ViTKD: Feature-based Knowledge Distillation for Vision Transformers

Zhendong Yang[1,2*]   Zhe Li[3]   Ailing Zeng[2]   Zexian Li[4]   Chun Yuan[1†]   Yu Li[2†]

[1]Tsinghua Shenzhen International Graduate School   [2]International Digital Economy Academy (IDEA)
[3]Institute of Automation, Chinese Academy of Sciences   [4]Beihang University

yangzd21@mails.tsinghua.edu.cn   axel.li@outlook.com   lizexian0427@gmail.com

yuanc@sz.tsinghua.edu.cn   {zengailing, liyu}@idea.edu.cn

Abstract

Knowledge Distillation (KD) has been extensively studied as a means to enhance the performance of smaller models in Convolutional Neural Networks (CNNs). Recently, the Vision Transformer (ViT) has demonstrated remarkable success in various computer vision tasks, leading to an increased demand for KD in ViT. However, while logit-based KD has been applied to ViT, other feature-based KD methods for CNNs cannot be directly implemented due to the significant structure gap. In this paper, we conduct an analysis of the properties of different feature layers in ViT to identify a method for feature-based ViT distillation. Our findings reveal that both shallow and deep layers in ViT are equally important for distillation and require distinct distillation strategies. Based on these guidelines, we propose our feature-based method ViTKD, which mimics the shallow layers and generates the deep layer in the teacher. ViTKD leads to consistent and significant improvements in the students. On ImageNet-1K, we achieve performance boosts of 1.64% for DeiT-Tiny, 1.40% for DeiT-Small, and 1.70% for DeiT-Base. Downstream tasks also demonstrate the superiority of ViTKD. Additionally, ViTKD and logit-based KD are complementary and can be applied together directly, further enhancing the student's performance. Specifically, DeiT-T, S, and B achieve accuracies of 77.78%, 83.59%, and 85.41%, respectively, using this combined approach. Code is available at https://github.com/yzd-v/cls_KD.

1. Introduction

Knowledge Distillation (KD) [16] utilizes the output of the teacher model as soft labels to supervise the student model, bringing the lightweight models impressive improvements without extra costs for inference. It has been consistently

---

*This work was done when Zhendong was an intern at IDEA.
†Corresponding authors.

explored for Convolutional Neural Network (CNN) models and applied to many vision tasks successfully, including image classification [3, 19, 40, 47, 50], object detection [2, 17, 33, 41, 44, 48], and semantic segmentation [15, 21, 26, 39].

Recently, Vision Transformer (ViT) [10] has achieved great success in image classification and inspired various transformers [11, 22, 30, 45]. Similar to CNN models, the ViT models generally need more parameters to achieve better performance, making them harder to be deployed. Therefore, boosting the performance of small ViT models using KD is of great value. In this study, we explore *how to apply KD to ViT-based models*. One straightforward approach would be to transfer the KD methods used for CNNs to ViTs. In fact, some fundamental distillation works [16, 25] are structure-independent. For example, the logit-based distillation directly utilizes the model's final logit, enabling it to be used for both CNNs and ViTs. This has been confirmed by DeiT [29] and TinyViT [34].

However, most of the KD methods beyond logit-based distillation are specifically designed for CNN-based models and rely on intermediate features. Due to the vast architectural differences between CNNs and ViTs, these methods are not applicable to ViT-based models. While recent work MiniViT [46] has employed self-attention distillation and hidden-state distillation for vision transformers with various stages, such as Swin-Transformer [22], it is still not viable for models with multiple encoder layers like ViT [10].

Before developing a new feature-based KD method for ViT, we first conduct simple studies with two structure-independent methods FitNet [25] and MGD [42]. We explored distilling knowledge from the last layer like CNN's general distillation, the last 6 layers like PKD [27] for BERT's [9] distillation, and the whole 12 layers of a teacher model (DeiT-Small) to a student model (DeiT-Tiny). Surprisingly, the results for all the intuitive feature distillations shown in Tab. 1 are not satisfactory which consistently degrade the performance of the student (DeiT-Tiny). Specifically, the Top-1 accuracy of the student is just 73.36% when

Distillation setting	**T**: DeiT Small, **S**: DeiT Tiny	
	Top-1 Acc. (%)	Top-5 Acc. (%)
baseline	*74.42*	*92.29*
MGD [42]	74.46 (+0.04)	92.28(-0.01)
Last layer ([25])	73.36 (-1.06)	91.88 (-0.41)
Last 6 layers ([25])	73.76 (-0.66)	92.01 (-0.18)
All 12 layers ([25])	74.24 (-0.18)	92.23 (-0.06)

Table 1. The distillation results of existing feature-based KD methods, including recent MGD and classical FitNet on the last layer, the last 6 layers, and all the 12 layers.

Figure 1. DeiT-Tiny's (upper) and DeiT-Small's (under) attention maps from shallow to deep layers. The X-axis and Y-axis mean the key and query tokens, respectively. The attention maps are obtained by *softmax* and reflects the response between the query and key tokens. The color is brighter with a larger response between the query and key tokens.

distilling on the last layer with FitNet. This distillation on the last layer is widely used for CNN's distillation, but here it causes a 1.06% accuracy drop. When distilling the whole 12 layers, the accuracy drop reduces to 0.18%. This preliminary study suggests that distillation on the shallow layers is as crucial as that on the deep layers in ViT's distillation.

To gain a better understanding of ViT's features, we visualize the attention maps of the student and teacher across various layers, as shown in Fig. 1. For the shallow layers (*e.g.*, layers 0 and 1), both the student and teacher mainly focus on the diagonal, indicating a self-attention pattern. In contrast, for the deeper layers (*e.g.*, layers 10 and 11), there is a greater difference between the attention patterns of the student and teacher. Attention is determined by a few sparse key tokens, and the student and teacher focus on different tokens. This discrepancy makes it challenging for the student to mimic the teacher's final feature directly. Therefore, our findings suggest that different layers may require different knowledge distillation methods.

Accordingly, we perform a series of controlled experiments to examine the effects of different distillation methods and different layers. As a consequence, we propose a nontrivial way for feature-based ViT distillation, named **ViTKD**. ViTKD treats the shallow and deep layers with different distillation methods, which is shown in Fig. 2. We conduct extensive experiments to demonstrate its effectiveness. For instance, we boost the student DeiT-Tiny from 74.42% to 76.06%, DeiT-Small from 80.55% to 81.95% and DeiT-Base from 81.76% to 83.46% on ImageNet-1K. Besides, all loss functions in ViTKD are only calculated on feature maps, so it can be easily combined with the logit-based distillation. With the combination, we can further advance the students' Top-1 accuracy to 77.78%, 83.59% and 85.41%. We also demonstrate the models trained with ViTKD are beneficial to downstream tasks like object detection, human pose estimation and semantic segmentation. In a nutshell, the contributions of this paper are:

• We reveal that the feature-based KD method for CNNs is unsuitable for ViTs. If we align them directly, it will result in a performance drop. Besides, the distillation on the shallow layers is also important for ViT, which differs
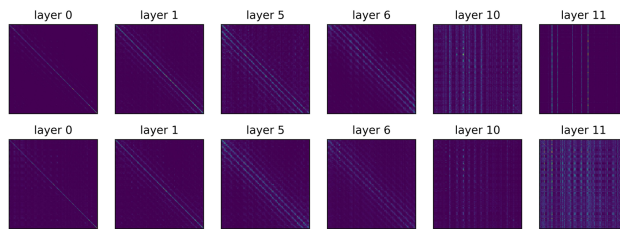
from the conclusion in KD for CNNs.
• We provide an insight that treats different layers with different distillation methods. Based on this insight, we propose a simple and effective KD method named ViTKD.
• We verify the effectiveness of our method via extensive experiments on ImageNet [8], bringing significant performance gains. In addition, we also demonstrate the e superiority of the models trained with ViTKD for various downstream tasks on COCO [20] and ADE20K [49].

2. Related work

2.1. Vision Transformer

Vision Transformer [10] was proposed for image classification. It applies the self-attention [32] architecture to computer vision tasks successfully. DeiT [29] explores the training setting and introduces logit-based KD to ViT with another distillation token. CaiT [30] modifies the architecture and obtains a deeper model. T2T-ViT [45] helps each token to get a better local feature. Swin Transformer [22] utilizes shifted windows to brings greater efficiency by limiting self-attention computation. Vision Transformer has also been applied to other downstream tasks, such as detection [18, 22] and segmentation [1, 37]. However, such models consume many resources for inferring and need to be improved for better application.

2.2. Knowledge Distillation

Knowledge distillation (KD) is a method to improve a compact model without extra time cost for inference. It was proposed by Hinton *et al.* [16], which uses the teacher's output to guide the students. The following works can be divided into the logit-based methods and feature-based methods according to the distillation areas. WSLD [50] analyzes soft labels from a perspective of bias-variance trade-off and distributes different weights for different samples. DKD [47] decouple the logit according to the target class. SRRL [40] utilize the teacher's linear layer to help the student to get
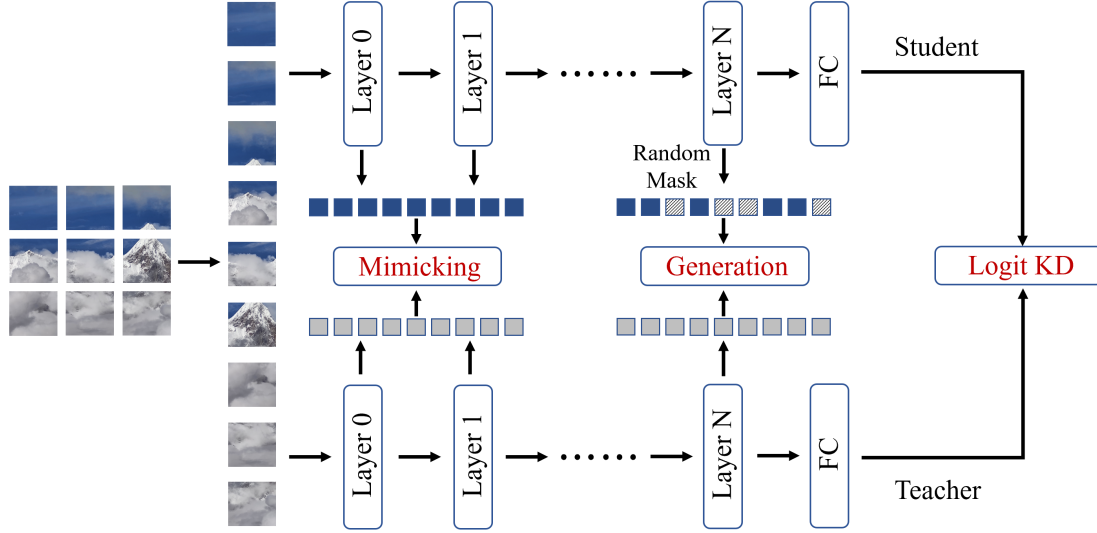
Figure 2. Illustration of the proposed ViTKD. ViTKD is a feature-based distillation method that includes shallow layers' *Mimicking* and deep layer's *Generation*. It can be directly combined with the output logit-based distillation method together.

better features and logit. NKD [43] decomposes and normalizes classic KD, achieving state-of-the-art performance.

Feature-based KD methods calculate the loss on the feature maps. FitNet [25] distills on the intermediate feature directly. RKD [23] transfers the relation from the teacher's feature. CRD [28] introduces contrastive method for feature KD. KR [3] distills the knowledge from the teacher's multi-level features. MGD [42] forces the student to generate the teacher's feature instead of mimicking. However, these feature-based methods are designed for CNNs.

3. Methodology

In this paper, we analyze the difference between ViT's different layers and treat them with different distillation methods. Specifically, for the shallow layers with a small difference, we force such layers to mimic the teacher's corresponding layers, learning how to focus on the tokens themselves. While for the deep layers with a big gap between student and teacher, which have stronger semantic information, we force the student to generate the teacher's feature instead of mimicking directly.

3.1. Mimicking for Shallow Layers

As Fig. 1 shows, the student and teacher's shallow layers have similar attention. Besides, the attention appears mainly on the diagonal. So we force the student to mimic teacher's feature of the first two layers. The analysis of the choice for the shallow layers is in Sec. 5.2. For each sample, we can denote student's and teacher's feature as $\mathcal{F}^S \in \mathcal{R}^{N \times D_S}$ and $\mathcal{F}^T \in \mathcal{R}^{N \times D_T}$, respectively. For the *mimicking* method on shallow layers, we utilize a lin-

ear layer to align the embedding dimension of the student's D_S and the teacher's D_T. The mimicking loss for shallow layers' distillation is as:

$$\mathcal{L}_{lr} = \sum_{i=1}^{N} \sum_{j=1}^{D_T} \left(\mathcal{F}_{i,j}^T - fc(\mathcal{F}^S)_{i,j} \right)^2, \quad (1)$$

where $fc(\cdot)$ is a linear layer to reshape the $\mathcal{F}^S$ to the same dimension as $\mathcal{F}^T$. N, D_T denote the number of patch tokens and the embedding dimension of the teacher's feature.

3.2. Generation for Deep Layers

For the deep layers, the student's and teacher's features become much more different and *mimicking* method fails on it, as shown in Tab. 1. So we try to utilize student's last feature to generate the teacher's last feature, avoiding mimicking directly. The analysis of the choice for the deep layers is in Sec. 5.2. The last feature has the best representation of the original input image. Such feature tokens already contain the information of adjacent tokens to a certain extent. Therefore, we can use partial tokens to generate the complete feature map. This way aims at generating the teacher's feature by student's masked feature, which can help the student achieve a better representation.

We first also use a linear layer to align the student's and teacher's feature embedding dimensions. Then, we set a random mask $Mask \in \mathcal{R}^{N \times 1}$ and use the learnable masked tokens to replace the student's original feature tokens:

$$\hat{\mathcal{F}}_i^S = \begin{cases} masked\ token, & \text{if } r_i < \lambda \\ original\ token, & \text{Otherwise}, \end{cases} \quad (2)$$

$$Mask_i = \begin{cases} 1, & \text{if } r_i < \lambda \\ 0, & \text{Otherwise,} \end{cases} \quad (3)$$

where r_i is a random number uniformly distributed in $[0, 1]$ and $i \in [0, N-1]$ is the coordinates of the tokens dimension. λ controls the masked ratio. The $masked\ token$ is the parameter to learn and will be updated during training.

Finally, we use the new masked feature $\hat{\mathcal{F}}_i^S$ to generate the teacher's full feature through a generative block $\mathcal{G}$, which can be formulated as follows:

$$\mathcal{G}(\hat{\mathcal{F}}^S) \longrightarrow \mathcal{F}^T. \quad (4)$$

For the generative block $\mathcal{G}$, we apply a convolutional projector, which includes two 3×3 conventional layers and one activation layer $ReLU$. Finally, we only calculate the loss of the masked tokens. For the *generation* method for deep layer, we design the distillation loss $\mathcal{L}_{gen}$ as:

$$\mathcal{L}_{gen} = \sum_{i=1}^{N} \sum_{j=1}^{D} Mask_i \big(\mathcal{F}_{i,j}^T - \mathcal{G}(\hat{\mathcal{F}}_{i,j}^S) \big)^2. \quad (5)$$

3.3. ViTKD

Combing the distillation on shallow (**first two**) and deep (**last**) layers, we propose ViTKD, as shown in Fig. 2. When the number of the student's and teacher's layers is different, we also pick the first two and last layers for distillation. To sum up, we train the student model with the total loss:

$$\mathcal{L} = \mathcal{L}_{ori} + \alpha \mathcal{L}_{lr} + \beta \mathcal{L}_{gen}, \quad (6)$$

where $\mathcal{L}_{ori}$ is the original loss for the models. α and β are two hyper-parameters.

4. Experiment

4.1. Settings

Datasets. We explore ViTKD on ImageNet-1K [8], which contains 1000 categories. We use 1.2M images for training and 50k images for testing. For downstream tasks, we evaluate our model on COCO [20] and ADE20K [49].

Implementation details. ViTKD uses α and β to balance the distillation loss in Eq. 6. Another hyper-parameter λ is used to adjust the masked ratio for deep layer distillation in Eq. 2. We adopt the hyper-parameters $\{\alpha = 3 \times 10^{-5}, \beta = 3 \times 10^{-6}, \lambda = 0.5\}$ for all the experiments. Besides, to keep the model to be the same for the feature and logit distillation, we remove the extra distillation token which is used for logit distillation in DeiT. The image resolution for all the experiments is 224×224. The experiments are conducted on 8 Nvidia 3090 GPUs with MMClassification [6] in Pytorch [24]. Unless specified, we evaluate the model with the last epoch. The training setting follows DeiT [29], which is trained with Mixup, Cutmix,

RandAugment and Erasing. We use Adamw as the optimizer with 1024 batch size. The learning rate with warm up decays as cosine. We take larger DeiT [29] and DeiT III [31] models as the teacher to distill lighter DeiT models. The DeiT teacher is trained from scratch on ImageNet-1K, and DeiT III teacher is pre-trained on ImageNet-21K.

4.2. Main Results

Our ViTKD can bring ViTs better performance via feature distillation. To prove this, we first conduct experiments with different teacher-student distillation pairs, as shown in Tab. 2. We compare our ViTKD with two other methods, including FitNet [25] and recent MGD [42]. To further show the effectiveness of ViTKD, we also compare it with original KD [16] and Hard KD [29] from DeiT. As shown in Tab. 2, ViTKD surpasses other feature-based methods significantly and brings remarkable accuracy gains, *e.g.*, the DeiT III-Small [31] teacher boosts the student's Top-1 accuracy from 74.42% to 76.06%. Besides, ViTKD brings comparable improvements as DeiT's Hard KD and even surpasses the classic KD method. Comparing the results between different teachers, we find the student achieves better performance with a stronger teacher, *e.g.*, the student DeiT-Tiny achieves 75.40% and 76.06% Top-1 accuracy with the DeiT-Small and DeiT III-Small teacher, respectively.

We also conduct experiments with more and larger models to show the generalization of ViTKD, as shown in Tab. 3. For larger models such as Swin-T [22], DeiT-S and DeiT-B, ViTKD can also bring them significant improvements, helping it to achieve 81.70%, 81.95% and 83.46%, respectively. Besides, ViTKD is a feature-based method and can be combined with other logit-based methods to further improve the student. This is also one of the feature-based methods' advantages. Therefore, we try to add the state-of-the-art logit-based method NKD [43] to ViTKD, as shown in Fig. 3. In this way, different students all get another significant accuracy improvement, *e.g.*, the student DeiT-Small gets another 1.64% gains and achieves 83.59% Top-1 accuracy with a DeiT III-Base teacher. Surprisingly, the student DeiT-Small is just trained on ImageNet-1K, but its performance surpasses DeiT III-Small's 82.76%, which needs to be pre-trained on ImageNet-21K.

4.3. Downstream Tasks

The model trained with ViTKD achieves significant improvements for classification. To further evaluate the effectiveness of the model with ViTKD, we try to apply it to various downstream tasks. For object detection on COCO [20], we use Mask-RCNN [13] as the detector and follow the setting from ViTDet [18] on detectron2 [35]. For pose estimation, we use ViTPose [38] for the 17 body key points. For segmentation on ADE20K [49], we use UPerNet [36] and train it on MMSegmentation [7].

Teacher	Type	Student	Top-1 Accuracy	Top-5 Accuracy
DeiT-Small (80.69)	baseline	*DeiT-Tiny*	*74.42*	*92.29*
	logit	KD	75.01 (+0.59)	92.52
		Hard KD	**75.10** (+0.68)	**92.55**
	feature	FitNet	73.36 (-1.06)	91.88
		MGD	74.46 (+0.04)	92.28
		ViTKD (Ours)	**75.40** (+0.98)	**92.66**
DeiT III-Small* (82.76)	baseline	*DeiT-Tiny*	*74.42*	*92.29*
	logit	KD	76.01 (+1.59)	93.26
		Hard KD	**76.08** (+1.66)	**93.30**
	feature	FitNet	73.72 (-0.70)	91.88
		MGD	74.53 (+0.07)	92.28
		ViTKD (Ours)	**76.06** (+1.64)	**93.16**

Table 2. We reproduce the results of DeiT with MMClassification on ImageNet-1K. ∗ indicates the teacher is pre-trained on ImageNet-21K. We compare with two other feature-based distillation methods that can be applied to ViTs.

Teacher	Student	*Baseline*	+ ViTKD
Swin-B*	Swin-T	*81.18*	81.70 (+0.52)
DeiT III-B*	DeiT-S	*80.55*	81.95 (+1.40)
DeiT III-L*	DeiT-B	*81.76*	83.46 (+1.70)

Table 3. Results of distilling more and larger models on ImageNet-1K. ∗ indicates the teacher is pre-trained on ImageNet-21K.

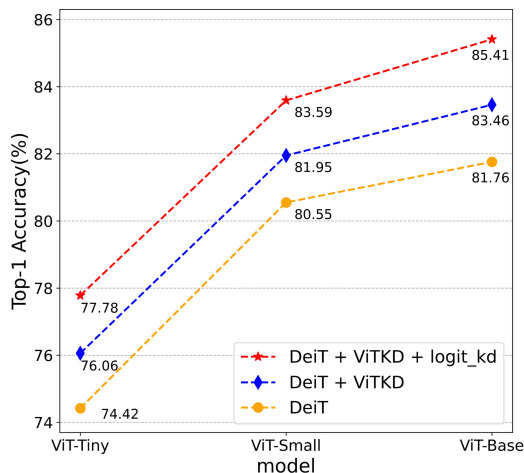| Dataset	Metric	*DeiT-S*	+ ViTKD
ImageNet	Top-1 Acc.	*80.55*	81.95 (+1.40)
COCO	AP^{box}	*45.07*	46.28 (+1.21)
	AP^{mask}	*40.14*	41.05 (+0.91)
	AP^{pose}	*71.80*	72.80 (+1.00)
ADE20K	mIOU	*42.96*	44.94 (+1.98)

Table 4. The downstream tasks results with ViTKD.

Figure 3. Results of combing ViTKD with logit-based KD on ImageNet-1K. The teacher for DeiT-T, S and B is DeiT III-S, B, and L, respectively. The teachers are pre-trained on ImageNet-21K.

As the results in Tab. 4, DeiT-Small trained with ViTKD brings the detector 1.21 Box mAP and 0.91 Mask mAP gains. For human pose estimation, ViTKD brings 1.00 AP gains. For segmentation, ViTKD boosts the mIoU performance from 42.96 to 44.94. The results demonstrate the model trained with ViTKD has not only better performance for classification but also stronger semantic information for various downstream tasks.

5. Analyses

5.1. Student's Attention after Distillation

ViTKD combines mimicking and generation methods for different layers for ViT-based models. In this subsection, we present a visualization and comparison of the average attention maps from the teacher, student, and student with ViTKD to explore how ViTKD influences the student, as shown in Fig. 4. Comparing the attention maps between the original student and teacher, we observe a significant and much smaller difference in the deep and shallow layers, respectively. After distillation with ViTKD, the student's shallow layers have attention maps similar to those of the teacher. However, for the deep layer, as described in Sec.1, the significant gap between the attention maps of the teacher and student makes it difficult for the student to mimic the teacher directly. ViTKD forces the learnable masked tokens to be similar to those of the teacher, instead of using the original tokens of the student. As a result, after our ViTKD's generation, the student's deep features are still dissimilar to those of the teacher. However, ViTKD helps the student generate the teacher's features with its random masked tokens, resulting in the student's deep layer having stronger semantic information. We demonstrate this by using the deep layer for downstream tasks in Sec. 4.3, including detection, human pose estimation, and segmentation.
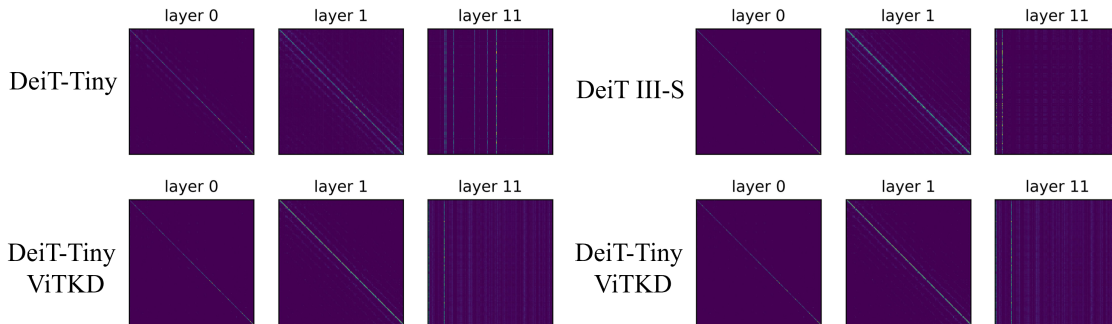
Figure 4. Visualization of the average attention map from the DeiT-Tiny (student), DeiT III-Small (teacher) and DeiT-Tiny after distillation with ViTKD. For a convenient comparison, the two visualizations in the second row are both DeiT-Tiny with ViTKD.

	Layers	Top-1 Acc. (%)
	0	75.01
	0,1	75.12
shallow	5,6	75.01
	0,1,6	75.14
	0,1,2,3,4,5,6	75.16
deep	11	74.72
	10,11	74.70

Table 5. The effects of distillation on different layers.

Losses	DeiT-Tiny (Student)			
$\mathcal{L}_{lr}$	-	✓	-	✓
$\mathcal{L}_{gen}$	-	-	✓	✓
DeiT-S (T)	74.42	75.12	74.72	**75.40**
DeiT III-S* (T)	74.42	75.31	74.79	**76.06**

Table 6. Ablation study results of the losses of shallow layers' *Mimicking* and deep layer's *Generation*. ∗ means the teacher is pre-trained on ImageNet-21K.

5.2. Different Layers for Distillation

As shown in Fig. 1, the attention distributions of the middle layers are also similar to that of shallow layers. Accordingly, we first explore the effects of distillation on such layers in Tab. 5. In general, distillation on either the shallow or middle layers can benefit the student. Besides comparing the improvements from different layers, we find that the knowledge from the shallow layers is much more helpful than that from the middle layer for distillation. Furthermore, when distilling the whole first seven layers together, the accuracy improvement is just 0.04% above the first two layers. Considering the trade-offs between time consumption and performance, we choose the first two layers for shallow distillation eventually. While for the deep layer's distillation, calculating distillation loss on the last layer takes less time but brings more improvement.

5.3. Effects of Shallow and Deep Layers' Losses

As described in the method, we distill the shallow layers and deep layers by mimicking and generation, respectively. In this subsection, we conduct experiments of *Mimicking* loss $\mathcal{L}_{lr}$ and *Generation* loss $\mathcal{L}_{gen}$ to investigate their influences on the student with DeiT-Tiny. As shown in Tab. 6, both the knowledge from shallow and deep layers are helpful for the student. When just applying a single loss, the $\mathcal{L}_{lr}$ on shallow layers benefits the student much more than $\mathcal{L}_{gen}$ on the deep layer. This phenomenon shows that incipient attention knowledge really matters for ViT's feature distillation, which is completely different from the CNN-based model's feature distillation. Furthermore, these two losses are complementary to each other. For example, when combing $\mathcal{L}_{lr}$ and $\mathcal{L}_{gen}$ together, the student with a DeiT III-Small teacher achieve 76.06% Top-1 Accuracy, which is much higher than just applying $\mathcal{L}_{lr}$'s 75.31% and $\mathcal{L}_{gen}$'s 74.79%.

5.4. Different Generative Block for Deep Distillation

For *generation*, we randomly mask the student's tokens and utilize a generative block to restore the feature. In this section, we discuss the effects of different generative blocks, including cross-attention block [5], self-attention block [14], and convolutional projector [42]. We use two teachers to distill the student DeiT-Tiny on ImageNet-1K. As our results shown in Tab. 7, the cross-attention blocks impair the student's performance noticeably. Instead, the self-attention and convolutional block can improve the accuracy of the student. The largest gains are obtained by using the convolutional projector as the generative block.

5.5. Different Alignments for Shallow Distillation

When using *mimicking*, we align the embedding dimensions of the student and teacher by a linear layer. Here we try another alignment way called correlation matrix to describe the response among different tokens and force the student to learn the correlation matrix of the teacher. In this case, we do not need the adaption layer to align the embedding

		DeiT-Small (80.69)	DeiT III-Small* (82.76)
	Teacher	DeiT-Small (80.69)	DeiT III-Small* (82.76)
	Student	DeiT-Tiny (74.42)	DeiT-Tiny (74.42)
Generation	Cross-attention	73.77 (-0.65)	73.98 (-0.44)
	Self-attention	74.61 (+0.19)	74.65 (+0.23)
	Conv. projector	**74.72** (+0.30)	**74.79** (+0.37)

Table 7. The comparisons of different generative blocks for deep layer's distillation on ImageNet-1K.

		DeiT III-S* (82.76)	DeiT-B (81.76)
	Teacher	DeiT III-S* (82.76)	DeiT-B (81.76)
	Student	DeiT-T (74.42)	DeiT-T (74.42)
Mimicking	Linear layer	**75.31** (+0.89)	**75.15** (+0.73)
	Correlation matrix	74.94 (+0.52)	75.01 (+0.59)

Table 8. The comparisons of different alignments for shallow layers' distillation on ImageNet-1K.

Teacher	DeiT-S (80.69)	DeiT III-S (82.76)
MHA	75.06	75.02
FFN	**75.12**	**75.31**

Table 9. Different modules for shallow layers.

λ	*baseline*	0	0.3	0.5	0.7	0.9
Acc.	*74.42*	75.77	75.86	76.06	75.94	75.83

Table 10. Sensitivity study of hyper-parameter λ, the masked ratio in deep layer's distillation.

dimension. The correlation matrix can be calculated as:

$$\mathcal{M} = \frac{\mathcal{F}\mathcal{F}^{Tr}}{\sqrt{D}}, \tag{7}$$

where $\mathcal{F} \in \mathcal{R}^{N \times D}$ denotes the student or teacher's feature. D is embedding dimension and Tr denotes transposition, so $\mathcal{F}^{Tr} \in \mathcal{R}^{D \times N}$. In this case, the student's and teacher's relation matrices have the same shape $\mathcal{M} \in \mathcal{R}^{N \times N}$ and describe the response between different tokens.

We pick the first two layers for distillation by mimicking in Tab. 8. Both alignments for transferring the knowledge from the shallow layer by directly mimicking make great progress. Mimicking by 'linear layer' performs better than the 'correlation matrix' way. When the teacher performs better, the 'linear layer' way benefits the student much more than the 'correlation matrix' way.

5.6. Different Modules for Distillation

ViTs are built by stacking several encoder layers. Each encoder layer consists of a multi-head attention (MHA) module and a feed-forward network (FFN) module. We further conduct experiments on the shallow layers of DeiT-tiny to explore how to choose the modules. The results in Tab. 9 demonstrate that distilling on the MHA-out or FFN-out fea-

ture both benefit the student and the knowledge from FFN-out feature is better than that from MHA-out feature.

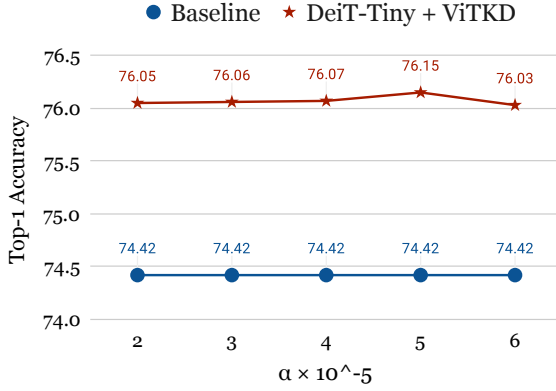## 5.7. Sensitivity Study of Hyper-parameters

We use two hyper-parameters α and β in Eq. 6 to balance the shallow layer's loss $\mathcal{L}_{lr}$ and the deep layer's loss $\mathcal{L}_{gen}$, respectively. To explore the sensitivity of them, we conduct experiments by adopting DeiT III-Small to distill DeiT-Tiny. As shown in Fig. 5a and Fig. 5b, ViTKD is not sensitive to α or β which is just used for balancing the loss. Specifically, when α varies from 2 to 6, the student's worst accuracy is 76.03%, which is just 0.12% lower than the highest accuracy. Besides, it is still 1.61% higher than the baseline, demonstrating ViTKD is not sensitive to the hyper-parameters for loss scale. As for the masked ratio λ in Eq. 2, it is used to control the ratio of masked tokens. We conduct experiments to explore it, as shown in Tab. 10. When λ is 0, it means there are no masked tokens and the performance is poor. When it is too large, *e.g.*, 0.9, the left tokens are too poor to generate the teacher's feature and the performance is also affected. Besides, λ is applied just for deep distillation. So the influence of ViTKD is limited. The lowest performance is still 1.35% higher than the baseline.

## 5.8. Different Methods for Different Layers

Our ViTKD applies mimicking way to shallow layers and generation way to deep layer, respectively. Here we try to apply the mimicking or generation way to the shallow and deep layers together. As shown in Tab. 11, both mimicking and generation ways bring the student limited improvements. Our ViTKD achieves significant results and boosts the student from 74.42% to 76.06%.

5.9. Comparison with More Methods

We also compare our ViTKD with more methods which include both feature and logits distillation for ViT-liked models, as shown in Tab. 12. Our ViTKD achieves comparable results via feature distillation. When combining the

(a) α for the shallow layers' loss $\mathcal{L}_{lr}$
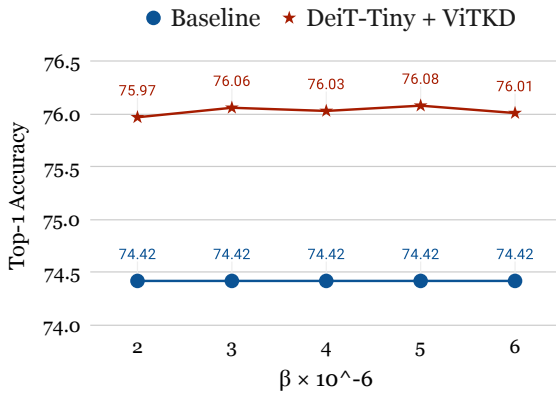
(b) β for the deep layers' loss $\mathcal{L}_{gen}$

Figure 5. Sensitivity study of hyper-parameters α (a) and β (b).

Methods	Layers	Top-1 Acc. (%)
baseline	-	*74.42*
Mimicking	shallow	75.31
	deep	73.72
	shallow + deep	74.65
Generation	shallow	74.69
	deep	74.79
	shallow + deep	75.02
ViTKD	shallow + deep	76.06

Table 11. Results of applying different methods to different layers.

logit-based kd together, the performance gets another significant improvement. The results demonstrate our ViTKD surpasses other methods in a fair setting.

5.10. Time Cost of ViTKD

We report the time for a training epoch on 8 Nvidia 3090 GPUs in Tab. 13. We test DeiT-Tiny with a DeiT-Small teacher. The logit-based method also needs to obtain intermediate features before final logits. Therefore, it's reasonable to utilize features and logits together for distillation.

Methods	Type	Top-1 Acc. (%)
DearKD [4]	feature + logit	81.5
Manifold [12]	feature + logit	82.2
ViTKD	feature	82.0
ViTKD + logit-kd	feature + logit	83.6

Table 12. The comparisons of more methods on ImageNet-1k.

Methods	Type	Time (min)	Acc. (%)
NKD	logit	7.6	75.48
ViTKD	feature	7.8	75.40
ViTKD + NKD	feature + logit	7.9	76.18

Table 13. The time coset of ViTKD.

Methods	Type	COCO Det mAP
baseline	-	*47.23*
ViTKD	feature	48.13
ViTKD + logit-kd	feature + logit	48.83

Table 14. The downstream tasks of larger models DeiT-Base.

For both settings, ViTKD's time consumption is similar to the logit-based method.

5.11. Downstream Tasks of Larger Models

In Sec. 4.3, we report the downstream results of DeiT-Small. Here we add the object detection results of a larger model DeiT-Base with our distillation method, as shown in Tab. 14. The student also achieves consistent improvements and can be further improved with logit-based KD together.

6. Conclusion

In this paper, we propose a feature-based distillation method for ViTs. We point out that distillation on the shallow layers is also important for ViTs and different layers need different distillation methods. Based on this insight, we propose a simple and effective method ViTKD, which includes the distillation on shallow layers via *mimicking* and deep layers via *generation*. ViTKD brings significant improvements on image classification and also benefits the downstream tasks. Besides, ViTKD is truly a feature-based method and can be easily combined with other logit-based methods to further improve the performance. ViTKD attempts to apply feature-based KD for ViTs. We believe the insight that treating different layers with different methods can still be further explored and be extended to other tasks.

References

[1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. In *International Conference on Learning Representations*, 2022. 2

[2] Weihan Cao, Yifan Zhang, Jianfei Gao, Anda Cheng, Ke Cheng, and Jian Cheng. Pkd: General distillation framework for object detectors via pearson correlation coefficient. *Advances in Neural Information Processing Systems*, 2022. 1

[3] 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, 3

[4] Xianing Chen, Qiong Cao, Yujie Zhong, Jing Zhang, Shenghua Gao, and Dacheng Tao. Dearkd: data-efficient early knowledge distillation for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12052–12062, 2022. 8

[5] Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. Context autoencoder for self-supervised representation learning. *arXiv preprint arXiv:2202.03026*, 2022. 6

[6] MMClassification Contributors. Openmmlab's image classification toolbox and benchmark. https://github.com/open-mmlab/mmclassification, 2020. 4

[7] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 4

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2, 4

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019. 1

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 1, 2

[11] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, pages 15908–15919, 2021. 1

[12] Zhiwei Hao, Jianyuan Guo, Ding Jia, Kai Han, Yehui Tang, Chao Zhang, Han Hu, and Yunhe Wang. Learning efficient vision transformers via fine-grained manifold distillation. *Advances in Neural Information Processing Systems*, pages 9164–9175, 2022. 8

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017. 4

[14] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 6

[15] 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. 1

[16] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 1, 2, 4

[17] Gang Li, Xiang Li, Yujie Wang, Shanshan Zhang, Yichao Wu, and Ding Liang. Knowledge distillation for object detection via rank mimicking and prediction-guided feature imitation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1306–1313, 2022. 1

[18] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European Conference on Computer Vision*, pages 280–296, 2022. 2, 4

[19] Sihao Lin, Hongwei Xie, Bing Wang, Kaicheng Yu, Xiaojun Chang, Xiaodan Liang, and Gang Wang. Knowledge distillation via the target-aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10915–10924, 2022. 1

[20] 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, 2014. 2, 4

[21] 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. 1

[22] 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. 1, 2, 4

[23] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019. 3

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019. 4

[25] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015. 1, 2, 3, 4

[26] Changyong Shu, Yifan Liu, Jianfei Gao, Zheng Yan, and Chunhua Shen. Channel-wise knowledge distillation for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5311–5320, 2021. 1

[27] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. In *EMNLP/IJCNLP (1)*, 2019. 1

[28] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2019. 3

[29] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357, 2021. 1, 2, 4

[30] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 32–42, 2021. 1, 2

[31] Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European Conference on Computer Vision*, pages 516–533, 2022. 4

[32] 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*, 2017. 2

[33] Luting Wang, Xiaojie Li, Yue Liao, Zeren Jiang, Jianlong Wu, Fei Wang, Chen Qian, and Si Liu. Head: Hetero-assists distillation for heterogeneous object detectors. In *European Conference on Computer Vision*, pages 314–331, 2022. 1

[34] Kan Wu, Jinnian Zhang, Houwen Peng, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. Tinyvit: Fast pretraining distillation for small vision transformers. In *European Conference on Computer Vision*, pages 68–85, 2022. 1

[35] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 4

[36] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision*, pages 418–434, 2018. 4

[37] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, pages 12077–12090, 2021. 2

[38] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. In *Advances in Neural Information Processing Systems*, 2022. 4

[39] Chuanguang Yang, Helong Zhou, Zhulin An, Xue Jiang, Yongjun Xu, and Qian Zhang. Cross-image relational knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12319–12328, 2022. 1

[40] Jing Yang, Brais Martinez, Adrian Bulat, and Georgios Tzimiropoulos. Knowledge distillation via softmax regression representation learning. In *International Conference on Learning Representations*, 2020. 1, 2

[41] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4643–4652, 2022. 1

[42] Zhendong Yang, Zhe Li, Mingqi Shao, Dachuan Shi, Zehuan Yuan, and Chun Yuan. Masked generative distillation. In *European Conference on Computer Vision*, pages 53–69, 2022. 1, 2, 3, 4, 6

[43] Zhendong Yang, Ailing Zeng, Zhe Li, Tianke Zhang, Chun Yuan, and Yu Li. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17185–17194, 2023. 3, 4

[44] Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4210–4220, 2023. 1

[45] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 558–567, 2021. 1, 2

[46] Jinnian Zhang, Houwen Peng, Kan Wu, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. Minivit: Compressing vision transformers with weight multiplexing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12145–12154, 2022. 1

[47] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11953–11962, 2022. 1, 2

[48] Zhaohui Zheng, Rongguang Ye, Ping Wang, Dongwei Ren, Wangmeng Zuo, Qibin Hou, and Ming-Ming Cheng. Localization distillation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9407–9416, 2022. 1

[49] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 2, 4

[50] Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. Rethinking soft labels for knowledge distillation: A bias–variance tradeoff perspective. In *International Conference on Learning Representations*, 2020. 1, 2