Self-Regulated Feature Learning via Teacher-free Feature Distillation

Lujun Li[0000−0002−4329−2707]

Chinese Academy of Sciences, Beijing, China
lilujunai@gmail.com

Abstract Knowledge distillation conditioned on intermediate feature representations always leads to significant performance improvements. Conventional feature distillation framework demands extra selecting/training budgets of teachers and complex transformations to align the features between teacher-student models. To address the problem, we analyze teacher roles in feature distillation and have an intriguing observation: additional teacher architectures are not always necessary. Then we propose Tf-FD, a simple yet effective **T**eacher-***f***ree **F**eature **D**istillation framework, reusing channel-wise and layer-wise meaningful features within the student to provide teacher-like knowledge without an additional model. In particular, our framework is subdivided into intra-layer and inter-layer distillation. The intra-layer Tf-FD performs feature salience ranking and transfers the knowledge from salient feature to redundant feature within the same layer. For inter-layer Tf-FD, we deal with distilling high-level semantic knowledge embedded in the deeper layer representations to guide the training of shallow layers. Benefiting from the small gap between these self-features, Tf-FD simply needs to optimize extra feature mimicking losses without complex transformations. Furthermore, we provide insightful discussions to shed light on Tf-FD from feature regularization perspectives. Our experiments conducted on classification and object detection tasks demonstrate that our technique achieves state-of-the-art results on different models with fast training speeds. Code is available at https://lilujunai.github.io/Teacher-free-Distillation/.

Keywords: Feature Regularization, Knowledge Distillation

1 Introduction

Despite the tremendous success of deep learning in various tasks [2,11,24,53], it is still difficult to employ deep neural networks to solve real-world problems because of the limitations of calculation and memory assets. To alleviate this issue, there have been several efforts [13,28,65,66] to drive down the computational cost of deep neural networks, and Knowledge Distillation (KD) [16] is one of the examples. KD is an effective training process that achieves a higher precision-efficiency trade-off at runtime by transferring the knowledge learnt by a high-capacity teacher model to a low-capacity student model.

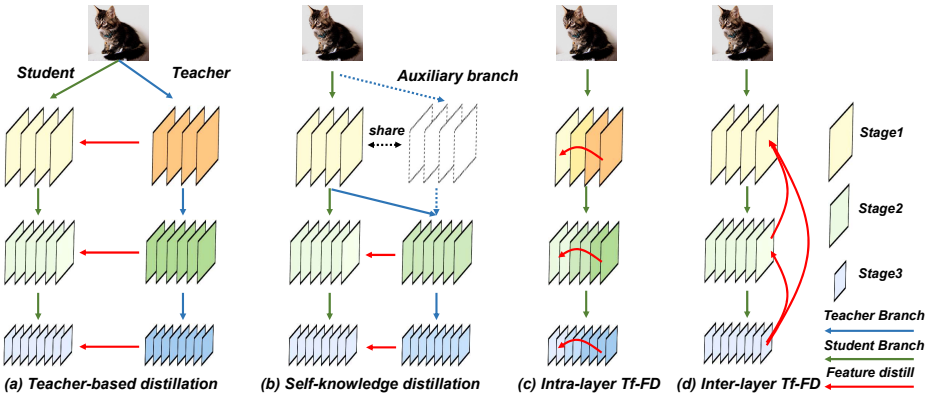

Figure 1: Comparison of teacher-based distillation (a), self-knowledge distillation (b), our intra-layer Tf-FD (c) and inter-layer Tf-FD (d). We use ResNet-20 as a student model on CIFAR-100. Different teacher architectures, pre-trained ResNet-110/ResNet-20, online ResNet-110/ResNet-20 in (a) and auxiliary branch of ResNet-20 in (b) improve baseline by 1.66%, 1.36%, 1.88%, 1.42% and 0.98% gains for top-1 accuracy, respectively. Our intra-layer and inter-layer Tf-FD obtain 1.42% and 1.25% gains.

The original KD [16] uses the logit outcomes of the teacher network as knowledge. For further exploiting the knowledge, the feature distillation methods [44,61] enable student to imitate the intermediate feature of the teacher in order to further utilize its knowledge. Subsequent works [1,15,20,22,52,61] focus on extracting and matching informative knowledge conditioned on the feature representations of a pre-defined teacher model. However, the pipeline of these traditional teacher-student learning suffers from three critical problems: (a) It requires substantial efforts and experiments to find proper teacher models, especially for large student models. (b) Training teacher model needs extra training resources, which brings heavy burdens for applications. (c) Teacher-based distillation methods always employ complex feature transformations (*e.g.*, encoder-decoder [22]) or matching strategies [5] to perform better semantic alignment due to the feature gap. These issues limit the extensive application of feature distillation.

A question naturally arises: is an extra teacher model necessary for feature distillation? To make it clear, we investigate behaviours of teacher models in distillation works, including teacher-based distillation methods [16, 34] and self-knowledge distillation (self-KD) methods [21, 25, 31, 41, 62]. As demonstrated in Figure 1, for the teacher-based techniques, another high-capability model is typically selected as the teacher model. Meanwhile, self-KD methods obtain the teacher model by constructing auxiliary branches, which share the shallow layer with the student model. Therefore, in these two frameworks, the teacher-student model can be regarded as a super-network [8, 18] with a teacher branch and a student branch. We evaluate different types of models as a teacher branch to investigate their effect for feature distillation. The results (see the captions in

Figure 1) indicate that all these various teacher modules can bring considerable distillation gains. That is to say, for the super-network, features located in different branches can play the role of the teacher model for other sub-networks. This observation encourages us to explore whether features from sub-networks in other dimensions (*e.g.*, depth and width) can similarly produce distillation boosts. Consequently, we discard the teacher branch and employ the features located in different layers and channels for distillation (see Figure 1 (c) and (d)). Magically, such a completely teacher-free feature distillation approach also yields significant performance gains.

Inspired by the above observations, we present a simple yet effective **T**eacher-***f***ree **F**eature **D**istillation (Tf-FD) framework. Different from the current teacher-student framework, our approach takes supervision from the intermediate features within the student network itself to perform distillation without additional teacher models. Specially, intra-layer Tf-FD and inter-layer Tf-FD are developed in our framework, respectively. For intra-layer Tf-FD, we first reorganize intra-layer features depending on their salience, which is calculated based on the l_p-norm of each feature. These larger l_p-norm features contain more meaningful knowledge [28]. Then, intra-layer Tf-FD allows salient features to distill redundant ones. The inter-layer Tf-FD leverages the fact that deeper layers contain rich contextual information [6] and achieves the knowledge distillation chain from deep to shallow layers by minimizing self-training losses. The merits of Tf-FD lie in three-fold. First, it proposes a simple distillation pipeline that can successfully broaden the usage of distillations without additional teacher seeking and training costs. Second, Tf-FD only needs to employ simple l_2 distances for the feature mimicking loss, which benefits from fast training speed. Third, self-feature knowledge mined by Tf-FD is orthogonal to knowledge from other models and self-logits. Thus, Tf-FD could naturally combine with teacher-based KD, and logit regularizes to obtain additional gains. We further shed light on the Tf-FD from a regularization perspective. In principle, our Tf-FD plays the role of a new regularizer via self-features, which provide semantic disturbance to obtain significant performance gains. Thus, Tf-FD outperforms other regularizers in terms of enhancing the feature consistency of the lightweight model.

Comprehensive experiments are implemented on a variety of deep models and datasets. For performance improvement, our approach surpasses previous regularization techniques with $0.75\% \sim 0.99\%$ obvious margins and yields $1.07\% \sim 1.56\%$ gains than baseline on CIFAR-100. On the large-scale ImageNet dataset, our approach still achieves 0.71% gains, which outperforms other training techniques. For training efficiency, Tf-FD achieves at least $3\times$ faster training speed than teacher-based KDs. Moreover, Tf-FD with orthogonal logits KD on the outputs surpasses the recent contrastive training distillation (*e.g.*, CRD [50]). On downstream tasks, Tf-FD improves the AP by 0.99 on the Faster R-CNN detector on the MS-COCO dataset, demonstrating the generality of our approach.

In conclusion, we make the following major contributions in this paper:

- By analyzing and exploring teacher models in feature distillation, we point out that the distillation process on intermediate features does not rely on

additional teacher architectures. This motivates us to propose a novel Teacher-free Feature Distillation (Tf-FD) framework.

- Tf-FD explores new distillation schemas where the student learns from the salient feature maps in the same layer (intra-layer Tf-FD) and the deeper layers representations (inter-layer Tf-FD) without any additional teacher model and complex transformations. As a result, Tf-FD merits faster training speeds, superior accuracy gains, and extensive generalizability.
- We further discuss the relationship between Tf-FD and feature regularization. Tf-FD implicitly utilizes self-features as regularization distortion by optimizing the distillation loss. We hope this discussion could facilitate future research for feature distillation works to some extent.

2 Related Work

We summarize the current distillation and regularization works in this part.

Feature distillation vs feature regularization. Knowledge distillation use logits [3, 16] or feature knowledge [44] from a high-capacity teacher to drive the student's training. The intermediate features of a network contain extensive spatial and structural information regarding image content [7]. Accordingly, feature distillation methods [7, 44, 57, 61] are emphasized in designed to convince the student model to simulate the teacher model's feature representations. As feature maps from different layers of the student and teacher networks typically have non-matching dimensions (*e.g.*, widths, heights, and channels), existing feature distillation methods adopt various transformations to match their dimensions and different distance metrics to measure differences. FitNets [44], for instance, uses l_2-loss to emulate the middle features of teacher-student networks, and AT [61] applies feature distillation on the attention map. However, choosing a suitable teacher model for feature distillation is not easy. In sharp contrast to these methods, Tf-FD is a complete teacher-free feature distillation without any extra structure. It opens up a new avenue for distillation design conditioned on the intermediate representations. Feature regularization methods [10, 37, 49] can effectively prevent neural network overfitting by injecting noise into feature space. For instance, DropBlock [10] randomly removes some consecutive portions of a feature map, while SpatialDropout [51] randomly abandons the entire channels. However, these methods depend on some unique strategies to avoid severe semantic damage, which will be detrimental to the performance of the CNNs [49]. Our Tf-FD can be regarded as a feature regularization method and uses self-features as a noise, which contains more semantic information than random masks. Tf-FD develops the connections between feature regularization and distillation from this point.

Self-knowledge distillation vs teacher-free distillation. Some self-knowledge distillation frameworks [21, 25, 41, 62] generate extra auxiliary branches [25, 29, 47], classifiers [41] and FPN [21] to present online logits distillation. Nevertheless, these methods are not teacher-free distillation methods. They necessitate careful designing and training of auxiliary structures, which

may enable student network optimization challenging [19]. Also, these methods are teacher-based and mainly work on the outputted logits, not the intermediate features. Other Self-KDs [27, 59] use additional data views as teachers/peers but may lose helpful information in the augmentation process on some tasks (*e.g.*, object detection). Our Tf-FD does not require any additional teacher structures or additional forward and backward passes. SAD [17] adds attention supervision based on the nature of the feature map on specific lane line detection task. However, it also introduces additional parameters on feature alignment and is not present as a general approach for classification and object detection. Our inter-layer Tf-FD performs dense cross-layer distillation, but only residual supervision exists with SAD [17]. Recent methods [58, 63] let the student model learn from the manually designed smooth distribution on the outputted logits like label smoothing [48]. Tf-FD mainly acts on the intermediate features, not the outputs, and can be well combined with these methods, further expanding the family of teacher-free distillations.

3 Teacher-free Feature Distillation

In this section, we first review feature distillation methods with a general formulation in § 3.1. Then, the formulation and insights of our Teacher-free Feature Distillation (Tf-FD) are presented in § 3.2. Finally, we discuss the relationship between Tf-FD and feature regularization in § 3.3.

3.1 Revisiting Conventional Feature Distillation

We first briefly review the fundamental concept of knowledge distillation within the feature level to further comprehend our methodology. Conventional feature distillation methods [44, 61], which explicitly optimize the feature distillation loss, promote the student model to learn the feature spaces of the teacher. Given that x stands for the training data and $\mathcal{H}$ for a collection of layer location pairs for feature distillation. The generic objective function for a target student model S with features $\boldsymbol{\psi_S}$ and its teacher model T with features $\boldsymbol{\psi_T}$ is defined as:

$$\mathcal{L}_{\mathrm{S}} = \mathcal{L}_{\mathrm{CE}}\big(\theta_S, x\big) + \mu \sum_{h \in \mathcal{H}} \mathcal{D}_f\big(T_s^h(\boldsymbol{\psi_S}), T_t^h(\boldsymbol{\psi_T})\big), \tag{1}$$

where θ_S denotes the parameters of the student model. The student and teacher transformations, T_s and T_t are used to align the feature channel and spatial dimensions. The distance function quantifying the difference of intermediate features is $\mathcal{D}_f(\cdot)$. The weighting factor called μ is used to balance loss terms.

3.2 Formulation of Teacher-free Feature Distillation

Our Tf-FD aims to realize feature distillation via optimizing self-training losses to make the design as general as possible. As illustrated in Figure 2, merely given

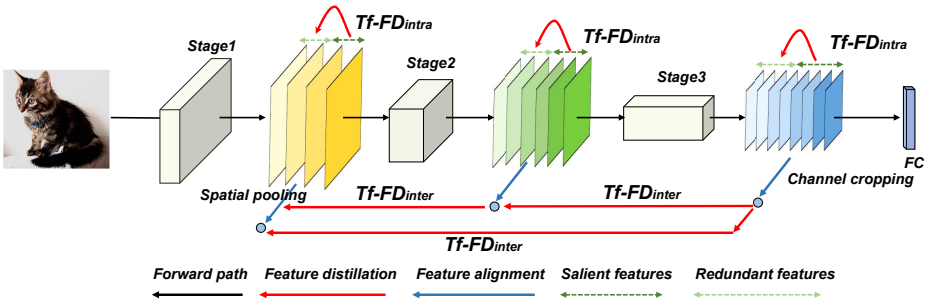

Figure 2: An illustration of our Tf-FD, including intra-layer and inter-layer parts. In the training phase, for intra-layer Tf-FD, we first rank the features on the same layer according to the feature salience. Then the top half of salient features are leveraged to distill the remaining features. And inter-layer Tf-FD capitalizes on features in deeper layers to supervise shallow ones. In the inference phase, the model can be inferred separately.

a student network and training data, Tf-FD achieves such a goal by learning from the salient feature maps in the same layer (intra-layer distillation) and the deeper layers representations (inter-layer distillation).

Intra-layer teacher-free feature distillation. Over-parameterized models tend to produce *redundant* features that contain poor visual concepts [28]. We present intra Tf-FD that uses salient features to supervise redundant features to address this problem. Specifically, we first sort the features of the same channel according to the l_p-norm [35, 64] ($p = 2$) and then use the top half features to distill the bottom half ones. We reduce shallow features to the same resolution as deep ones for feature alignment via average pooling. For channel alignment, we crop the wider deep features into multiple groups, each having the same number of features as the shallow ones. Then, the intra-layer Tf-FD can directly calculate l_2 loss for $\mathcal{D}_f(\cdot)$, which can be formulated as:

$$\mathcal{L}_{\text{intra}} = \frac{1}{\delta}\sum_{i=1}^{\delta}\mathcal{D}_f(\overline{\boldsymbol{\psi}_{\boldsymbol{S}_i}}, \widetilde{\boldsymbol{\psi}_{\boldsymbol{S}_i}}) = \frac{1}{\delta}\sum_{i=1}^{\delta}||\overline{\boldsymbol{\psi}_{\boldsymbol{S}_i}} - \widetilde{\boldsymbol{\psi}_{\boldsymbol{S}_i}}||^2, \tag{2}$$

where δ denotes the number of total layers, $\overline{\boldsymbol{\psi}_{\boldsymbol{S}_i}}$ is the bottom half redundant features and $\widetilde{\boldsymbol{\psi}_{\boldsymbol{S}_i}}$ is the top half salient features. Different from other channel-wise architecture designs (*e.g.*, GhostNet [12]) with extra inference costs, intra-layer Tf-FD is cost-free in inference by optimizing loss rather than changing the model.

Inter-layer teacher-free feature distillation. There are extensive computer vision applications [9, 17, 30] and information-bottleneck theory [9, 17, 55] demonstrating a solid fact: the features of the deep layer contain more task-relevant semantic visual concepts. Thus, deep features always obtain significant gains in the distillation framework [5]. Our inter-layer Tf-FD uses self-features in the deep layer of the student network to supervise shallow ones, which are updated

by the l_2 sum loss during back propagation. The loss of inter-layer Tf-FD can be written as:

$$\mathcal{L}_{\text{inter}} = \frac{1}{\gamma}\sum_{i=1}^{L-1}\sum_{j>i}^{L}\mathcal{D}_f\big(T_{s_i}(\boldsymbol{\psi_{S_i}}), T_{s_j}(\boldsymbol{\psi_{S_j}})\big) = \frac{1}{\gamma}\sum_{i=1}^{L-1}\sum_{j>i}^{L}||T_{s_i}(\boldsymbol{\psi_{S_i}}) - T_{s_j}(\boldsymbol{\psi_{S_j}})||^2, \tag{3}$$

where γ denotes the number of pair loss, L is the number of layers of selected features, we use l_2 distance as D_f, and T_s represents feature alignment. In particular, we use a pooling operation and channel cropping to align features in spatial and channel dimensions without complex transformation. To reduce computation and semantic conflicts in dense cross-layer distillation in Tf-FD, we also propose simple inter-layer Tf-FD for the residual feature pairs in i and $i+1$ layers as $\frac{1}{\gamma}\sum_{i=1}^{L-1}\sum_{j=i+1}^{L}||T_{s_i}(\boldsymbol{\psi_{S_i}}) - T_{s_j}(\boldsymbol{\psi_{S_j}})||^2$. Note that $\boldsymbol{\psi_{S_j}}$ in Equation (3) is frozen when updating losses.

Comparison between inter-layer Tf-FD with residual connection and BYOT. (a) Residual connections alleviate vanishing gradients in deep networks via summation of block-wise cross-layer features. It is adopted in CNN architecture engineering (*e.g.*, ResNet) and cannot be removed in inference. While inter-layer Tf-FD does not exist in inference. (b) The BYOT uses the auxiliary classifier in Deep Supervision [26] to change the original student model into a new multi-exit architecture like ONE [25] and transfers knowledge between these branches. Similar to Deep Supervision, BYOT prevents models from the vanishing gradient problem in terms of optimization. However, shallow classifiers with fewer layers in BYOT have much weaker performance than the student model (41.26% vs 68.12% for ResNet-18 on ImageNet as BYOT reported), and its different optimization properties would affect the optimization of the student network [19]. In sharp contrast, inter-layer Tf-FD improves feature consistency by directly optimizing the cross-layer feature loss without extra parameters.

Overall optimization objectives of Tf-FD. In the vanilla Tf-FD method, we train the student network with three losses (α and $\boldsymbol{\beta}$ are weighting factors):

$$\mathcal{L}_{\text{T}f\text{-FD}} = \mathcal{L}_{\text{CE}}\big(\theta_S, x\big) + \boldsymbol{\beta}\mathcal{L}_{\text{intra}} + \alpha\mathcal{L}_{\text{inter}}, \tag{4}$$

Augmenting Tf-FD with logits teacher-based distillations. Being a generic feature regularizer framework, Tf-FD itself provides a new teacher-free feature KD framework when selecting/training extra teachers are difficult. When pre-trained teachers are available, since different sources of knowledge (other models and self logits), our Tf-FD could naturally combine with teacher-based KD losses $\mathcal{L}_{\text{KD}}$ to train student models. To explore this potential, we apply distillation on the outputted logits from the two heads of the student network to promote the performance of our Tf-FD. We call the resulting method Tf-FD†. Specifically, Tf-FD† performs logits distillation with KL divergence $\mathcal{D}_{kl}\big(\theta_S \| \theta_T\big)$, and the total training objective can be expressed as:

$$\mathcal{L}_{\text{T}f\text{-FD}\dagger} = \mathcal{L}_{\text{CE}}\big(\theta_S, x\big) + \mathcal{D}_{kl}\big(\theta_T, \theta_S\big) + \boldsymbol{\beta}\mathcal{L}_{\text{intra}} + \alpha\mathcal{L}_{\text{inter}}, \tag{5}$$

3.3 Discussion of the Relationships with Feature Regularization

Decoupled from the additional teacher model, our Tf-FD extends the feature distillation to a more generic training method. Tf-FD improves the generalization of the model by supervision of self-features, which can also be considered as feature noise. Therefore, we discuss why Tf-FD works from the perspective of feature regularization. To make the analysis process as clear as possible, we select the inter-layer Tf-FD as the distillation function alone. For the i^{th} layer of network, its features $\boldsymbol{\psi_{S_i}}$ are supervised by deeper features $\{\boldsymbol{\psi_{S_j}}|j \in \{i+1, i+2, \cdots, L\}\}$. Thus, updated loss function for i^{th} layer can be defined as $\mathcal{L}_{\boldsymbol{i^{\text{th}}-layer}} = \mathcal{L}_{\text{CE}} + \alpha \times \frac{1}{\gamma} \sum_{j>i}^{L} ||T_{s_i}(\boldsymbol{\psi_{S_i}}) - T_{s_j}(\boldsymbol{\psi_{S_j}})||^2$. Similar to the error update formula of the parameters, the $\boldsymbol{\psi_{S_i}}$ is updated as following:

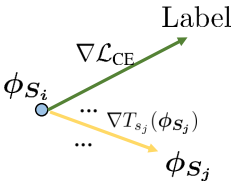

Figure 3: Illustration of feature updating.

$$\widehat{\boldsymbol{\psi_{S_i}}} = \boldsymbol{\psi_{S_i}} - \eta\nabla\big(\mathcal{L}_{\text{CE}}\big(\theta_S, x\big) + \alpha \times \frac{1}{\gamma}\sum_{j>i}^{L} ||T_{s_i}(\boldsymbol{\psi_{S_i}}) - T_{s_j}(\boldsymbol{\psi_{S_j}})||^2\big). \quad (6)$$

where η is the learning rate, and $\widehat{\boldsymbol{\psi_{S_i}}}$ is the updated $\boldsymbol{\psi_{S_i}}$. As mentioned above, we adopt the simple feature alignment for T_s. Therefore, we simplify its role in the following analysis.

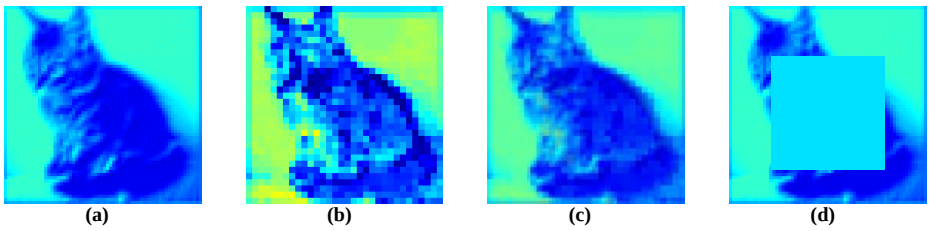

Figure 4: Schematic diagram of Tf-FD (c) and Dropblock [10] (d) for feature of Conv2_x (a) and Conv3_x (b) of ResNet-18 on ImageNet. Tf-FD implicitly employs Conv3_x features as noises for Conv2_x features when explicitly optimizing this pair of inter Tf-FD loss. Moreover, Dropblock applies a random mask for feature regularization on the Conv2_x features.

As shown in Figure 3, during the training process, the $\boldsymbol{\psi_{S_i}}$ needs to be updated in the direction of $\boldsymbol{\psi_{S_j}}$ so that feature distillation loss is reduced to make Equation (6) converge. This illustrates that Tf-FD implicitly applies a feature distortion of $\boldsymbol{\psi_{S_j}}$ to $\boldsymbol{\psi_{S_i}}$ by optimizing the distillation loss. The regularization effect of Tf-FD depends on the feature knowledge contained in $\boldsymbol{\psi_{S_j}}$ and the weighting factor α. Therefore, Tf-FD seeks to leverage privileged within the

network itself to maximize regularization gains. As shown in Figure 4, while Dropblock [10] utilizes random masks that often cause semantic damage, Tf-FD preserves more semantic information. Furthermore, when $\boldsymbol{\psi_{S_j}}$ is features $\boldsymbol{\psi_T}$ from other teacher models, the Equation (6) reveals the feature regularization role played by general feature distillation. From this perspective, different teacher models in the previous exploration experiments provide feature regularization disturbances and thus all achieve performance gains. This explains why feature distillation could work without additional teacher modules.

Table 1: Top-1 accuracies (%) of teacher-free methods, self-knowledge distillations (self-KD) and teacher-based distillation (Tb-KD) reported in CRD [50] under the same training setting of 240 epochs. Note that teacher models are only for teacher-based distillations, and Tf-FD is completely free of teacher models.

Method	Student	ResNet-20	ResNet-32	WRN-16-2	ResNet-8×4	VGG-8
		69.06	71.14	73.26	72.50	70.36
Tf method	Dropout [24]	69.22	71.31	73.31	72.68	70.52
	DropBlock [10]	69.65	71.56	73.42	72.87	70.76
	SAD [17]	69.76	71.48	73.68	72.71	70.72
	LS [48]	69.87	71.86	73.65	72.91	70.87
	Tf-KD [58]	70.02	72.06	73.88	73.05	71.05
	Tf-FD (ours)	**70.62**	**72.55**	**74.33**	**73.62**	**71.62**
Self-KD	CS-KD [59]	70.12	72.26	73.98	73.10	71.26
	BYOT [62]	70.37	72.46	73.70	72.98	70.88
	ONE [25]	70.77	72.78	74.68	73.51	72.01
	Teacher	ResNet-110	ResNet-110	WRN-40-2	ResNet-32×4	VGG-13
		74.31	74.31	75.61	79.42	74.64
Tb-KD	FitNets [44]	68.99	71.06	73.58	73.50	71.02
	AT [53]	70.22	72.31	74.08	73.44	71.43
	SP [52]	70.04	72.69	73.83	72.94	72.68
	PKT [38]	70.25	72.61	74.54	73.64	72.88
	AB [15]	69.53	70.98	72.50	73.17	70.94
	NST [20]	69.53	71.96	73.68	73.30	71.53
	KD [16]	70.67	73.08	74.92	73.33	72.98
	CRD [50]	71.46	73.48	75.64	75.51	73.94
	Tf-FD† (ours)	**71.56**	**73.68**	**75.68**	**75.65**	**74.08**

4 Experiments

In this section, we first evaluate our Tf-FD/Tf-FD† on CIFAR-100 in § 4.1 and ImageNet in § 4.2. Apart from image classification, Tf-FD is also effective for downstream tasks, such as object detection in § 4.3. Comprehensive ablation experiments are performed to analyze the key design in § 4.4.

4.1 Experiments on CIFAR-100

Implementation. The CIFAR-100 dataset [23] is used for the trials without extra strong data augmentation. We conduct experiments on ResNets [14], WRNs [60] and VGG [46] with CRD's settings [50], whose training epochs are 240. The weight decay is 5×10^{-4}, and the optimizer is SGD. Initialized at 0.1, the multi-step learning rate increases by 0.1 every 150, 180, and 210 epochs.

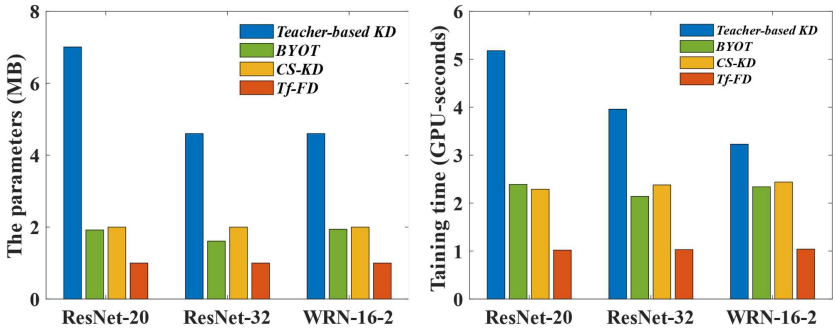

Figure 5: (Left) The total training parameters and (Right) training time of teacher-based KD [61], BYOT [62], CS-KD [59] and our Tf-FD, which are measured on a single NVIDIA 2080Ti. Number of Y-axis represents the improved ratios compared to baseline.

Comparison results. In Table 1, we report the results of various regularization, self-KDs, and teacher-based methods on CIFAR-100. For ResNet-like models, Tf-FD obtains 1.12% $\sim$ 1.56% absolute accuracy gains, which shows its practical value for different depth and width networks. Besides, on WRN and VGG, Tf-FD outperforms baselines with 1.07% $\sim$ 1.26% margins. Compared to feature regularizers, Tf-FD outperforms DropBlock [10] with 0.75% $\sim$ 0.99% margins. This proves that more semantic feature distortions of Tf-FD can obtain significant performance gains. Furthermore, Tf-FD achieves superior performance than these typical self-distillation methods (*e.g.*, BYOT [62] and CS-KD [59]). Compared to feature distillation methods with a strong pre-trained teacher model, Tf-FD achieves competitive performance gains, indicating that our framework without the teacher model can still effectively boost performance. Tf-FD's basic performance is already better than SOTA teacher-free logits KDs/ regularizers and can be further improved with advanced loss functions, demonstrating its effectiveness. Extra teachers contain richer knowledge than students themselves. Thus, teacher-based KDs usually have superior accuracy than teacher-free KDs (including Tf-FD). In particular, the combination (Tf-FD†) of Tf-FD with logits KD [16] obtains 0.60% $\sim$ 1.22% gains than KD, which illustrate their orthogonality. Compared to recent SOTA teacher-based KDs with contrastive training with pair-wise augmentations (*e.g.*, CRD [50]), Tf-FD† achieves superior

accuracy and training efficiency. In summary, our Tf-FD can noticeably improve the performance of the student network without additional overhead, which effectively expands the application of feature distillation.

Training efficiency. Furthermore, we compare the training cost between Tf-FD and teacher-based feature distillation under the same settings. As shown in Figure 5, Tf-FD does not introduce additional parameters and achieves $3\times \sim 5\times$ training acceleration than BYOT, CS-KD, and teacher-based KD.

Table 2: Top-1/Top-5 accuracies (%) of teacher-free methods, self-knowledge distillations (self-KD) and teacher-based knowledge distillations (KDs) on ImageNet dataset. Most results of other methods are references to the original paper report. N/A means no published result is available.

Model	ResNet-18 [Student]			Model	ResNet-34 [Teacher]		
Type	Method	Top-1	Top-5	Type	Method	Top-1	Top-5
	Student	69.75	89.07		Teacher	73.31	91.42
Tf-method	Dropout [24]	69.79	89.16	Tb-KD	KD [16]	70.66	89.88
	DropBlock [10]	69.88	89.32		AT [53]	70.70	90.00
	SAD [17]	69.82	89.24		AFD [7]	70.39	N/A
	LS [48]	69.93	N/A		SP [52]	70.62	89.88
	Tf-KD [58]	70.15	N/A		CC [39]	69.96	89.17
	Tf-FD (ours)	**70.46**	**89.72**		VID [1]	70.30	N/A
Self-KD	BYOT [62]	69.84	N/A		FitNets [44]	70.31	N/A
	FRSKD [21]	70.17	N/A		SemCKD [36]	70.87	N/A
	ONE [25]	70.55	N/A		**Tf-FD† (ours)**	**71.00**	**90.22**

4.2 Experiments on ImageNet

Detailed implementation. The experiments on ImageNet [45] are carried out via ResNet-18 [14]. We use the same training configurations (*e.g.*, 100 training epochs) with most distillation techniques. Warm-up and early-decay schedules are performed for loss weight of Tf-FD.

Comparison results. Table 2 reports the performance of our approach on ImageNet. Tf-FD improves baseline models of ResNet-18 by 0.71% gains and outperforms regularization approaches and self-KDs methods with $0.29\% \sim 0.61\%$ margins, which supports its superiority on the large-scale dataset. Despite the fact that traditional teacher-based distillation methods use the pre-trained ResNet-34 as a teacher, Tf-FD produces very competitive performance in teacher-free configurations. Equipped with knowledge distillation for outputted logits, Tf-FD† obtains 1.25% gain than baseline and surpasses other teacher-based approaches.

Table 3: Results on object detection [40]. R50 represents using ResNet-50 as backbone. Note that teacher models are only for other feature distillation methods, and Tf-FD is completely free of teacher models.

Detector	Model	AP	AP_{50}	AP_{75}	AP_L	AP_M	AP_S
Faster R-CNN (R101-FPN)	Teacher baseline	42.04	62.48	45.88	54.60	45.55	25.22
Faster R-CNN (R50-FPN)	Student baseline	37.93	58.84	41.05	49.10	41.14	22.44
	KD [16]	38.35	59.41	41.71	49.48	41.80	22.73
	FitNets [44]	38.76	59.62	41.80	50.70	42.20	22.32
	FGFI [54]	39.44	60.27	43.04	51.97	42.51	22.89
	Tf-FD† (ours)	**38.92**	**59.71**	**41.93**	**50.88**	**41.92**	**21.96**

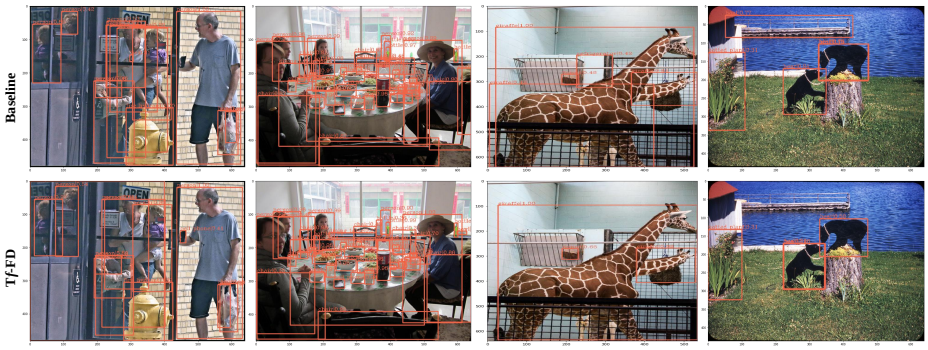

Figure 6: Visualization of bounding-box detection outputs of Faster R-CNN via ResNet-50 backbone on the MS-COCO2017. The two figures on the left illustrate that Tf-FD is more effective in capturing small objects than baseline. The two figures on the right indicate that fewer false positives occur in Tf-FD.

4.3 Extension to Object Detection

Implementation. We evaluate Tf-FD on MS-COCO2017 dataset [32] , which includes more than 120K images encompassing 80 categories. We apply Tf-FD to Faster R-CNN [43] and employ Detectron2 as the baseline. Note that the Tf-FD distillation is carried out at the detection fine-tuning stage with advanced feature losses [54, 56]. All models are trained using a 2× learning schedule, and their performance is evaluated on the MS-COCO2017 validation set.

Comparison results. Table 3 demonstrate that Tf-FD improves the AP 0.99 on Faster R-CNN. Compared with other distillation methods with strong teacher models, Tf-FD outperforms KD [16] and FitNets [44] and obtains competitive gains with FGFI [54], which is particularly designed for object detection. As shown in Figure 6, visualization results demonstrate the effectiveness of Tf-FD in small object detection and reducing false positives. The success of challenging object detection tasks demonstrates the generality and effectiveness of our approach. Besides this simple extension, we are also designing and investigating specially

Table 4: Ablation study of each loss added to different blocks of ResNet-20 on CIFAR-100. S2, S3, and S4 refer to Conv2_x features, Conv3_x features, and Conv4_x features, respectively. S3 $\rightarrow$ S2 means that Conv3_x features are employed to distill Conv2_x features. $\uparrow$ refers to the performance gain.

$\mathcal{L}_{\text{inter}}$	S3 $\rightarrow$ S2	✓	×	×	✓	×	×	×	×	✓
	S4 $\rightarrow$ S3	×	✓	×	✓	×	×	×	×	✓
	S4 $\rightarrow$ S2	×	×	✓	✓	×	×	×	×	✓
$\mathcal{L}_{\text{intra}}$	S2	×	×	×	×	✓	×	×	✓	✓
	S3	×	×	×	×	×	✓	×	✓	✓
	S4	×	×	×	×	×	×	✓	✓	✓
ResNet-20 Top-1 (%)		69.84 (0.78↑)	70.29 (1.23↑)	70.18 (1.12↑)	70.31 (1.25↑)	70.26 (1.20↑)	70.29 (1.23↑)	70.41 (1.35↑)	70.51 (1.45↑)	70.62 (1.56↑)

designed feature regularizers for object detection and semantic segmentation [42] following our Tf-FD idea.

4.4 Ablation Study

We concentrate on the effect of each element of our approach in this section.

Design of each loss. The ablation research on CIFAR-100 with ResNet-20 is conducted in Table 4 to illustrate the individual efficacy of various components in Tf-FD. It is observed that (a) A single loss of Tf-FD can also obtain 0.78% $\sim$ 1.23% accuracy gains. (b) The intra-layer Tf-FD in the deep layer obtains more obvious performance improvement. This is consistent with the fact that some feature regularization methods [10, 49] work well on the final stage of the neural network.

Advanced feature mimicking loss for Tf-FD. In Figure 7 (b), we explore different feature mimicking losses for Tf-FD for ResNet-20 on CIFAR-100. The AT [61], SP [52] and ICKD [33] achieve more obvious performance than the simple l_2 loss, indicating that the specially designed mimicking loss can further improve the performance of Tf-FD. The AB [15] use complex feature mapping, resulting in the loss of valuable feature knowledge. We adopt the simple l_2 loss for Tf-FD for all previous analyses and experiments.

Sensitivity study for hyper-parameters α and β. α and β are loss weights of $\mathcal{L}_{inter}$ and $\mathcal{L}_{intra}$. As shown in Figure 7, experiments on CIFAR-100 and ResNet-20 are conducted to study their sensitivity. The results demonstrate that (α, β) = (0.0005, 0.0008) is the best solution for the hyper-parameter setting. Even in the worst situation when $\alpha = 0.01$ and $\beta = 0.0001$, Tf-FD still achieves 0.71% accuracy improvements than the baseline and outperforms some KDs (*e.g.*, FitNets [44] and AB [15]) in Table 1.

Attention map visualization. Our Tf-FD would help the network pay attention to important information. Figure 8 illustrates that the gradient activation map of Tf -FD is more concerned with the correct region than Dropblock.

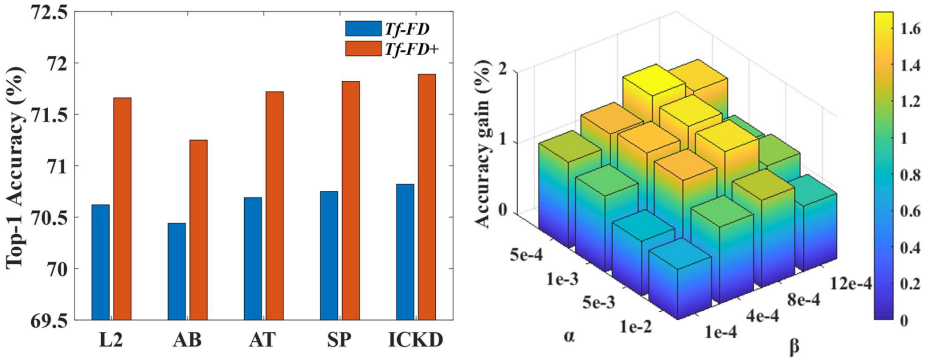

Figure 7: (Left) Comparison results of different feature mimicking loss for Tf-FD. (Right) Hyper-parameter analysis: the top-1 accuracy (%) of Tf-FD with various α (Y-axis) and $\boldsymbol{\beta}$ (X-axis) for ResNet-20 trained on CIFAR-100.

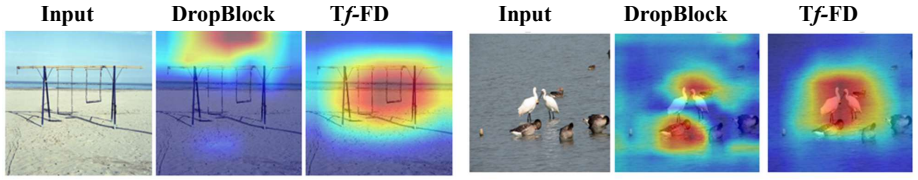

Figure 8: Comparison on the Grad-CAM++ ([4]) visualization results between the features of the Dropblock, and Tf-FD on ImageNet.

5 Conclusion

In this Tf-FD work, we develop a novel paradigm for performing feature distillation efficiently without teacher models. Based on our insight that feature distillation does not depend on additional modules, Tf-FD achieves this goal by capitalizing on channel-wise and layer-wise salient self-features without setting complicated feature alignment and assuming additional teacher modules to be available. From the perspective of regularization, Tf -FD exerts meaningful feature perturbations by optimizing the loss. This insight opens new doors for the community to trade technical routes for both feature regularization and distillation. In future work, we will make an effort to analyze Tf-FD from a theoretical perspective and explore its application with unique designs on downstream tasks such as FPN-free detectors, VGG-like model optimization, and weakly supervised semantic segmentation [42], etc. We hope this elegant and practical approach will inspire more investigation into the interpretability and widespread applications of knowledge distillation for feature representations.

References

1. Ahn, S., Hu, S.X., Damianou, A., Lawrence, N.D., Dai, Z.: Variational information distillation for knowledge transfer. In: CVPR (2019) 2, 11
2. Brown, T.B., Mann, B., Subbiah, N.R.M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., M, D., Ziegler, Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. arXiv preprint, arXiv:2005.14165 (2020) 1
3. Bucila, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: KDD (2006) 4
4. Chattopadhyay, A., Sarkar, A., Howlader, P., Balasubramanian, V.: Grad-cam++: Improved visual explanations for deep convolutional networks. In: WACV (2018) 14
5. Chen, D., Mei, J.P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., Chen, C.: Cross-layer distillation with semantic calibration. arXiv preprint, arXiv:2012.03236 (2020) 2, 6
6. Cheng, X., Rao, Z., Chen, Y., Zhang, Q.: Explaining knowledge distillation by quantifying the knowledge. In: CVPR (2020) 3
7. Chung, I., Park, S., Kim, J., Kwak, N.: Feature-map-level online adversarial knowledge distillation. In: ICML (2020) 4, 11
8. Dong, P., Niu, X., Li, L., Xie, L., Zou, W., Ye, T., Wei, Z., Pan, H.: Prior-guided one-shot neural architecture search. arXiv preprint arXiv:2206.13329 (2022) 2
9. Dong, Z., Hanwang, Z., Jinhui, T., Xiansheng, H., Qianru, S.: Self-regulation for semantic segmentation. International Conference on Computer Vision (ICCV) (2021) 6
10. Ghiasi, G., Lin, T.Y., Le, Q.V.: Dropblock: A regularization method for convolutional networks. In: NeurIPS (2018) 4, 8, 9, 10, 11, 13
11. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 1
12. Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., Xu, C.: Ghostnet: More features from cheap operations. In: CVPR (2020) 6
13. Han, S., Pool, J., Tran, J., Dally, W.J.: Learning both weights and connections for efficient neural networks. In: NeurIPS (2015) 1
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 10, 11
15. Heo, B., Lee, M., Yun, S., Choi, J.Y.: Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In: AAAI (2019) 2, 9, 13
16. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint, arXiv:1503.02531 (2015) 1, 2, 4, 9, 10, 11, 12
17. Hou, Y., Ma, Z., Liu, C., Loy, C.C.: Learning lightweight lane detection cnns by self attention distillation. In: ICCV (2019) 5, 6, 9, 11
18. Hu, Y., Wang, X., Li, L., Gu, Q.: Improving one-shot nas with shrinking-and-expanding supernet. Pattern Recognition (2021) 2
19. Huang, G., Chen, D., Li, T., Wu, F., van der Maaten, L., Weinberger, K.Q.: Multi-scale dense networks for resource efficient image classification. In: ICLR (2018) 5, 7
20. Huang, Z., Wang, N.: Like what you like: Knowledge distill via neuron selectivity transfer. arXiv preprint, arXiv:1707.01219 (2017) 2, 9
21. Ji, M., Shin, S., Hwang, S., Park, G., Moon, I.C.: Refine myself by teaching myself: Feature refinement via self-knowledge distillation. arXiv preprint, arXiv:2103.08273 (2021) 2, 4, 11

22. Kim, J., Park, S., Kwak, N.: Paraphrasing complex network: Network compression via factor transfer. In: NeurIPS (2018) 2
23. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech Report (2009) 10
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS (2012) 1, 9, 11
25. Lan, X., Zhu, X., Gong, S.: Knowledge distillation by on-the-fly native ensemble. In: NeurIPS (2018) 2, 4, 7, 9, 11
26. Lee, C.Y., Xie, S., Gallagher, P., Zhang, Z., Tu, Z.: Deeply-supervised nets. In: AISTATS (2015) 7
27. Lee, H., Hwang, S.J., Shin, J.: Self-supervised label augmentation via input transformations. In: ICML (2020) 5
28. Li, H., Kadav, A., Durdanovic, I., Samet, H., Graf, H.P.: Pruning filters for efficient convnets. In: ICLR (2017) 1, 3, 6
29. Li, L., Shiuan-Ni, L., Yang, Y., Jin, Z.: Boosting online feature transfer via separable feature fusion. In: IJCNN (2022) 4
30. Li, L., Shiuan-Ni, L., Yang, Y., Jin, Z.: Teacher-free distillation via regularizing intermediate representation. In: IJCNN (2022) 6
31. Li, L., Wang, Y., Yao, A., Qian, Y., Zhou, X., He, K.: Explicit connection distillation (2020) 2
32. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollar, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) 12
33. Liu, L., Huang, Q., Lin, S., Xie, H., Wang, B., Chang, X., Liang, X.: Exploring inter-channel correlation for diversity-preserved knowledgedistillation. In: ICCV (2021) 13
34. Liu, Y., Jia, X., Tan, M., Vemulapalli, R., Zhu, Y., Green, B., Wang, X.: Search to distill: Pearls are everywhere but not the eyes. In: CVPR (2020) 2
35. Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S., Zhang, C.: Learning efficient convolutional networks through network slimming. In: ICCV (2017) 6
36. Malinin, A., Mlodozeniec, B., Gales, M.: Ensemble distribution distillation. In: ICLR (2020) 11
37. Pan, H., Jiang, H., Niu, X., Dou, Y.: Dropfilter: A novel regularization method for learning convolutional neural networks. arXiv preprint, arXiv:1811.06783 (2018) 4
38. Passalis, N., Tefas, A.: Learning deep representations with probabilistic knowledge transfer. In: ECCV (2018) 9
39. Peng, B., Jin, X., Liu, J., Zhou, S., Wu, Y., Liu, Y., Li, D.s., Zhang, Z.: Correlation congruence for knowledge distillation. In: ICCV (2019) 11
40. Pengguang, C., Shu, L., Hengshuang, Z., Jia, J.: Distilling knowledge via knowledge review. In: CVPR (2021) 12
41. Phuong, M., Lampert, C.H.: Distillation-based training for multi-exit architectures. In: ICCV (2019) 2, 4
42. Qin, J., Wu, J., Xiao, X., Li, L., Wang, X.: Activation modulation and recalibration scheme for weakly supervised semantic segmentation. In: AAAI (2022) 13, 14
43. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv preprint, arXiv:1506.01497 (2015) 12
44. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. In: ICLR (2015) 2, 4, 5, 9, 11, 12, 13
45. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Li, F.F.: Imagenet large scale visual recognition challenge. IJCV (2015) 11

46. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint, arXiv:1409.1556 (2014) 10
47. Sun, D., Yao, A.: Deeply-supervised knowledge synergy. In: CVPR (2019) 4
48. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: ICCV (2016) 5, 9, 11
49. Tang, Y., Wang, Y., Xu, Y., Shi, B., Xu, C., Xu, C., Xu, C.: Beyond dropout: Feature map distortion to regularize deep neural networks. In: AAAI (2020) 4, 13
50. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. In: ICLR (2020) 3, 9, 10
51. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: CVPR (2015) 4
52. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: ICCV (2019) 2, 9, 11, 13
53. Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. arXiv preprint, arXiv:1706.03762 (2017) 1, 9, 11
54. Wang, T., Yuan, L., Zhang, X., Feng, J.: Distilling object detectors with fine-grained feature imitation. In: CVPR (2019) 12
55. Wolchover, N., Reading, L.: New theory cracks open the black box of deep learning. Quanta Magazine (2017) 6
56. Yang, Z., Li, Z., Jiang, X., Gong, Y., Yuan, Z., Zhao, D., Yuan, C.: Focal and global knowledge distillation for detectors. In: CVPR (2022) 12
57. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In: CVPR (2017) 4
58. Yuan, L., Tay, F.E., Li, G., Wang, T., Feng, J.: Revisiting knowledge distillation via label smoothing regularization. In: CVPR (2020) 5, 9, 11
59. Yun, S., Park, J.S., Lee, K., Shin, J.: Regularizing class-wise predictions via self-knowledge distillation. In: CVPR (2020) 5, 9, 10
60. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016) 10
61. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In: ICLR (2017) 2, 4, 5, 10, 13
62. Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., Ma, K.: Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In: ICCV (2019) 2, 4, 9, 10, 11
63. Zhang, Z., Sabuncu, M.R.: Self-distillation as instance-specific label smoothing. arXiv preprint, arXiv:2006.05065 (2020) 5
64. Zhao, C., Ni, B., Zhang, J., Zhao, Q., Zhang, W., Tian, Q.: Variational convolutional neural network pruning. In: CVPR (2019) 6
65. Zhou, A., Yao, A., Guo, Y., Xu, L., Chen, Y.: Incremental network quantization: Towards lossless cnns with low-precision weights. In: ICLR (2017) 1
66. Zhou, S., Yuxin, W., Ni, Z., Zhou, X., Wen, H., Zou, Y.: Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. arXiv preprint, arXiv:1606.06160 (2016) 1