, the universal perspective, with well-learned class-wise magnitude, achieves better performance by adopting the dot product. However, the magnitude of the adaptive perspective is determined by the individual feature map and thus the magnitude might be biased towards the feature vectors with large magnitude, causing inappropriate representation for those features with low magnitude. Also, the magnitude values of features belonging to the same category vary in different images due to the varying co-occurrent contextual information. Thus we instead only focus on the semantic relation by adopting cosine similarity to alleviate the issues caused by the magnitude instability of the adaptive perspective that is formed merely based on individual samples.

Besides, it is worth noting that, since the purposes of "Main-Cos" and "Adapt-Cos" are different, we have carefully tuned the values of the scaler τ_m for "Main-Cos" to have a fair comparison with "Adapt-Cos" in Table 12. Specifically, according to the sensitive analysis in Table 11, τ of "Adapt-Cos" is set to 0.1 (i.e., $1/\tau = 10$), while directly applying $\tau_m = 0.1$ to "Main-Cos" significantly worsens the performance as shown in Table 13 where $1/\tau_m = 40$ (i.e., $\tau_m = 0.025$) achieves the best performance. Thus models with "Main-Cos" in Table 12 are implemented with $\tau_m = 0.025$.

5.6 Extensions

Although our method is motivated from the perspective of semantic segmentation tasks, it also generalizes well to the tasks of object detection and instance segmentation. Implementation details and results are presented as follows.

TABLE 13
Different Values of τ_m for the Baseline Model Implemented With "Main-Cos"

$1/\tau_m$	10	20	30	40	50
Baseline-II	40.09	41.56	41.88	**41.90**	41.62
KD-II	40.22	41.79	41.92	**42.03**	41.88
APD-III	40.99	43.15	43.09	**43.24**	43.11

5.6.1 Object Detection

Implementation Details. We use the most popular Faster-RCNN-FPN detector in Detectron2 [40] with different backbones as our strong baselines. We use the standard training policies provided in Detectron2 excepts for the number of GPUs. The original models in Detectron2 are trained using 8 GPUs. The official $1\times$ training policy is to train 90,000 iterations with 16 images per batch. The learning rate is initialized as 0.02 and decayed by 10 at 60,000 and 80,000 iterations. The baseline and other models are trained on 4 GPUs, thus we halve the batch size to 8 and double the total iterations to 180,000. The initial learning rate is 0.01 and it decays by 10 at 120,000 and 160,000 iterations. Our reproduction yields similar baseline performance and costs the same overall GPU time. We use the standard multi-scale training augmentations. The input images are randomly resized to one of the sizes {640, 672, 704, 736, 768, 800} and then images are randomly horizontal fliped with a probability of 0.5. We do NOT use any augmentations during the inference.

We apply the proposed APD to the features after the RoI Align operation. We simulate the scenario in the semantic segmentation tasks and assume every feature vector in the feature map belongs to the class of the corresponding proposal. Then we consider all proposals in a mini-batch as a whole and generate adaptive perspectives in a batch-wise manner.

We reimplement the KD loss proposed by Hinton *et al.* on the logits of the classification branch in the RoI head. The loss weight is also set to 10. We notice that, in object detection, the teacher and student may have different proposals, causing a mismatch between the features after the RoI Align operation as well as the final predicted logits. To address this issue, since only the student's proposals are used for generating the final task losses, we let the teacher network adopt the proposals yielded by the student, thus the teacher's features and logits are aligned with that of the student.

SSD [2] and FGFI [37] are the distillation methods specifically designed for object detection. However, the baseline methods used by them are relatively weaker than the popular ones. So we re-implement SSD and FGFI on our stronger baseline according to the paper or the official code provided by the authors. OFD [11] is another distillation method that improves the student detector by proposing a marginal loss to leverage BN's information to guide the distillation process. Also, we make a comparison with the recent state-of-

TABLE 14
Object Detection Results on COCO 2017 *Val*

Method	Backbone	Schedule	mAP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
Teacher	ResNet101	-	42.04	62.48	45.88	25.22	45.55	54.60
Student	ResNet18	1x	33.26	53.61	35.26	18.96	35.68	43.16
KD [12]	ResNet18	1x	33.68	54.10	35.93	19.65	36.17	43.22
FitNet [28]	ResNet18	1x	34.13	54.16	36.71	18.88	36.50	44.69
SSD [2]	ResNet18	1x	33.89	53.35	36.46	18.59	36.20	44.43
FGFI [37]	ResNet18	1x	34.16	54.43	36.60	18.79	36.57	44.97
OFD [11]	ResNet18	1x	33.36	53.34	35.60	18.61	35.53	43.87
SKD [20]	ResNet18	1x	33.97	54.66	36.62	18.71	36.67	44.14
IFVD [39]	ResNet18	1x	34.20	54.63	36.66	19.16	36.65	44.71
FBKD [51]	ResNet18	1x	35.20	55.20	38.30	18.80	37.90	**47.80**
CWD [32]	ResNet18	1x	35.40	54.60	38.50	18.20	38.20	49.00
Ours	ResNet18	1x	**35.47**	**56.68**	**38.00**	**20.41**	**38.17**	46.14
Student	ResNet18	2x	35.13	55.40	38.20	19.66	37.81	45.13
KD [12]	ResNet18	2x	35.56	55.56	38.58	19.43	38.88	47.09
FitNet [28]	ResNet18	2x	35.64	55.23	38.64	19.52	38.75	46.27
SSD [2]	ResNet18	2x	35.60	55.27	38.60	20.22	37.97	46.95
FGFI [37]	ResNet18	2x	35.93	56.41	38.72	19.86	38.14	46.41
OFD [11]	ResNet18	2x	35.49	55.68	38.17	19.71	38.01	47.01
SKD [20]	ResNet18	2x	35.56	56.04	38.55	19.99	38.18	45.98
IFVD [39]	ResNet18	2x	35.87	56.58	38.59	20.65	38.56	46.20
FBKD [51]	ResNet18	2x	37.00	57.20	39.70	19.90	39.70	50.30
CWD [32]	ResNet18	2x	37.00	56.70	**40.20**	19.40	**40.30**	**50.40**
Ours	ResNet18	2x	**37.08**	**57.99**	40.13	**21.59**	39.88	48.38
Teacher	ResNet50	-	40.22	61.02	43.81	24.16	43.53	51.98
Student	MobileV2	1x	29.47	48.87	30.90	16.33	30.77	38.86
KD [12]	MobileV2	1x	30.13	50.28	31.35	16.69	31.91	39.56
FitNet [28]	MobileV2	1x	30.20	49.80	31.69	16.39	31.64	39.69
SSD [2]	MobileV2	1x	29.96	48.76	31.65	16.51	31.56	39.75
FGFI [37]	MobileV2	1x	30.27	49.87	31.60	17.03	31.82	40.06
OFD [11]	MobileV2	1x	29.73	48.39	31.67	16.26	31.63	39.29
SKD [20]	MobileV2	1x	31.52	50.72	33.35	17.66	33.52	40.75
IFVD [39]	MobileV2	1x	30.67	50.30	32.43	17.09	33.62	38.38
FBKD [51]	MobileV2	1x	32.20	52.80	33.70	18.00	34.50	43.80
CWD [32]	MobileV2	1x	31.20	49.00	33.30	14.70	32.60	**44.00**
Ours	MobileV2	1x	**32.58**	**53.23**	**34.41**	**19.12**	**34.66**	42.35

the-art distillation method named FBKD [51] that adopts the attention guided and non-local distillation on detectors.

Since object detection is also a task of dense prediction, we compare with SKD and IFVD that are originally designed for semantic segmentation, and both SKD and IFVD are re-implemented according to their official implementations. Specifically, we apply the SKD loss on the features after the FPN structure with a 2×2 down-sampling, following its default configurations. However, since class labels are required by IFVD, we apply the IFVD loss on the features after the RoI Align operation. Our code for object detection will also be made publicly available.

Results. We summarize our results on COCO [18] with the Faster-RCNN-FPN [27] detector in Table 14. We re-implement the classic distillation methods KD and FitNet, as well as one recent method FBKD [51] that achieves state-of-the-art performance for distillation in object detection. Moreover, to comprehensively compare with the methods in semantic segmentation, we also apply SKD and IFVD to the object detection task, since both segmentation and detection tasks require structured dense prediction. It can be observed in Table 14 that our method still outperforms most of the other methods by a large margin on the detection task, including SSD and FGFI that are specifically designed for detection. Besides, the proposed method achieves comparable results to the recent state-of-the-art distillation method in object detection (i.e., FBKD). These results further demonstrate the effectiveness and generalization ability of our method.

We present the qualitative comparison between SKD and IFVD on COCO2017 *val* set in Fig. 7 where it is observed that our predictions are generally better than the others.

5.6.2 Instance Segmentation

We further adapt our method to the instance segmentation task on COCO 2017 dataset. Instance segmentation is a more challenging task aiming to segment every object in each image. The Mask-RCNN with FPN in Detectron2 is adopted as our baseline. The training process of instance segmentation is similar to that of object detection, following the standard training policies provided in Detectron 2.

The results are summarized in Table 15 where our method improves the results of instance segmentation task by a large margin, while the other related methods barely improve the baseline performance. The challenging instance

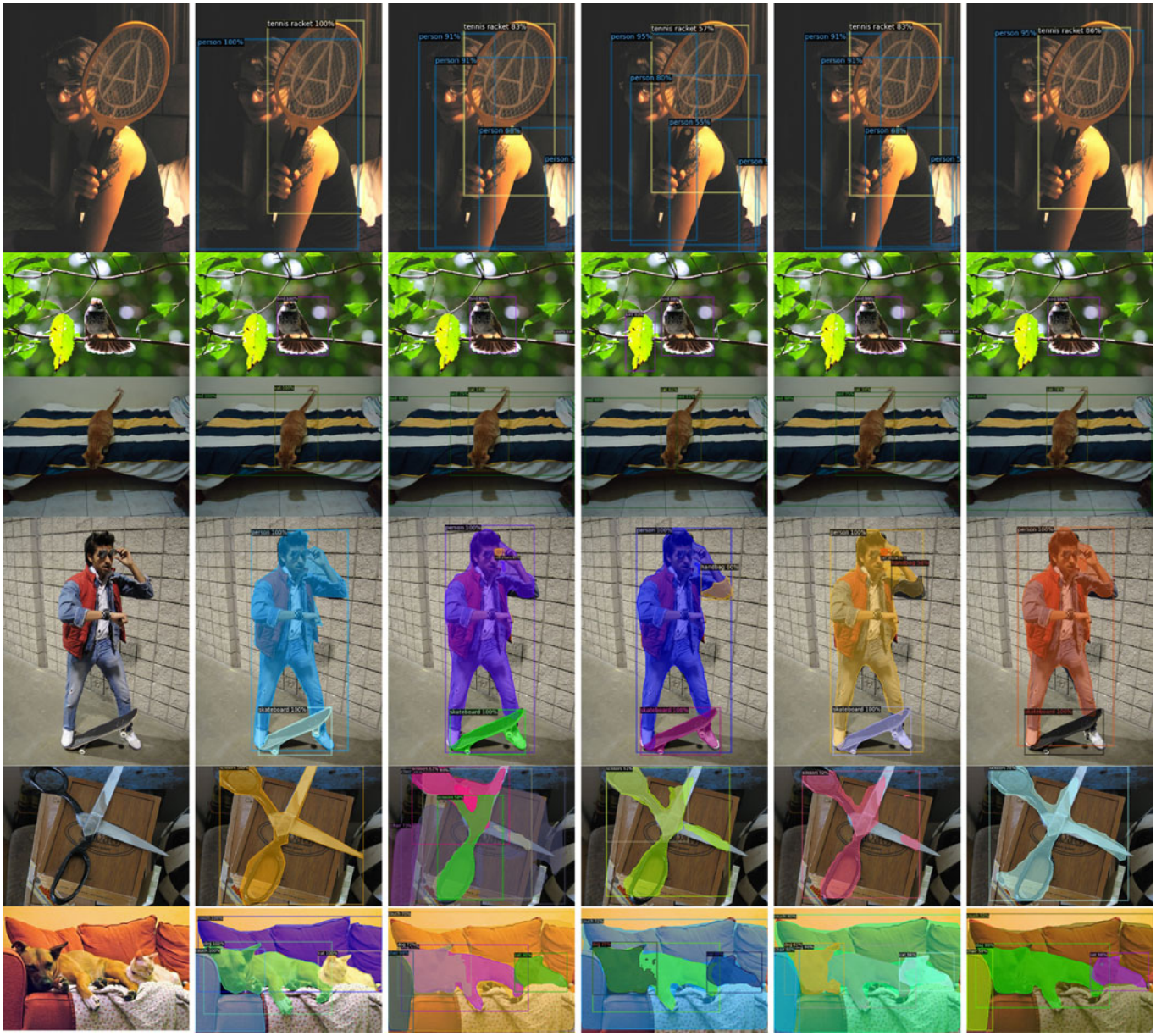


Fig. 7. Visual comparison of object detection (first three rows) and instance segmentation (last three rows) on COCO2017 *val* set.

TABLE 15
Instance Segmentation Results on COCO 2017 *Val*

Method	Backbone	mAP^{box}	mAP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}	AP^{mask}_{s}	AP^{mask}_{m}	AP^{mask}_{l}
Teacher	ResNet101	42.90	38.63	60.45	41.28	19.48	41.33	55.29
Student	ResNet18	33.98	31.25	51.07	33.10	14.18	32.80	45.53
KD [12]	ResNet18	34.53	31.66	51.85	33.59	14.80	33.38	45.73
FitNet [28]	ResNet18	34.69	31.75	51.46	33.82	14.50	33.25	46.76
SSD [2]	ResNet18	34.17	31.10	50.59	32.92	14.14	32.40	45.84
FGFI [37]	ResNet18	34.73	31.85	51.59	33.72	14.95	33.25	46.94
OFD [11]	ResNet18	34.29	31.56	51.02	33.31	14.19	32.73	46.59
SKD [20]	ResNet18	34.53	31.62	51.90	33.54	14.48	33.44	46.10
IFVD [39]	ResNet18	34.59	31.64	52.06	33.38	14.93	33.49	46.34
FBKD [51]	ResNet18	35.40	32.10	52.50	34.00	14.20	34.10	**48.10**
CWD [32]	ResNet18	35.60	32.50	52.00	34.70	15.70	**35.00**	46.10
Ours	ResNet18	**35.90**	**32.84**	**53.70**	**34.71**	**15.77**	34.79	47.81

The Results are Measured in Box mAP and Mask mAP.

segmentation task further demonstrates the superiority of our proposed method. The qualitative comparison on COCO2017 *val* set is shown in Fig. 7.

6 Conclusion

We have presented the proposed Adaptive Perspective Distillation (APD). Different from the previous distillation methods that distill knowledge via pixel-wise predictions obtained by the fixed perspective (i.e., classifier), APD aims at creating adaptive perspectives for individual samples, revealing more details on the encoded feature for helping student models achieve better performance. APD has no structural constraints on the base model and thus can be easily applied to normal semantic segmentation frameworks. APD is also complementary to other existing knowledge distillation methods in segmentation. The extensive comparison with state-of-the-art knowledge distillation methods for semantic segmentation demonstrate the effectiveness and generalization ability of APD.

Acknowledgments

Zhuotao Tian and Pengguang Chen contributed equally.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[2] G. Chen, W. Choi, X. Yu, T. X. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 742–751.

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[4] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 833–851.

[5] MMSegmentation Contributors, "MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark," 2020. [Online]. Available: https://github.com/open-mmlab/mmsegmentation

[6] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[7] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3146–3154.

[8] J. He, Z. Deng, L. Zhou, Y. Wang, and Y. Qiao, "Adaptive pyramid context network for semantic segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7519–7528.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[10] T. He, C. Shen, Z. Tian, D. Gong, C. Sun, and Y. Yan, "Knowledge adaptation for efficient semantic segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 578–587.

[11] B. Heo, J. Kim, S. Yun, H. Park, N. Kwak, and J. Y. Choi, "A comprehensive overhaul of feature distillation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 1921–1930.

[12] Geoffrey E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[13] Q. Hou, L. Zhang, M.-M. Cheng, and J. Feng, "Strip pooling: Rethinking spatial pooling for scene parsing," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4003–4012.

[14] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 603–612.

[15] D. Kothandaraman, A. M. Nambiar, and A. Mittal, "Domain adaptive knowledge distillation for driving scene semantic segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 134–143.

[16] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with gaussian edge potentials," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 109–117.

[17] X. Li, Z. Zhong, J. Wu, Y. Yang, Z. Lin, and H. Liu, "Expectation-maximization attention networks for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 9167–9176.

[18] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[19] W. Liu, A. Rabinovich, and A. C. Berg, "ParseNet: Looking wider to see better," 2015, *arXiv:1506.04579*.

[20] Y. Liu, C. Shu, J. Wang, and C. Shen, "Structured knowledge distillation for dense prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 12, 2020, doi: 10.1109/TPAMI.2020.3001940.

[21] S. Mehta, M. Rastegari, A. Caspi, L. G. Shapiro, and H. Hajishirzi, "EspNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 552–568.

[22] R. Mottaghi *et al.*, "The role of context for object detection and semantic segmentation in the wild," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 891–898.

[23] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 1520–1528.

[24] S. Park and Y. S. Heo, "Knowledge distillation for semantic segmentation using channel and spatial correlations and adaptive cross entropy," *Sensors*, vol. 20, no. 16, 2020, Art. no. 4616.

[25] N. Passalis and A. Tefas, "Probabilistic knowledge transfer for deep representation learning," 2018, *arXiv:1803.10837*.

[26] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[27] S. Ren, K.R. HeGirshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[28] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for thin deep nets," in *Proc. 3rd Int. Conf. Learn. Representations*, 2015.

[29] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.

[30] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.

[31] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[32] C. Shu, Y. Liu, J. Gao, Z. Yan, and C. Shen, "Channel-wise knowledge distillation for dense prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 5291–5300.

[33] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[34] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," in *Proc. 8th Int. Conf. Learn. Representations*, 2020.

[35] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[36] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[37] T. Wang, L. Yuan, X. Zhang, and J. Feng, "Distilling object detectors with fine-grained feature imitation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4933–4942.

[38] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.

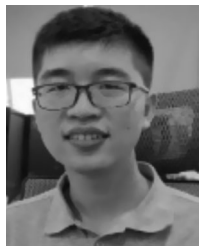
[39] Y. Wang, W. Zhou, T. Jiang, X. Bai, and Y. Xu, "Intra-class feature variation distillation for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 346–362.

[40] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," 2019. [Online]. Available: https://github.com/facebookresearch/detectron2

[41] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1492–1500.
[42] G. Xu, Z. Liu, X. Li, and C. C. Loy, "Knowledge distillation meets self-supervision," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 588–604.
[43] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3684–3692.
[44] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation," 2020, *arXiv:2004.02147*.
[45] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 325–341.
[46] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. 4th Int. Conf. Learn. Representations*, 2016.
[47] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 173–190.
[48] Y. Yuan and J. Wang, "OCNet: Object context network for scene parsing," 2018, *arXiv:1809.00916*.
[49] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in *Proc. 5th Int. Conf. Learn. Representations*, 2017.
[50] H. Zhang *et al.*, "Context encoding for semantic segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7151–7160.
[51] L. Zhang and K. Ma, "Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors," in *Proc. 9th Int. Conf. Learn. Representations*, 2021.
[52] H. Zhao, "SemSeg," 2019. [Online]. Available: https://github.com/hszhao/semseg
[53] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 405–420.
[54] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.
[55] H. Zhao *et al.*, "PSANet: Point-wise spatial attention network for scene parsing," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 267–283.
[56] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ADE20K dataset," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 633–641.

Zhuotao Tian received the BEng degree (Honors) in computer science from the School of Computer Science and Technology, Harbin Institute of Technology (HIT) in 2018. He is currently working toward the PhD degree with the Chinese University of Hong Kong (CUHK), under the supervision of Prof. Jiaya Jia. He serves as a reviewer for IJCV, CVPR, ICCV, ECCV, ICLR, AAAI. His research interests include few-shot learning, semi-supervised learning, semantic segmentation and scene text detection.

Pengguang Chen received the BEng degree in computer science from the Department of Computer Science and Technology, Nanjing University in 2018. He is currently working toward the 3rd year PhD degree with the Chinese University of Hong Kong (CUHK), under the supervision of Prof. Jiaya Jia. He serves as a reviewer for CVPR, ICCV, ECCV. His research interests include neural architecture search, self-supervised learning, knowledge distillation and semantic segmentation.



Xin Lai received the BEng degree in computer science and technology from the Harbin Institute of Technology (HIT) in 2020. He is currently working toward the 1st-year PhD degree with the Computer Science and Engineering Department of the Chinese University of Hong Kong (CUHK). His research interests focus on computer vision and deep learning, especially on semi-supervised learning, few-shot learning and domain generalization techniques.

Li Jiang received the BS degree in computer science and technology from the Harbin Institute of Technology, China in 2017. She is currently working toward the PhD degree with the Department of Computer Science and Engineering at The Chinese University of Hong Kong. She serves as a reviewer for CVPR, ICCV, ECCV. Her research interests include computer vision, semantic/instance segmentation and 3D scene understanding.

Shu Liu (Member, IEEE) received the BS degree from the Huazhong University of Science and Technology and the PhD degree from the Chinese University of Hong Kong. He currently serves as co-founder and technical head with SmartMore. He was the winner of 2017 COCO Instance Segmentation Competition and received the outstanding reviewer of ICCV in 2019. He continuously served as a reviewer for *IEEE Transactions on Pattern Analysis and Machine Intelligence*, CVPR, ICCV, NIPS, ICLR and etc. His research interests lie in deep learning and computer vision.

Hengshuang Zhao (Member, IEEE) received the PhD degree in computer science and engineering from the Chinese University of Hong Kong in 2019. He is a postdoctoral researcher with the University of Oxford. His team won champions of ImageNet Scene Parsing Challenge, LSUN Semantic Segmentation Challenge and WAD Drivable Area Segmentation Challenge at ECCV'16, CVPR'17, and CVPR'18 respectively. He is recognized as outstanding/top reviewers of ICCV'19 and NeurIPS'19. He received the rising star award at the world artificial intelligence conference 2020. His general research interests cover the broad area of computer vision and machine learning, with special emphasis on high-level scene recognition and pixel-level scene understanding.

Bei Yu (Member, IEEE) received the PhD degree from The University of Texas at Austin in 2014. He is currently an associate professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. He has served as TPC chair of ACM/IEEE Workshop on Machine Learning for CAD, and in many journal editorial boards and conference committees. He is the editor of *IEEE Technical Committee on Cyber-Physical Systems* Newsletter. He received seven best paper awards from ASPDAC 2021, ICTAI 2019, Integration, the *VLSI Journal* in 2018, ISPD 2017, SPIE Advanced Lithography Conference 2016, ICCAD 2013, ASPDAC 2012, and six ICCAD/ISPD contest awards.

Ming-Chang Yang (Member, IEEE) received the BS degree from the Department of Computer Science from National Chiao-Tung University, Hsinchu, Taiwan, in 2010 and the master's and PhD degrees from the Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan, in 2012 and 2016, respectively. Currently, he is an assistant professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. His primary research interests include emerging non-volatile memory and storage technologies, memory and storage systems, and next-generation memory/storage architecture designs.

Jiaya Jia (Fellow, IEEE) received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2004 and is currently a full professor with the Department of Computer Science and Engineering at the Chinese University of Hong Kong (CUHK). He assumes the position of associate editor-in-chief for *IEEE Transactionson on Pattern Analysis and Machine Intelligence* and is in the editorial board of *International Journal of Computer Vision (IJCV)*. He continuously served as area chairs for ICCV, CVPR, AAAI, ECCV, and several other conferences for the organization. He was on program committees of major conferences in graphics and computational imaging, including ICCP, SIGGRAPH, and SIGGRAPH Asia.