Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks

Lin Wang, *Student Member, IEEE,* and Kuk-Jin Yoon, *Member, IEEE*

Abstract—Deep neural models in recent years have been successful in almost every field, including extremely complex problem statements. However, these models are huge in size, with millions (and even billions) of parameters, thus demanding more heavy computation power and failing to be deployed on edge devices. Besides, the performance boost is highly dependent on redundant labelled data. To achieve faster speeds and to handle the problems caused by the lack of data, knowledge distillation (KD) has been proposed to transfer information learned from one model to another. KD is often characterized by the so-called 'Student-Teacher' (S-T) learning framework and has been broadly applied in model compression and knowledge transfer. This paper is about KD and S-T learning, which are being actively studied in recent years. First, we aim to provide explanations of what KD is and how/why it works. Then, we provide a comprehensive survey on the recent progress of KD methods together with S-T frameworks typically for vision tasks. In general, we consider some fundamental questions that have been driving this research area and thoroughly generalize the research progress and technical details. Additionally, we systematically analyze the research status of KD in vision applications. Finally, we discuss the potentials and open challenges of existing methods and prospect the future directions of KD and S-T learning.

Index Terms—Knowledge distillation (KD), Student-Teacher learning (S-T), Deep neural networks (DNN), Visual intelligence.

1 Introduction

The success of deep neural networks (DNNs) generally depends on the elaborate design of DNN architectures. In large-scale machine learning, especially for tasks like image and speech recognition, most DNN-based models are over-parameterized to extract the most salient features and to ensure generalization. Such cumbersome models are usually very deep and wide, which require a considerable amount of computation for training and are impossible to be operated in real-time. Thus, to achieve faster speeds, many researchers have been trying to utilize the cumbersome models that are trained to obtain lightweight DNN models, which can be deployed in edge devices. That is, when the cumbersome model has been trained, it can be used to learn a small model that is more suitable for real-time applications or deployment [94] as depicted in Fig. 1(a).

On the other hand, the plausible performance of DNNs is also heavily dependent on very large and highly redundant datasets. For such a reason, many endeavours have been done to retrench the amount of labelled training data without hurting too much the performance of DNNs. A popular approach for handling such a lack of data is to *transfer knowledge* from one source task to facilitate the learning on the target task. One typical example is semi-supervised learning in which a model is trained with the only resort to a small portion of labelled data and a large number of unlabelled data. Since the supervised cost is undefined for the unla-

• L. Wang and K.-J. Yoon are with the Visual Intelligence Lab., Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Guseong-dong, Yuseong-gu, Daejeon 34141, Republic of Korea.
E-mail: kjyoon@kaist.ac.kr (corresponding author)

Manuscript received April 19, 2020; revised August 26, 2020.

what is semi-supervised
what is transfer learning

belled examples, it is crucial to apply consistency cost or add regularization to matching two predictions for both labelled and unlabelled data. In this case, knowledge is transferred in the model that assumes a dual role as *teacher* and *student* [222]. For the unlabelled data, the student learns as before; however, the teacher generates targets, which are then used by the student for learning. The common goal of such a learning metric is to form a better teacher model from the student without additional training, as shown in Fig. 1(b). Another typical example is self-supervised learning, where the model is trained with artificial labels constructed by the input transformations (*e.g.*, rotation, flipping, color change, cropping). In such a situation, the knowledge from the input transformations is transferred to supervise the model itself to improve its performance as illustrated in Fig. 1(c).

This paper is about *knowledge distillation (KD) and student-teacher (S-T) learning*, a topic that has been actively studied in recent years. Generally speaking, KD is widely regarded as a primary mechanism that enables humans to quickly learn new complex concepts when given only small training sets with the same or different categories [79]. In deep learning, KD is an effective technique that has been widely used to transfer information from one network to train another network constructively. KD was first defined by [26] and generalized by Hinton *et al.* [94]. KD has been broadly applied to two distinctive fields: model compression (refer to Fig. 1(a) and knowledge transfer (refer to Fig. 1 (b) and (c)). For model compression, a smaller student model is trained to mimic a pretrained larger model or ensemble of models. Although various forms of knowledge are defined based on the purposes or tasks, one common characteristic of KD is symbolized by its *S-T framework*, where the model that provides knowledge is called the teacher and the model

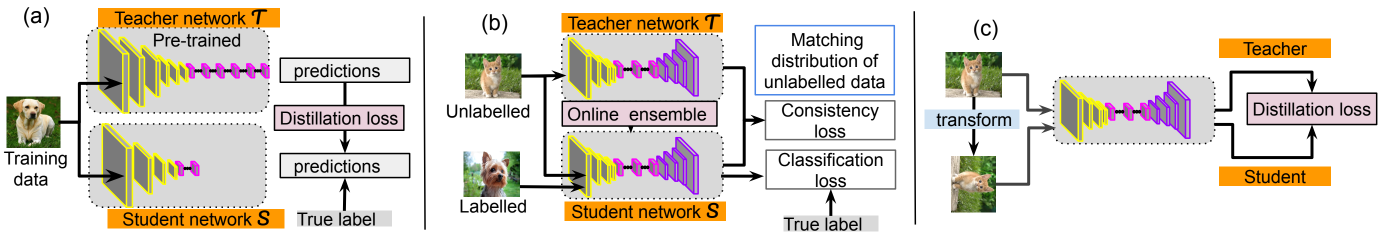


Fig. 1. Illustrations of KD methods with S-T frameworks. (a) for model compression and for knowledge transfer, *e.g.*, (b) semi-supervised learning and (c) self-supervised learning.

that learns the knowledge is called the student.

In this work, we focus on analyzing and categorizing existing KD methods accompanied by various types of S-T structures for model compression and knowledge transfer. We review and survey this rapidly developing area with particular emphasis on the recent progress. Although KD has been applied to various fields, such as visual intelligence, speech recognition, natural language processing (NLP), etc., this paper mostly focuses on the KD methods in the vision field, as most demonstrations have been done on computer vision tasks. KD methods used in the area like NLP, speech recognition can be conveniently explained using the KD prototypes in vision. As the most studied KD methods are for model compression, we systematically discuss the technical details, challenges, and potentials. Meanwhile, we also concentrate on the KD methods for knowledge transfer in semi-supervised learning, self-supervised learning, etc., and we highlight the techniques that take S-T learning as a way of learning metric.

We take into account some fundamental questions that have been driving this research area. Specifically speaking, what is the theoretical principle for KD and S-T learning? What makes one distillation method better than others? Is using multiple teachers better than one teacher? Can bigger models always make better teachers and teach more robust students? Does a student need to learn knowledge only if a teacher model exists? Is the student able to learn itself? Is off-line KD always better than online learning, to name a few?

With the questions being discussed, we incorporate the potentials of existing KD methods and prospect the future directions of the KD methods together with S-T frameworks. We especially stress the importance of recently developed technologies, such as neural architecture search (NAS), graph neural networks (GNNs), gating mechanisms for empowering KD. Besides, we also emphasize the potential of KD methods for tackling tricky problems in particular vision fields such as 360° vision and event-based vision.

The main contributions of this paper are three-fold:

- We give a comprehensive overview of KD and S-T learning methods, including problem definition, theoretical analysis, a family of KD methods with deep learning and vision applications.
- We provide a systematic overview and analysis of recent advances of KD methods and S-T frameworks hierarchically and structurally and offer insights and summaries for the potentials and challenges of each category.
- We discuss the problems and open issues and identify new trends and further direction to provide insightful guidance in this research area.

The organization of this paper is as follows. First, we explain why we need to care about KD and S-T learning in Sec.2. Then, we provide a theoretical analysis of KD in Sec.3. Followed by Sec.4 to Sec.14, we categorize the existing methods and analyze the challenges and potential. Fig. 2 shows the taxonomy of KD with S-T learning to be covered in this survey in a hierarchically-structured way. In Sec.15, based on the taxonomy, we will discuss the answers to the questions raised in Sec.1. Sec.16 will present the future potentials of KD and S-T learning, followed by a conclusion in Sec.17.

2 WHAT IS KD AND WHY CONCERNING IT?

What's KD? KD was first proposed by Hinton *et al.* [94]. KD refers to the method that helps the training process of a smaller student network under the supervision of a larger teacher network. Unlike other compression methods, KD can downsize a network regardless of the structural difference between the teacher and the student network. In [94], the knowledge is transferred from the teacher model to the student by minimizing the difference between the logits (the inputs to the final softmax) produced by the teacher model and those produced by the student model.

However, in many situations, the output of softmax function on the teacher's logits has the correct class at a very high probability, with all other class probabilities very close to zero. In such a circumstance, it does not provide much information beyond the ground truth labels already provided in the dataset. To tackle such a problem, Hinton *et al.* [94] introduced the concept of 'softmax temperature', which can make the target to be 'soft.' Given the logits z from a network, the class probability p_i of an image is calculated as:

$$p_i = \frac{\exp(\frac{z_i}{T})}{\sum_j \exp(\frac{z_i}{T})} \quad (1)$$

where T is the temperature parameter. When $T = 1$, we get the standard softmax function. As T increases, the probability distribution produced by the softmax function becomes softer, providing more information as to which classes the teacher found more similar to the predicted class. The information provided in the teacher model is called *'dark knowledge'* [94]. It is the dark knowledge that affects the overall flow of information to be distilled. When computing the distillation loss, the same T used in the teacher is used to compute the logits of the student.

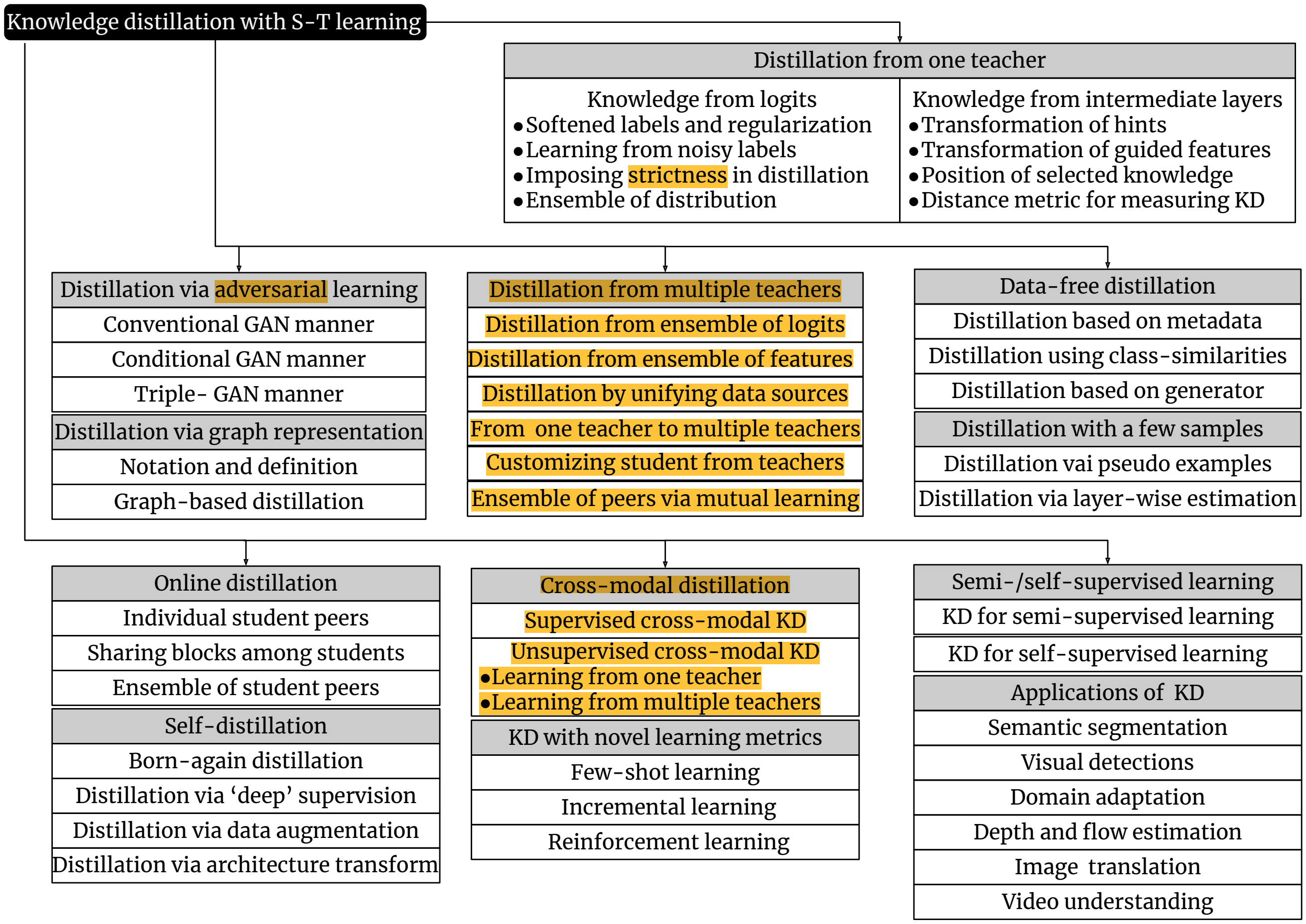


Fig. 2. Hierarchically-structured taxonomy of knowledge distillation with S-T learning.

For the images with ground truth, Hinton *et al.* [94] stated that it is beneficial to train the student model together with the ground truth labels in addition to the teacher's soft labels. Therefore, we also calculate the 'student loss' ($T = 1$) between the student's predicted class probabilities and the ground truth labels. The overall loss function, compromising the student loss, and distillation loss is calculated as:

$$\begin{aligned}\mathcal{L}(x, W) = \alpha * H(y, \sigma(z_s; T = 1) + \\ \beta * H(\sigma(z_t; T = \tau), \sigma(z_s, T = \tau))\end{aligned} \tag{2}$$

where x is the input, W are the parameters of student model, H is the loss function (*e.g.*, cross-entropy loss), y is the ground truth label, σ is the softmax function parameterized by the temperature T, and α and β are coefficients. z_s and z_t are the logits of the student and teacher respectively.

Why concerning KD? KD has become a field in itself in the machine learning community, with broad applications to computer vision, speech recognition, NLP, etc. From 2014 to now, many research papers [1], [2] have been presented in the prime conferences, such as CVPR, ICCV, ECCV, NIPS, ICML, ICLR, etc., and the power of KD has been extended to many learning processes (*e.g.*, few-shot learning) except to model compression. The trend in recent years is that KD with S-T frameworks, has become a crucial tool for knowledge transfer, together with model compression in many tasks. The rapid increase in scientific activity on KD has been accompanied and nourished by a remarkable string of empirical successes both in academia and industry. The particular highlights on some representative applications are given in Sec.15, and in the following section, we provide systematic theoretical analysis.

3 A THEORETICAL ANALYSIS OF KD

Many KD methods have been proposed with various intuitions. However, there is no commonly agreed theory on how knowledge is transferred, thus making it difficult to effectively evaluate the empirical results and less actionable to design new methods in a more disciplined way. Recently, Ahh *et al.* [7], Hegde *et al.* [90] and Tian *et al.* [224] formulate KD as a maximization of mutual information between the representations of the teacher and the student networks. Note that the representations here can be modeled with either the logit information or the intermediate features. From the perspective of representation learning and information theory, the mutual information reflects the joint distribution or mutual dependence between the teacher and the student and quantifies how much information is transferred. We do agree that maximizing the mutual information between the teacher and the student is crucial for learning constructive knowledge from the teacher. We now give more detailed explanations about this.

Based on Bayes's rule, the mutual information between two paired representations can be defined as:

$$\begin{aligned} I(T;S) &= H(R(T)) - H(R(T)|R(S)) \\ &= -\mathbb{E}_T[\log p(R(T))] + \mathbb{E}_{T,S}[\log p((R(T)|R(S))] \end{aligned} \tag{3}$$

where $R(T)$ and $R(S)$ are the representations from both the teacher and the student, and $H(\cdot)$ is the entropy function. Intuitively, the mutual information is the degree of certainty in the information provided in $R(T)$ when $R(S)$ is known. Therefore, maximizing $\mathbb{E}_{T,S}[\log p((R(T)|R(S))]$ w.r.t. the parameters of the student network S increases a lower bound on mutual information. However, the true distribution of $p((R(T)|R(S))$ is unknown, instead it is desirable to estimate it by fitting a variations distribution $q((R(T)|R(S))$ to approximate the true distribution $p((R(T)|R(S))$. Then Eqn. 3 can be rewritten as:

$$\begin{aligned} I(T;S) &= H(R(T)) + \mathbb{E}_{T,S}[\log p(R(T)|R(S))] \\ &= H(R(T)) + \mathbb{E}_{T,S}[\log q((R(T)|R(S))] + \\ &\quad \mathbb{E}_S[KL(p(R(T)|R(S))||q(R(T)|R(S))] \end{aligned} \tag{4}$$

Assuming there is sufficiently expressive way of modeling q, based on Gibbs' inequality, Eqn. 4 can be updated as:

$$I(T;S) \geq H(R(T)) + \mathbb{E}_{T,S}[\log q((R(T)|R(S))] \tag{5}$$

Note that the last term in Eqn. 4 is non-negative since $KL(\cdot)$ function is non-negative and $H(R(T))$ is constant w.r.t the parameters to be optimized. By modeling q, it is easy to quantify the amount of knowledge being learned by the student. In general, q can be modeled by Gaussian distribution or Monte Carlo approximzation, noise contrastive estimation (NCE), etc. We do believe that theoretically explaining how KD works is connected to representation learning, where the correlations and higher-order output dependencies between the teacher and the student are captured. The critical challenge is how to increase the lower bounds of information, which is also pointed in [192].

In summary, we have theoretically analyzed how KD works and mentioned that the *representation* of knowledge is crucial for the transfer of knowledge and learning of the student network. One reason why explicitly dealing with the representation of knowledge from the teacher is significant and challenging, is because the knowledge from the teacher expresses much general learned information (*e.g.*feature information, logits, data usage, etc.) that is helpful for building up a keen student. In the following sections, we will provide a hierarchically-structured taxonomy for the KD methods regarding how the information is transferred for both teacher and student, how knowledge is measured, and how the teacher is defined.

4 Distillation from One Teacher

Overall insight: *Transferring knowledge from a large teacher network to a smaller student network can be achieved using either the logits or feature information from the teacher.*

4.1 Knowledge from logits

4.1.1 Softened labels and regularization

Hinton *et al*. [94] and Ba and Caruana [18] propose to shift the knowledge from teacher network to student network by learning the class distribution via softened softmax (also called 'soft labels') given in Eqn. (1). The softened labels are in fact achieved by introducing temperature scaling to increase of small probabilities. These KD methods achieved some surprising results on vision and speech recognition tasks. Recently, Mangalam *et al*. [162] introduce a special method based class re-weighting to compress U-net to a small one. Re-weighting of the classes, in fact, softens the label distribution by obstructing inherent class imbalance. Opposite to [94], Ding *et al*. [52], Hegde *et al*. [90], Tian *et al*. [224], Cho *et al*. [42] and Wen *et al*. [246] point out that the trade-off (see Eqn. 2) between the soft label and the hard label is scarcely to be optimal, and since α, β and T are fixed during training time, it lacks enough flexibility to cope with the situation without given softened labels. Ding *et al*. [52] instead propose *residual label* and *residual loss* to enable the student to use the erroneous experience in the training phase, thus preventing over-fitting and improving the performance. Similarly, Tian *et al*. [224] formulate the teacher's knowledge as structured knowledge and train a student to capture significantly more *mutual information* in contrastive learning. Hegde *et al*. [90] propose to train a *variational* student by adding sparsity regularizer based on variational inference, similar to the method in [7]. The sparsification of the student training reduces over-fitting and improves the accuracy of classification. Wen *et al*. [246] notice that the knowledge from the teacher is useful, but uncertain supervision also influences the result. Therefore, they propose to fix the incorrect predictions (knowledge) of the teacher via smooth regularization and avoid overly uncertain supervision using dynamic temperature.

On the other hand, Cho *et al*. [41], Yang *et al*. [263] and Liu *et al*. [140] focus on different perspectives of regularization to avoid under-/over-fitting. Cho *et al*. [41] discover that early-stopped teacher makes better student especially when the capacity of the teacher is larger than the student's. Stopping the training of the teacher early is akin to regularizing the teacher, and stopping knowledge distillation close to convergence allows the student to fit the training better. Liu *et al*. [140] focus on modeling the distribution of the parameters as prior knowledge, which is modeled by aggregating distribution (logits) space from the teacher network. Then the prior knowledge is penalized by a sparse recording penalty for constraining the student to avoid over-regularization. Mishra *et al*. [172] combine network quantization with model compression by training an apprentice using KD techniques and showed that the performance of low-precision networks could be significantly improved by distilling the logits of the teacher network. Yang *et al*. [263] propose snapshot distillation method to perform S-T (similar architecture) optimization in *one generation* based on a cycle learning rate policy (refer to Eqn. 2 and Eqn. 6) in which the last snapshot of each cycle (*e.g.*,W_T^{l-1} in iteration $l-1$) serves as a teacher in the next cycle (*e.g.*, W_T^l in iteration l). Thus, the idea of snapshot distillation is to extract supervision signals in earlier epochs in the same generation, meanwhile to make sure the difference between teacher and student is sufficiently large to avoid under-fitting. The snapshot distillation loss can be described as:

$$\mathcal{L}(x, W_{l-1}) = \alpha * H(y, \sigma(z_s^{l-1}; T = 1) + \beta * H(\sigma(z_t^l; T = \tau), \sigma(z_t^{l-1}, T = \tau)) \tag{6}$$

where the W_{l-1} is the weights of student at iteration $l-1$. z_s^{l-1} and z_t^{l-1} represent the logits of student and teacher at iteration $l-1$. *More detailed analysis for the methods with mutual information and one generation will be discussed in Sec. 8.*

4.1.2 Learning from noisy labels

[137], [202], [252], [256] propose methods that utilize the similar knowledge (softened labels) as in [94] but focus on data issue. Specifically, Li *et al*. [137] assume that there are a small clean dataset D_c and a large noisy dataset D_n, while Xie *et al*. [252] and Xu *et al*. [256] use both labeled and unlabeled data to improve the performance of student. In [137], the aim of distillation is to use the large amount of noisy data D_n to augment the small clean dataset D_c to learn a better visual representation and classifier. That is , the knowledge is distilled from the small clean dataset D_c to facilitate a better model from the entire noisy dataset D_n. The method is essentially different from [94] focusing on inferior model instead of inferior dataset. The same loss function in Eqn. 2 is used, however, $z_t = \sigma[f_{D_c}(x)]$, where f_{D_c} is an auxiliary model trained from the clean dataset D_c. Besides, a risk function on the unreliable label $\bar{y}$ is defined as $R_{\bar{y}} = \mathbb{E}_{D_t}[||\bar{y} - y^*||]^2$, where y^* is the unknown ground truth label and D_t is the unseen test dataset. $R_{\bar{y}}$ is an indicator that measures the level of noise in the distillation process.

Xu *et al*. [256] probe a positive-unlabeled classifier for addressing the problem of requesting the entire original training data, which can not be easily uploaded to the cloud. Besides, Xie *et al*. [252] train a *noisy* student by following three steps: 1) train a teacher model on labeled data, 2) use the teacher to generate pseudo labels on unlabeled images, and 3) train a student model on the combination of labeled images and pseudo labeled images, meanwhile injecting noise (adversarial perturbation) to the student for better generalization and robustness. In such a way, the student generalizes better than the teacher. Similarly, Sarfraz *et al*. [202] study adversarial perturbation and consider it as a crucial element in improving both generalization and robustness of student. Based on how humans learn, two learning theories for the S-T model are proposed: fickle teacher and soft randomization. The fickle teacher is to transfer the teacher's uncertainty to the student using Dropout [215] in the teacher model. The soft randomization is to improve the adversarial robustness of student model by adding Gaussian noise in the knowledge distillation. In this setting, the original distillation objective for the student in Eqn. 2 can be updated as:

$$\mathcal{L}(x + \delta, W) = \alpha * H(y, \sigma(z_s; T = 1) + \beta * H(\sigma(z_t; T = \tau), \sigma(z_s, T = \tau)) \tag{7}$$

where δ is the variation of adversarial perturbation. It is shown that using the teacher model trained on clean images to train the student model with adversarial perturbation can retain the adversarial robustness and mitigate the loss in generalization.

4.1.3 Imposing strictness in distillation

In contrast, Yang *et al*. [262], Yu *et al*. [275], Arora *et al*. [14], RKD [189] and Peng *et al*. [191] shift to a new perspective focusing more on putting *strictness* to the distillation process via optimization (*e.g*., distribution embedding, mutual relations, etc). Yang *et al*. [262] initiate to put strictness on the teacher while Yu *et al*. [275] propose two teaching metrics to impose strictness to the student. Yang *et al*. [262] observe that, except learning *primary class* (namely, the ground truth), learning *secondary class* (high confidence scores in the *dark knowledge* in [94]) may help to alleviate the risk of over-fitting of student. They thus introduce a framework of optimizing neural networks in *generations* (namely, iterations), which requires training a patriarch model M^0 only supervised by the dataset. After m generations, the student M^m is trained by m-th generation with the supervision of a teacher M^{m-1}. Since the secondary information is crucial for training robust teacher, they pick up a fixed integer K standing for the semantically similar class for each image, and then the objective is to compute the gap between the confidence scores of the primary class and other $K-1$ classes with highest scores, which can be described as:

$$\mathcal{L}(x, W^T) = \alpha * H(y, \sigma(z_t; T = 1) + \beta * [f_{a_1}^T - \frac{1}{K-1}\sum_{k=2}^{K} f_{a_k}^T] \tag{8}$$

where f_{a_k} indicates the k-th largest elements of the output (logits) z_t. Note that this S-T optimization is similar to BAN [63]; however, the goal here is to help the student learn inter-class similarity and potentially prevent over-fitting. Besides, different from [63], the teacher here is deeper and larger than the student. Yu *et al*. [275] extend [94] for metric learning by projecting the information (logits) learned from images to embedding space (called embedding networks). The embeddings are typically used to perform distance computation between data pairs of teacher and student. In this point of view, the knowledge computed based on the embedding network is the actual knowledge since it represents the data distribution. They design two different teachers: absolute teacher and relative teacher. For the absolute teacher, the aim is to minimize the distance between the teacher and student embeddings while the relative teacher is to enforce the student to learn any embedding as long as it results in the similar distance between data points. They also explore hints [94] and attention [276] to strengthen the distillation of embedding networks. *We will give more explicit explanations of these two techniques in Sec. 4.2.*

Arora *et al*. [14] propose an embedding module that captures interactions between query and document information for question answering. The embedding of the output representation (logits) includes a simple attention model with query encoder, prober history encoder, responder history encoder, and document encoder. The attention model minimizes the summation of cross-entropy loss and KL-divergence loss, inspired by Hinton *et al*. [94]. On the other hand, Wang *et al*. [234] and RKD [189] consider another type of strictness, namely the *mutual relation or relation knowledge* of the two examples in the learned representations for both teacher and student. This approach is very similar to the relative teacher in Yu *et al*. [275] since both aim to measure

TABLE 1
A taxonomy of KD methods using logits. The given equations here are the generalized objective functions, and they may vary in individual work.

Method	Sub-category	Description	KD objective function	References
KD from logits	Softened labels and regularization	Distillation using soft labels and add regularizatio to avoid under-/over-fitting	Eqn. 6	[18], [42], [90], [94], [162], [224], [246] [7], [41], [90], [140], [172], [172], [246], [263]
	Learning from noisy labels	Adding noise or using noisy data	Eqn. 6 or Eqn. 7	[137], [202], [215], [252], [256]
	Imposing strictness	Adding optimization methods to teacher or student	Eqn. 8 or Eqn. 6	[14], [63], [189], [191], [234], [262], [275]
	Ensemble of distribution	Estimating model or data uncertainty	Eqn. 9	[41], [161], [171], [192], [283]

the distance between teacher's and student's embeddings. However, RKD [189] also considers the angle-wise relational measure, similar to persevering secondary information in Yang *et al*. [262].

4.1.4 Ensemble of distribution

Although various methods have been proposed to extract knowledge from logits, some works [41], [161], [171], [283] show that KD is not always practical due to knowledge uncertainty. The performance of the student degrades when the gap between the student and the teacher is large. Malinin *et al*. [161] point out that estimating the model's uncertainty is crucial since it ensures more reliable knowledge to be transferred. They stress on the ensemble approaches to estimate the data uncertainty and distributional uncertainty. To estimate distributional uncertainty, an ensemble distribution distillation approach is proposed to not only capture the mean of ensemble soft labels but also the diversity of the distribution by annealing the temperature of the softmax. Meanwhile, Phuong *et al*. [192] propose a similar approach of matching the distribution of distillation-based muti-exit architectures, in which a sequence of feature layers is augmented with early exits at different depths. In this way, the loss defined in Eqn. 2 becomes:

$$\begin{aligned}\mathcal{L}(x,W) = \frac{1}{K}\sum_{k=1}^{K}[\alpha * H(y, \sigma(p_s^k; T=1) +\\ \beta * H(\sigma(p_t^k; T=\tau), \sigma(p_s^k, T=\tau))]\end{aligned} \tag{9}$$

where K indicates the total number of exits, and p_s^k and p_t^k represent the k-th probabilistic output at exit k.

On the other aspect, [14], [32], [56], [63], [123], [127], [171], [186], [187], [194], [203], [213], [220], [222], [231], [247], [264], [274], [283] propose to add more teachers or other auxiliaries such as teaching assistant and small students to improve the robustness of ensemble distribution. We will explicitly analyze these approaches in the following Sec. 7.

4.1.5 Summary and open challenges

Table. 1 summarizes the KD methods using logits or 'soft labels'. We divide these methods into four categories. In overall, distillation using logits needs to transfer the dark knowledge to avoid over-/under-fitting carefully. Meanwhile, the gap of model capacity between teacher and student is also very crucial for effective distillation. Besides, the drawbacks of learning from logits are obvious. First, the effectiveness of distillation only limits to softmax loss and relies on the number of classes. Second, it is impossible to apply these methods to the KD problems in which there are no labels (*e.g.*, low-level vision).

Open challenges: The original idea in [94] is in its apparent generality: any student can learn from any teacher; however, it is shown that this promise of generality is hard to be achieved on some datasets [41], [276] (*e.g.*, IamgeNet [49]) even though regularization or strictness techniques are applied. When the capacity of the student is too low, it is hard for the student to emulate the logits information of the teacher successfully. Therefore, it is expected to improve the generality and provide a better representation of logits information, which can be easily absorbed by the student.

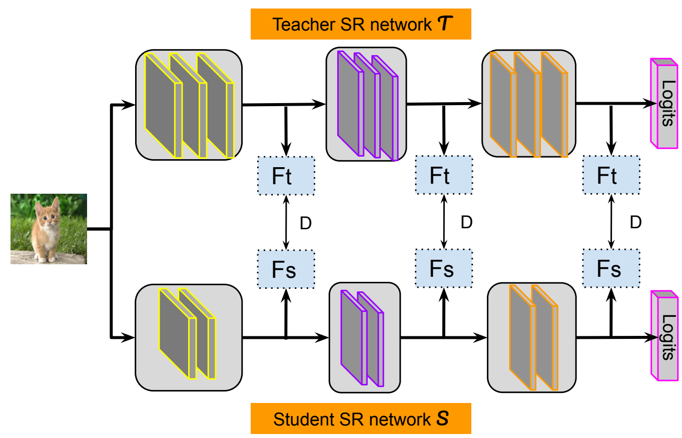


Fig. 3. An illustration of general feature-based distillation.

4.2 Knowledge from the intermediate layers

Overall insight: *Feature-based distillation enables learning richer information from teacher and provides more flexibility for performance improvement.*

Except distilling knowledge from the softened labels, Romero *et al*. [196] initially introduce *hint* learning rooted from [94]. A hint is defined as the outputs of a teacher's hidden layer, which is helps guide the student's learning process. The goal of student learning is to learn a feature representation that is the optimal prediction of teacher's intermediate representations. Essentially, the function of hints is a form of regularization; therefore, a pair of hint and guided (a hidden layer of the student) layers have to be carefully chosen such that the student is not over-regularized. Inspired by [196], many endeavours have been taken to study how to choose, transport and match the hint layer (or layers) and the guided layer (or layers) via various layer transform (*e.g.*, transformer [91], [115]) and distance (*e.g.*, MMD [103]) metrics. Generally, the hint learning objective can be written as:

$$\mathcal{L}(F_T, F_S) = D(TF_t(F_T), TF_s(F_S)) \tag{10}$$

Where F_T and F_S are the selected hint and guided layers of teacher and student. TF_t and TF_s are the transformer or regressor functions for the hint layer of teacher and guided layer of student. $D(\cdot)$ is the distance function(*e.g.*, l_2) measuring the similarity of hint and the guided layers.

Fig. 3 depicts the general paradigm of feature-based distillation. It is shown that various intermediate feature representations can be extracted from different positions and are transformed with a certain type of regressor or transformer. The similarity of the transformed representations is finally optimized via some distance metrics D (*e.g.*, L_1 or L_2 distance). In this paper, we carefully scrutinize various design considerations of feature-based KD methods and summarize four key factors that are usually considered: *transformation of the hint, transform of the guided layer, position of selected distillation feature and distance metric* [91]. In the following parts, we will analyze and categorize all existing feature-based KD methods concerning these four aspects.

4.2.1 Transformation of hints

As pointed in [7], the knowledge of teacher should be easy to learn by the student. To do this, teacher's hidden features are usually converted by a transformation function T_t. Note that transformation of teacher's knowledge is very crucial step for feature-based KD since there is risk of information missing in the process of transformation. The transformation methods of teacher's knowledge in AT [115], MINILM [241], FSP [270], ASL [133], Jacobian [214], KP [284], SVD [128], SP [229], MEAL [210], KSANC [31] and NST [103] cause the knowledge missing due to the reduction of feature dimension. Specifically, AT [115] and MINILM [241] focus on attention mechanisms (*e.g.*, self-attention [230]) via an attention transformer T_t to transform the activation tensor $F \in \mathbb{R}^{C\times H\times W}$ to C feature maps $F \in \mathbb{R}^{H\times W}$. FSP [270] and ASL [133] calculate the information flow of the distillation based on Gramian matrices, through which the tensor $F \in \mathbb{R}^{C\times H\times W}$ is transformed to $G \in \mathbb{R}^{C\times N}$, where N represents the number of matrices. Jacobian [214] and SVD [128] map the tensor $F \in \mathbb{R}^{C\times H\times W}$ to $G \in \mathbb{R}^{C\times N}$ based on Jacobians via first-order Taylor series and truncated SVD, respectively, thus inducing information losing. KP [284] projects $F \in \mathbb{R}^{C\times H\times W}$ to M feature maps $F \in \mathbb{R}^{M\times H\times W}$, causing lose of knowledge. Similarly, SP [229] proposes a similarity-preserving knowledge distillation based on the observation that semantically similar inputs tend to elicit similar activation patterns. To achieve this goal, the teacher's feature $F \in \mathbb{R}^{B\times C\times H\times W}$ is transformed to $G \in \mathbb{R}^{B\times B}$, where B is the batch size. Intuitively, the G encodes the similarity of the activations at the teacher layer, however, it leads to the information loss in transformation. MEAL [210] and KSANC [31] both use *pooing* to align the intermediate map of the teacher and student, thus there is information loss when transforming teacher's knowledge. NST [103] and PKT [190] match the distributions of neuron selectivity patterns or the affinity of data samples between teacher and student networks. The loss functions are based on minizing the maximum mean discrepancy (MMD) and Kullback-Leibler (KL) divergence between these distributions respectively, thus causing information loss when selecting neurons.

On the other hand, FT [115] proposes to extract good *factors* through which transportable features are made. The transformer TF_t is called *paraphraser* and the transformer TF_s is called *translator*. To extract the teacher factors, an adequately trained paraphraser is needed. Meanwhile, to enable the student to assimilate and digest the knowledge according to its own capacity, a user-defined paraphrase ratio is used in the paraphraser to control the factor of the transfer. Heo *et al*. [92] use the original teacher's feature in the form of binarized values, namely via a separating hyperplane (activation boundary (AB)) that determines whether neurons are activated or deactivated. Since AB only considers the activation of neurons, not the magnitude of neuron response, thus there is information loss in the feature binarization process. Similar information loss happens in IRG [140], where the teacher's feature space is transformed to vertex and edge in graph representation where relationship matrices are calculated. IR [4] distills the internal representations of the teacher model to the student model, however, since multiple layers in the teacher are compressed into one layer of the student, there is information loss when matching the features. Heo *et al*. [91] design TF_t with a margin ReLU function to exclude the negative (adverse) information and to allow using the positive (beneficial) information. The margin m is determined based on batch normalization [105] after 1×1 convolution in student's transformer TF_s.

Conversely, FitNet [196], RCO [108], Chung *et al*. [45], Wang *et al*. [240], Kulkarni *et al*. [120] do not add additional transformation to the teacher's knowledge, thus no information loses from teacher's side. However, not all knowledge included in the teacher is beneficial for the student. As pointed by Heo *et al*. [91], features include both adverse and beneficial information, thus it is important to impede the use of adverse information and avoid missing the beneficial information.

4.2.2 Transformation of the guided features

On the other aspect, the transformation TF_s of the guided features (namely, student transform) of the student is also an important step for effective KD. Interestingly, the SOTA works such as AT [276], MINILM [241], FSP [270], Jacobian [214], FT [115], SVD [128], SP [229], KP [284], IRG [140], RCO [108],MEAL [210], KSANC [31], NST [103], Kulkarni *et al*. [120] and Aguilar *et al*. [4] use the TF_s same as the TF_t, which means the same amount of information might lose in both transformations of the teacher and the student.

On the contrary, different from the transformation of teacher, FitNet [94], AB [92], Heo *et al*. [91] and VID [7] do

TABLE 2
A taxonomy of distilling knowledge from the intermediate layers (feature maps). KP incidates knowledge projection.

Method	Teacher's TF_t	Student's TF_s	Distillation position	Distance metric	Lost knowledge
FitNet [196]	None	1×1 Conv	Middle layer	L_1	None
AT [276]	Attention map	Attention map	End of layer group	L_2	Channel dims
KP [284]	Projection matrix	Projection matrix	Middle layers	L_1 + KP loss	Spatial dims
FSP [270]	FSP matrix	FSP matrix	End of layer group	L_2	Spatial dims
FT [115]	Encoder-decoder	Encoder-decoder	End of layer group	L_1	Channel + Spatial dims
AT [276]	Attention map	Attention map	End of layer group	L_2	Channel dimensions
MINILM [241]	Self-ttention	Self-attention	End of layer group	KL	Channel dimensions
Jacobian [214]	Gradient penalty	Gradient penalty	End of layer group	L_2	Channel dims
SVD [270]	Truncated SVD	Truncated SVD	End of layer group	L_2	Spatial dims
VID [7]	None	1×1 Conv	Middle layers	KL	None
IRG [140]	Instance graph	Instance graph	Middle layers	L_2	Spatial dims
RCO [108]	None	None	Teacher's train route	L_2	None
SP [229]	Similarity matrix	Similarity matrix	Middle layer	Frobenius norm	None
MEAL [210]	Adaptive pooling	Adaptive pooling	End of layer group	$L_{1/2}$/KL/L_{GAN}	None
Heo [210]	Margin ReLU	1×1 Conv	Pre-ReLU	Partial L_2	Negative features
AB [92]	Binarization	1×1 Conv	Pre-ReLU	Margin L_2	feature values
Chung [45]	None	None	End of layer	L_{GAN}	None
Wang [240]	None	Adaptation layer	Middle layer	Margin L_1	Channel + Spatial dims
KSANC [31]	Average pooling	Average pooling	Middle layers	L_2 + L_{GAN}	Spatial dims
Kulkarni [120]	None	None	End of layer group	L_2	None
IR [4]	Attention matrix	Attention matrix	Middle layers	KL+ Cosine	None
Liu [140]	Transform matrix	Transform matrix	Middle layers	KL	Spatial dims
NST [103]	None	None	Intermediate layers	MMD	None

to change the dimension of teacher's feature representations and design TF_s with a 'bottleneck' layer (1×1 convolution) to make student's feature to match the dimension with the teacher. Note that Heo *et al*. [91] add one batch normalization layer after 1×1 convolution to calculate the margin of the proposed margin ReLU transformer of the teacher. There are some advantages of using 1×1 convolution in KD. First, it offers a channel-wise pooling without a reduction of spatial dimensionality. Second, it can be used to create a one-to-one linear projection of a stack of feature maps. Lastly, the projection created by 1×1 convolution can also be used to directly increase the number of feature maps in the distillation model. In such a case, the feature representation of student does not decrease but rather increase to match teacher's representation, which does not cause information loss in the transformation of the student.

Exceptionally, some works focus on a different aspect of the transformation of student's feature representations. Wang *et al*. [240] make the student imitate the fine-grained local feature regions close to object instances of teacher's representations. This goal is achieved by designing a particular adaptation function TF_s to fulfill the imitation task. IR [4] aims to let student acquire the abstraction in a hidden layer of the teacher by matching the internal representations. That is, the student is taught to know how to compress the knowledge from multiple layers of the teacher into a single layer of it. In such a setting, the transformation of the student's guided layer is done by a self-attention transformer. Chung *et al*. [45], on the other hand, propose to impose no transformation to both student and teacher, but rather add a discriminator to distinguish the feature map distributions of different networks (teacher or student).

4.2.3 Distillation positions of features

In addition to the transformation of teacher's and student's features, distillation position of the selected features is also very crucial in many cases. Earlier, FitNet [94], AB [92] and Wang *et al*. [240] use the end of an arbitrary middle layer as the distillation point, however it is shown to have poor distillation performance. Based on the definition of layer group [277], in which a group of layers have same spatial size, AT [276], FSP [270], Jacobian [214], MEAL [210], KSANC [31] and Kulkarni *et al*. [120] determine the distillation point at the end of each layer group, in contrast to FT [115] and NST [103] where the position lies only at the end of last layer group. Compared to FitNet, FT achieves better results since it focuses on more informational knowledge. Whereas IRG [140] considers all the above-mentioned critical positions, namely the distillation position lies not only in the end of earlier layer group but also in the end of the last layer group. Interestingly, VID [7], CRO [108], Chung *et al*. [45], SP [229], IR [4] and Liu *et al*. [140] generalize the selection of distillation positions by employing variational information maximization [20], curriculum learning [23], adversarial learning [74], similarity-presentation in representation learning [100], muti-task learning [47], reinforcement learning [173]. We will discuss more for these methods in later sections.

4.2.4 Distance metric for measuring distillation

The quality of KD from teacher to student is usually measured by various distance metrics. The most commonly used distance function is based on L_1 or L_2 distance. FitNet [196], NST [276], FSP [270], SVD [128], RCO [108], FT [115] and KSANC [31] are mainly based on L_2 distance, whereas MEAL [210], Wang *et al*. [240] and Kulkarni *et al*. [120] mainly use L_1 distance. On the other hand, Liu *et al*. [140]

and IR *et al.* [4] utilize KL-divergence loss to measure feature similarities. Besides, a cosine-similarity loss is adopted by IR [4] and RKD [189] to regularize the context representation on the feature distributions of teacher and student.

Besides, some works also resort to the adversarial loss for measuring the quality of KD. MEAL [210] shows that the student learning the distilled knowledge with discriminators is optimized better than the original model, and the student can learn the distilled knowledge from a teacher model that has arbitrary structures. Among the works focusing on feature-based distillation, KSANC [108] adds an adversarial loss at the last layer of both teacher and student networks, while MEAL [210] adds multi-stage discriminators in the position of every extracted feature representation. It is worth mentioning that using adversarial loss has shown great potential in improving the performance of KD. We will explicitly discuss the existing KD techniques based on adversarial learning in the following Sec. 5.

4.2.5 Potentials and open challenges

Table. 5 summarizes the existing feature-based KD methods. It is shown that most works employ feature transformations for both teacher and student. $L1$ or $L2$ loss is the most commonly used loss for measuring KD quality. A natural question one may ask is why we can not directly match the features of teacher and student? What's wrong with it? If we consider the activation of each spatial position as one feature, the flattened activation map of each filter is a sample of the space of selected neurons with dimension HW, which reflects how DNN learns an image [103]. Thus, when matching distribution, it is less considerable to directly match the samples since the sample density might lose in the space, as pointed in [196].

Potentials: Feature-based methods show more generalization capability and quite promising results. In the following, more flexible ways of determining the representative knowledge of features are expected. The approaches used in representation learning (*e.g.*, parameter estimation, graph models) might be reasonable solutions for these problems. Besides, neural architecture search (NAS) techniques may better handle the selection of features. Furthermore, feature-based KD methods are potential for cross-domain transfer and low-level vision problems.

Open challenges: Although we have discussed all existing feature-based methods, it is still hard to say which one is better. The reason is that information may lose in different aspects; however it is difficult to measure. Besides, most works randomly choose intermediate features as knowledge, and it is less intuitive why they can be the representative knowledge among all layers. Third, the distillation position of features is manually selected based on the network or task. Fourth, multiple features may not represent better knowledge than that of a single layer. Therefore, better ways to choose knowledge from layers and to represent knowledge could be exploited.

5 DISTILLATION VIA ADVERSARIAL LEARNING

Overall Insight: *GAN can help learn the correlation between classes and preserve the multi-modality of S-T framework, especially when student has relatively small capacity.*

In Sec. 4, we have discussed the two most popular approaches for KD. However, the key problem is that it is difficult for the student to learn the true data distribution from the teacher since the teacher normally can not perfectly model the real data distribution. Generative adversarial networks (GAN) [44], [74], [237], [238], [239] has been proved to be potential to learn the true data distribution in image translation. To this end, recent works [5], [22], [35], [45], [70], [72], [93], [98], [135], [141], [144], [147], [149], [195], [209], [210], [242], [244], [245], [259], [260], [278] have tried to explore adversarial learning to improve the performance for KD. These works are, in fact, built on three fundamental prototypes of GANs [74], [137], [170]. Therefore, we first formulate the principle of these three types of GANs, as illustrated in Fig. 4, and then analyze and categorize the existing GAN-based KD methods.

5.1 A basic formulation of GANs in KD

The first type of GAN, as shown in Fig. 4(a), is initially proposed to generate continuous data by training a generator G and a discriminator D penalizing the generator G for producing implausible results. The generator G produces synthetic examples $G(z)$ (*e.g.*, images) from the random noise z sampled using a certain distribution [74], which are fed to the discriminator D along with the real examples sampled from real data distribution $p(x)$. The discriminator D tries to distinguish the two inputs, and both generator G and discriminator D improve their respective abilities in a minmax game until the discriminator D is unable to distinguish the fake from the real. The objective function can be written as follows:

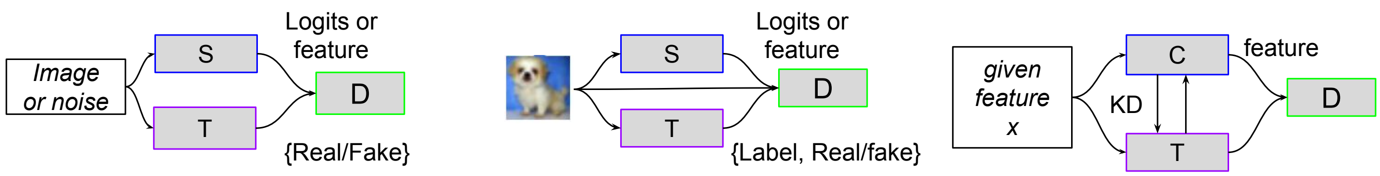
$$\min_G \max_D J(G,D) = \mathbb{E}_{x\sim p(x)}[log(D(x))] + \mathbb{E}_{z\sim p(z)}[log(1-D(G(z)))] \tag{11}$$

where $p_z(z)$ is the distribution of noise (*e.g.*, uniform or normal).

The second type of GAN for KD is built on conditional GAN (CGAN) [106], [170], [237], [239] as shown in Fig. 4(b). CGAN is trained to generate samples from a class conditional distribution c. The generator is replaced by some useful information rather than random noise. Hence, the objective of the generator is to generate realistic data, given the conditional information. Mathematically, the objective function can be written as:

$$\min_G \max_D J(G,D) = \mathbb{E}_{x\sim p(x)}[log(D(x|c))] + \mathbb{E}_{z\sim p(z)}[log(1-D(G(z|c)))] \tag{12}$$

Different from above-mentioned GANs, Triple-GAN [137] (the third type) introduces a three-player game where there are a classifier C, a generator G and a discriminator D as shown in Fig. 4(c). Adversarial learning of generators and discriminator overcomes some difficulties in [74], such as not being optimal optimization and generator failing to control the semantics of generated samples, etc. If we assume there is a pair of data (x,y) from the true distribution $p(x,y)$. After a sample x is sampled from $p(x)$, C assigns a pseudo label y following the conditional distribution $p_c(y|x)$, that is, C characterizes the conditional distribution $p_c(y|x) \approx p(y|x)$.The generator aims to model the conditional distribution in the other direction $p_g(x|y) \approx p(x|y)$,



(a) conventional GAN (b) conditional GAN (c) KDGAN

Fig. 4. An illustration of GAN-based KD methods. (a) KD based on conventional GAN [74]; (b) KD based on conditional GAN (CGAN) [170]; (c) KD based on TripleGAN [44].

while the discriminator distinguishes whether a pair of data (x,y) is from true distribution $p(x, y)$ or not. Thus the minmax game can be formulated as:

$$\begin{aligned}\min_{C,G}\max_{D} J(C, G, D) = \mathbb{E}_{(x,y)\sim p(x,y)}[log(D(x, y))]+ \\ \alpha\, \mathbb{E}_{(x,y)\sim p_c(x,y)}[log(1 - D(x, y))]+ \\ (1 - \alpha)\, \mathbb{E}_{(x,y)\sim p_g(x,y)}[log(1 - D(G(y, z), y))]\end{aligned} \tag{13}$$

where α is a hyper-parameter controlling the relative importance of C and G.

5.2 How GAN boosts KD?

Based on the above formulation of GANs, we now analyze how they are applied to boost the performance of KD with S-T learning.

5.2.1 KD based on the conventional GAN (first type)

Chen *et al*. [35] and Fang *et al*. [57] focus on distilling the knowledge of *logits* from teacher to student via the first type of GAN, as depicted in Fig. 4(a). [1] There are several benefits of predicting logits based on the discriminator. First, the learned loss, as described in Eqn 11, can be effective as image translation tasks like [106], [237], [239]. The second benefit is closely related to the multi-modality of the network output; therefore, it is not necessary to exactly mimic the output of one teacher network to achieve good student performance as usually done in [94], [196]. However, the low-level feature alignment is missing since the discriminator only captures the high-level statistics of teacher and student outputs (logits).

In contrast, Belagiannis *et al*. [22], Liu *et al*. [144], Hong *et al*. [98], Aguinaldo *et al*. [5], Chung *et al*. [45], Wang *et al*. [244], Wang *et al*. [245], Chen *et al*. [34] and Li *et al*. [135] aim to distinguish whether the *features* comes from teacher or student via adversarial learning, which effectively pushes the two distributions close to each other. [2] The reason for using the features of teacher and student as inputs to the discriminator is their *dimensionality*. The feature representations extracted from teacher are high-level abstract information and are easy for classification, which leads to a low probability for the discriminator to make a mistake [144]. However, the GAN training in this setting is sometimes not stable and even hard to converge, especially when the model capacity between student and teacher is big. To address this problem, some regularization techniques such as dropout [215] or l_2 or l_1 regularization [22] are added to Eqn. 11 for confining weights.

1. [35], [57] are data-free KD methods, which will be explicitly discussed in Sec. 9.

2. Note that [45] use LSGAN [163] loss, and [244] use WGAN-GP loss [76] to stabilize training.

5.2.2 KD based on CGAN (second type)

Xu *et al*. [258] and Yoo *et al*. [273] employ CGAN [170] for KD, where discriminator is trained to distinguish whether the *label distribution* (logits) is from the teacher or the student. The student, which is regarded as the generator, is adversarially trained to fool the discriminator. Liu *et al*. [147] also exploit CGAN for compressing image generation networks; however, the discriminator predicts the class label of teacher and student together with an auxiliary classifier (AC-GAN) [182].

Whereas, Roheda *et al*. [195], Zhai *et al*. [278], Li *et al*. [135], Chen *et al*. [34] and Liu *et al*. [149] focus on discriminating the *feature* space of teacher and student under CGAN framework. Interestingly, Chen *et al*. [34] deploy two discriminators, namely the teacher and student discriminators for compressing image translation networks.To avoid model collapse, Liu *et al*. [149] use Wassertein loss [76] to stabilize training.

5.2.3 KD based on TripleGAN (third type)

Different from the distillation methods based on conventional GAN and CGAN, Wang *et al*. [242] propose a three-player game named KDGAN consisting of a classifier (student), a teacher and a discriminator (similar to the prototype in TripleGAN [137]), as shown in Fig. 4(c). The classifier and the teacher learn from each other via distillation losses and are adversarially trained against the discriminator via adversarial loss in Eqn. 13. By simultaneously optimizing the distillation and adversarial loss, the classifier (student) will learn the true data distribution at equilibrium.

5.3 Summary and open challenges

In Table 3, we summarize existing GAN-based knowledge distillation methods regarding the field of application, input features of discriminator D, the number of discriminators used, and whether it is one-stage (without the teacher to be trained first). In general, most methods focus on classification tasks based on the first type of GAN (conventional GAN) [74] and use features as the inputs to the discriminator D. Besides, it is worth noting that most methods use only one discriminator for discerning the student from the teacher; however, some works such as [45], [244] and [34] employ multiple discriminators in their KD frameworks. On the other hand, one can see that most methods follow a two-stage KD paradigm where the teacher is trained first, and then knowledge is transferred to the student via KD

TABLE 3
A taxonomy of KD based on adversarial learning.

Method	GAN type	Purpose	Inputs of D	Number of D	Online KD
Chen [35]	First type	Classification	Logits	One	No
Belagiannis [22]	First type	Classification	Features	One	No
Liu [144]	First type	Classification Object detection	Features	One	No
Hong [98]	First type	Object detection	Features	Six	No
Wang [245]	First type	Classification	Features	One	No
Aguinaldo [5]	First type	Classification	Features	One	No
Chung [45]	First type (LSGAN [163])	Classification	Features	Two/Three	Yes
Wang [244]	First type(WGAN-GP [76])	Image generation	Features	One/Multiple	Yes
Chen [34]	First/Second type	Image translation	Features	Two	No
Liu [149]	Second type (WGAN-GP [76])	Semantic segmentation	Features	One	No
Xu [259]	Second type	Classification	Logits	One	No
Roheda [195]	Second type	Cross-domain surveillance	Features	One	Yes
Zhai [278]	Second type (BicyleGAN [291])	Image translation	Features	One	Yes
Liu [147]	Second (AC-GAN [182])	Image translation	Features	One	No
Wang [242]	Third type	Image translation	Features	One	No
Li [135]	First/Second type	Image translation	Features	One	No
Fang [57]	First type	Classification Semantic segmentation	Logits	One	No
Yoo [273]	Second type	Classification	Logits	One	No

loss. In contrast, some works such as [45], [195], [244], [278] also exploit online (one-stage) KD relieving the necessity of pre-training teacher networks. *We will provide more detailed analysis for the KD methods w.r.t online/two-stage distillation and image translation in Sec. 8 and in Sec. 15.5, respectively.*

Open challenges: The first challenge for GAN-based KD is the stability of training, especially when the capacity between teacher and student is big. Second, it is less intuitive that using only logits or only features or both as inputs to discriminator is good since there lacks theoretical support. Third, the advantages of using multiple discriminators is less clear and what features in which position are suitable for training GAN.

6 DISTILLATION WITH GRAPH REPRESENTATIONS

Overall insight: *Graphs are the most typical locally connected structures that better capture the features and hierarchical patterns for KD.*

Up to now, we have categorized and analyzed the most common KD methods using either logits or feature information. However, one critical issue about KD is data. Generally, training a DNN needs embedding a high-dimensional dataset to facilitate data analysis. Thus, the optimal goal of training a teacher model is not only to transform the training dataset into a low-dimensional space but also to analyze the intra-data relations [129], [151]. However, most KD methods did not consider intra-data relations. Here, we first introduce the definition of basic concepts of graph embedding and knowledge graph based on [28], [85]. We provide an analysis of existing graph-based KD methods and discuss some new perspectives about KD.

6.0.1 Notation and definition

Definition 1. *A **graph** can be depicted as $\mathcal{G} = (V, E)$, where $v \in V$ is a node and $e \in E$ is an edge. A graph $\mathcal{G}$ is associated with node type mapping function F_v: $V \rightarrow \mathcal{T}^v$ and an edge type mapping function F_e: $E \rightarrow \mathcal{T}^e$.*

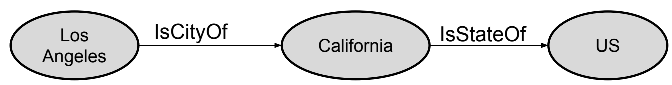


Fig. 5. An example of knowledge graph.

where $\mathcal{T}^v$ and $\mathcal{T}^e$ denote the node types and edge types, respectively. For any $v_i \in V$, there exists a particular mapping type: $F_v(v_i) \in \mathcal{T}^v$. The similar mapping comes to any $e_{ij} \in E$, which is mapped as $F_e(e_{ij}) \in \mathcal{T}^e$, where i and j indicate i-th and j-th nodes.

Definition 2. *A **homogeneous graph (directed graph)** can be depicted as $\mathcal{G}_{hom} = (V, E)$ is a type of graph in which $|\mathcal{T}^v| = |\mathcal{T}^e| = 1$. All nodes in this graph embedding belong to a single type and all edges are in one single type.*

Definition 3. *A **knowledge graph** defined as $\mathcal{G}_{kn} = (V, E)$ is an instance of directed heterogeneous graph whose nodes are entities and edges are subject-property-object triplet. Each edge has the form head entity, relation, tail entity, denoted as $< h, r, t >$, indicating a relationship from head h to tail t.*

$h, t \in V$ are entities and $r \in E$ is the relation. Hereby, we note $< h, r, t >$ a triplet for knowledge graph. An example is shown in Fig. 5, the knowledge graph includes two triplets $< h, r, t >$: $< LosAngeles, IsCityOf, California >$ and $< California, isStateOf, US >$.

Graph neural networks. A graph neural network (GNN) is a type of DNN that directly operates on the graph structure. A typical application is about node classification [204]. In node classification problem, i-th node v_i is characterized by its feature x_{v_i} and ground truth t_{v_i}. Thus, given a labeled

TABLE 4
A summary of notations used in Sec. 6.

Notations	Descriptions
$\lvert \cdot \rvert$	The cardinally of a set
$\mathcal{G} = (V, E)$	Graph $\mathcal{G}$ with a set of node V and set of edge E
v_i, e_{ij}	A node $v_i \in V$ and an edge e_{ij} linking v_i and v_j
$\mathbf{x}_{v_i}$, $\mathbf{x}_{e[v_i]}$	Features of v_i and features of edges of v_i
$\mathbf{h}_{ne[v_i]}$, $\mathbf{x}_{ne[v_i]}$	Features of states and of neighboring nodes of v_i
$F_v(v_i)$, $F_e(e_{ij})$	Mapping of node type v_i and edge type e_{ij}
$\mathcal{T}^v$, $\mathcal{T}^e$	The set of node types and set of edge types
$< h, r, t >$	Head, relation and tail in knowledge graph
N	Number of nodes in the graph
$\mathbf{h}_{v_i}$	Hidden state of i-th node v
f_t, f_o	local transition and output functions
F_t, F_o	global transition and output functions
$\mathbf{H}$, $\mathbf{O}$, $\mathbf{X}$	Stack of all hidden states, outputs, features
$\mathbf{H}^t$	Hidden state of t-th iteration of $\mathbf{H}$

graph $\mathcal{G}$, the goal is to leverage the labeled nodes to predict the unlabeled. It learns to represent each node with a d dimensional vector state h_{v_i} containing the information of its neighborhood. Specifically speaking, h_{v_i} can be mathematically described as [289]:

$$\mathbf{h}_{v_i} = f_t(\mathbf{x}_{v_i}, \mathbf{x}_{co[v_i]}, \mathbf{h}_{ne[v_i]}, \mathbf{x}_{ne[v_i]}) \tag{14}$$

$$\mathbf{o}_{v_i} = f_o(\mathbf{h}_{v_i}, \mathbf{x}_{v_i}) \tag{15}$$

where $\mathbf{x}_{co[v_i]}$ denotes the feature of the edges connecting with v_i, $\mathbf{h}_{ne[v_i]}$ denotes the embedding of the neighboring nodes of $\mathbf{v}_i$, and $\mathbf{x}_{ne[v_i]}$ denotes the features of the neighboring nodes of v_i. The function f_t is a transition function that projects these inputs onto a d-dimensional space, and f_o is the local output function that produces output. Note that f_t and f_o can be interpreted as the feedforward neural networks. If we denote $\mathbf{H}$, $\mathbf{O}$, $\mathbf{X}$ and $\mathbf{X}_N$ as the concatenation of outputs of stacking all the states, all the outputs, all the features, and all the node features, respectively, then $\mathbf{H}$ and $\mathbf{O}$ can be formulated as:

$$\mathbf{H} = F_t(\mathbf{H}, \mathbf{X}) \tag{16}$$

$$\mathbf{O} = F_o(\mathbf{H}, \mathbf{X}) \tag{17}$$

where F_t is the global transition function and F_o is the global output function. Note that F_t and F_o are the stacked functions of f_t and f_o in all nodes V in the graph.

Since we are aiming to get a unique solution for $\mathbf{h}_{v_i}$, in [204], [289], a neighborhood aggregation algorithm is applied, such that:

$$\mathbf{H}^{t+1} = F_t(\mathbf{H}^t, \mathbf{X}) \tag{18}$$

where $\mathbf{H}^t$ denotes t-th iteration of $\mathbf{H}$.

Given any initial state $\mathbf{H}(0)$, $\mathbf{H}^{t+1}$ in Eqn. 18 convergences exponentially to the solution in Eqn. 16. Based on the framework, f_t and f_o can be optimized via supervised loss when the target information t_v^i is known:

$$\mathcal{L} = \sum_{i=1}^{N} (t_v^i - o_v^i) \tag{19}$$

where N is the total number of supervised nodes in the graph.

6.1 Graph-based distillation

Based on the above explanation regarding the fundamentals of graph representations and GNN, we now delve into the existing graph-based distillation techniques. To our knowledge, Liu *et al.* [142] first introduce a graph modeling approach for the visual recognition task in the video. Since the action is the video is modeled initially as a bag pf visual-words (BoVW), which is sensitive to visual changes. However, some higher-level features are shared across views and enable connecting the action models of different views. To better capture the relationship from two vocabularies, they construct a bipartite graph $\mathcal{G} = (V, E)$ to partition them into visual-word clusters. Note that V is the union of vocabulary V_1 and V_2, and E are the weights attached to nodes. In such a way, the knowledge from BoVW can be transferred to visual-word clusters, which are more discriminative in the presence of view changes. Luo *et al.* [159] consider to incorporates rich, privileged information from a large-scale multimodal dataset in the source domain, and improves the learning in the target domain where training data and modalities are scarce. Regarding using S-T structures for KD, to date, there are several works such as [124], [129], [151], [160], [169], [236], [266].

GKD [124] and IRG [151] consider the geometry of the respective feature spaces by reducing intra-class variations, which allow for dimension-agnostic transfer of knowledge. This perspective is the opposite of Liu *et al.* [151] and RKD [189]. Specifically, instead of directly trying to explore the mutual relation between data points in student and teacher, GKD [124] regards this relation as a geometry of data space (see Fig. 6(a)). Given a batch of inputs $\mathbf{X}$, we can compute the inner representation $\mathbf{X}_l^S = [\mathbf{x}_l^S, \mathbf{x} \in \mathbf{X}$ and $\mathbf{X}_l^T = [\mathbf{x}_l^T, \mathbf{x} \in \mathbf{X}$ at layer l ($l \in \Lambda$) of teacher and student networks. Using cosine similarity metric, these representations can be used to build a k-nearest neighbor similarity graph for teacher $\mathcal{G}_l^T(\mathbf{X}) =< \mathbf{X}_l^T, \mathbf{W}_l^T >$ and for student $\mathcal{G}_l^S(\mathbf{X}) =< \mathbf{X}_l^S, \mathbf{W}_l^S >$. Note that $\mathbf{W}_l^T$ and $\mathbf{W}_l^S$ represent the edge weights, which represent the similarity between the i-th and j-th elements of $\mathbf{X}_l^T$ and $\mathbf{X}_l^S$. Based on graph representation for both teacher and student, the KD loss in Eqn. 10 can be updated as follows:

$$\mathcal{L} = \sum_{l \in \Lambda} D\left(\mathcal{G}_l^S(\mathbf{X}), \mathcal{G}_l^T(\mathbf{X})\right) \tag{20}$$

where distance metric D is based on L_2 distance. While IRG [151] essentially is similar to GKD [124] regarding the construction of the graph, however, IRG also takes into account the instance graph transformations. The aim of introducing feature space transformation across layers is because there might be too tight or dense constraints and descriptions fitting on the teacher's instance features at intermediate layers. Basically, the transformation of instance relation graph is composed of vertex transformation and edge transformation from $l_1 l$-th layer to l_2-th layer, as shown in Fig. 6 (b). Thus the loss in Eqn. 20 can be extended to:

$$\begin{aligned}\mathcal{L} = \sum_{l \in \Lambda} D_1(\mathcal{G}_l^S(\mathbf{X}), \mathcal{G}_l^T(\mathbf{X})) + \\ D_2\left((\Theta_T(\mathcal{G}_l^S(\mathbf{X})), \Theta_S(\mathcal{G}_l^T(\mathbf{X}))\right)\end{aligned} \tag{21}$$

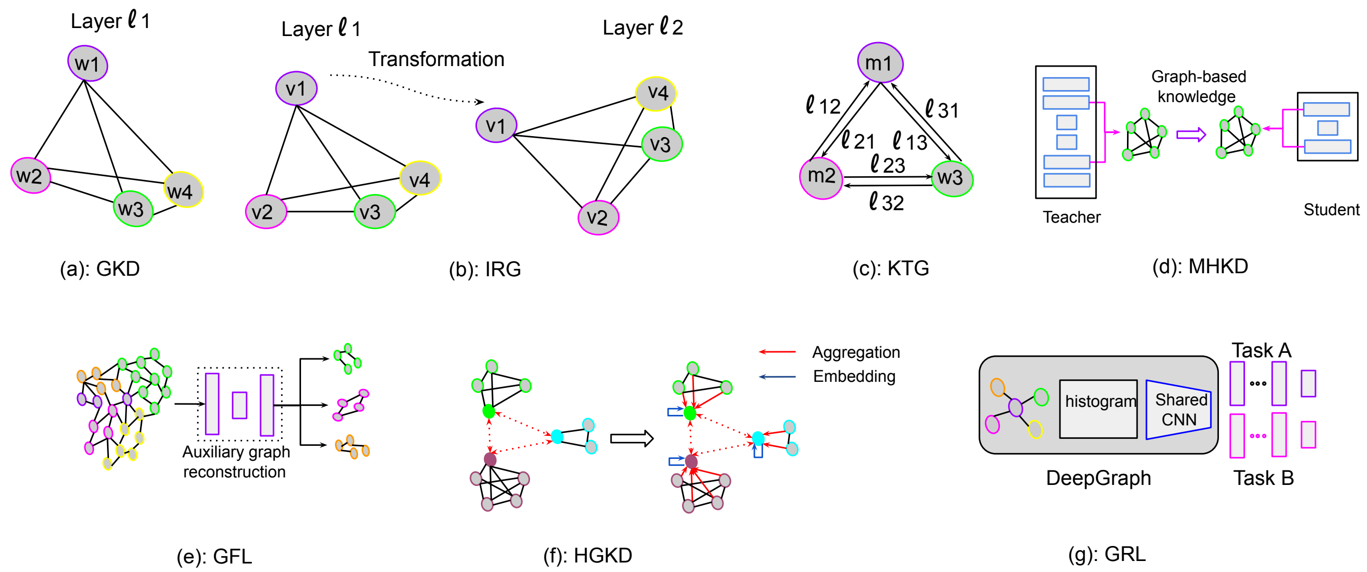


Fig. 6. An graphical illustration of graph-based KD methods. GKD [124], IRG [151], KTG [169], MHKD [129] all focus on graph-based knowledge distillation for model compression. GFL [266] and HGKD [236] aim to improve semi-supervised node classification via graph-based knowledge transfer, whereas GRL [160] exploits graph-based knowledge for multi-task learning.

TABLE 5
A summary of KD methods via graph representations.

Method	Purpose	Graph type	Knowledge type	Distance metric	Graph embedding
GKD [124]	Model compression	Heterogeneous graph	Layer-wise feature	L_2	GSP [183]
IRG [151]	Model compression	Knowledge graph	Middle layers	L_2	Instance relations
MHKD [129]	Model compression	Knowledge graph	Middle layers	KL	SVD [128] + Attention
KTG [169]	Model compression	Directed graph	Network model	L_1 + KP loss	–
GFL [266]	Few-shot learning	GNN	Class of nodes	Frobenius norm	HGR [272]
HGKD [236]	Few-shot learning	GNN	Class of nodes	Wasserstein	GraphSAGE [84]
GRL [160]	Multi-task leaning	GNN	Class of nodes	Cross-entropy	HKS [132]

where Θ_T and Θ_S are the transformation functions for teacher and student, and D_1 and D_2 are the distance metrics for instance relation and instance translation.

MHKD [129] is a method that enables distilling the data-based knowledge from teacher network to a graph using an attention network (see Fig. 6(d)). Similarly to IRG [140], feature transformation is also considered to capture the intra-data relations. The KD loss is based on KL-divergence loss using the embedded graphs from teacher and student. KTG [169] also exploits graph representation; however, it focuses on a different perspective of KD. The knowledge transfer graph provides a unified view of KD and has the potential to represent diverse knowledge patterns. Interestingly, each node in the graph represents a direction of knowledge transfer. On each edge, a loss function is defined for transferring knowledge between two nodes linked by each edge. Thus, combining different loss functions can represent collaborative knowledge learning with pair-wise knowledge transfer. Fig. 6(c) shows the knowledge graph of diverse collaborative distillation with three nodes, $L_{s,t}$ represents the loss function used for the training node.

On the other aspect, GFL [266], HGKT [236], GRL [160] and MHGD [129] all resort to GNN for the purpose of KD. HGKT and GFL focus on transfer the knowledge from seen classes to unseen classes in few-shot learning [117], [218]. GFL [266] leverages the knowledge learned the auxiliary graphs to improve semi-supervised node classification in the target graph. As shown in Fig. 6(e), GFL learns the representation of a whole graph and ensures a similarly structured knowledge to be transferred. The auxiliary graph reconstruction is achieved by using a graph autoencoder. While HGTK aims to build a heterogeneous graph focusing on transferring intra-class and inter-class knowledge simultaneously. Inspired by way of modeling class distribution in adversarial learning [74], [82], [93], [237], [242], [259], in which instances with same class are expected to have the same distribution, the knowledge is transferred from seen classes to new unseen classes based on learned aggregation and embedding functions, as shown in Fig. 6 (f). GRL [160] builds a multi-task KD method for representation learning based on DeepGraph [132].The knowledge is based on GNN to map the raw graphs to the metric values. The learned graph metrics are then used as auxiliary tasks, and the knowledge of the network is distilled into graph representations (see Fig. 6(g)). The graph representation structure is learned via a CNN by feeding the graph descriptor to it. Denote pairs of graph and graph-level labels as $\{(G_i, y_i)\}_{i=1}^{N}$, where $G_i \in \mathcal{G}$, $y_i \in \mathcal{Y}$, and $\mathcal{G}$, $\mathcal{Y}$ are the cardinally of all possible graphs and labels respectively. Then, the loss for learning the model parameters are described as:

$$\mathcal{L} = \mathbb{E}[D(y_i, f(G_i; \theta))] \tag{22}$$

where θ are the model parameters.

6.2 Open challenges

Graph representations are of great importance for tackling KD problems since they better capture the hierarchical patterns in locally connected structures. However, there are some challenges. First, graph representations are hard to generalize since they are limited to structured data or a specific type of data. Second, It is still challenging to more appropriately measure graph distances since existing distance measure (*e.g.*, l_2) may not fit well. Third, layer-wise distillation is difficult to achieve in graph KD since graph representation models and network structures in such cases are quite limited.

7 Distillation from multiple teachers

Overall insight: *The student can learn better knowledge from multiple teachers, which are more informative and instructive than a single teacher.*

While impressive progress has been achieved under the common S-T KD paradigm where the knowledge is transferred from one high-capacity teacher network to a student network, the capacity of knowledge in this setting is quite limited [187] and the knowledge diversity is scarce for some special cases, such as cross-model KD [279]. To this end, some works probe to learn a portable student from *multiple* teachers or ensemble of teachers. The intuition behind this can be explained in analogous to the cognitive process of human learning. In practice, a student does not solely learn from a single teacher but learn a concept of knowledge better provided with the instructive guidance from multiple teachers on the same task or heterogeneous teachers on different tasks. In such a way, the student can amalgamate and assimilate various illustrations of knowledge representations from multiple teacher networks and build a comprehensive knowledge system [203], [208], [274]. As a result, many new KD methods [11], [56], [62], [63], [89], [110], [123], [127], [140], [143], [148], [157], [165], [171], [186], [187], [194], [197], [198], [203], [207], [208], [213], [217], [220], [222], [227], [232], [247], [248], [250], [265], [268], [274], [279], [283], [290], [293] have been proposed. Although these works vary in various distillation scenarios and assumptions, they share some standard characteristics, which can be categorized into five types: ensemble of logits, ensemble of feature-level information, unifying data sources, obtaining sub-teacher networks from single teacher network, customizing student network from heterogeneous teachers and learning a student network with diverse peers via the ensemble of logits. We now explicitly analyze each category and provide insights on how and why they are valuable for the problems.

7.1 Distillation from the ensemble of logits

Model ensemble of logits is one of the popular methods in KD from multiple teachers as shown in Fig. 7(a). In such a setting, the student is encouraged to learn the softened output of the assembled teachers' logits (dark knowledge) via the cross-entropy loss as done in [11], [56], [63], [110], [123], [143], [148], [165], [171], [186], [220], [222], [227], [248], [265], [274], [290], which can be generalized into

$$\mathcal{L}_{Ens}^{logits} = H(\frac{1}{m}\sum_{i}^{m} N_{T_i}^{\tau}(x), N_S^{\tau}(x)) \tag{23}$$

where m is the total number of teachers, H is the cross-entropy loss, $N_{T_i}^{\tau}$ and N_S^{τ} are i-th teacher's and student's logits (or softmax ouputs), and τ is the temperature. The averaged softened output serves as the incorporation of multiple teacher networks in the output layer. Minimizing Eqn. 23 achieves the goal of KD at this layer. Note that the averaged softened output is more objective than any of the individuals since it can mitigate the unexpected bias of the softened output existing in some input data.

Different from the methods as mentioned above, [62], [123], [197], [250], [279] argue that taking the average of individual prediction may ignore the diversity and importance variety of the member teachers of an ensemble. Thus, they propose to learn the student model by imitating the summation of the teacher's predictions with a gating component. Thus, Eqn. 23 becomes as:

$$\mathcal{L}_{Ens}^{logits} = H(\sum_{i}^{m} g_i N_{T_i}^{\tau}(x), N_S^{\tau}(x)) \tag{24}$$

where g_i is the gating parameter. In Ruder *et al*. [197], the $g_i = sim(D_{S_i}, D_T)$ of their respective source domain D_S and target domain D_T.

Summary: Distilling knowledge from the ensemble of logits mainly depends on taking the average or the summation of individual teacher's logits. Taking the average alleviates the unexpected bias; however, it may ignore the diversity of individual teachers of an ensemble. The summation of logits of each teacher can be balanced by the gating parameter g_i, however, how to determine better the value of g_i is worth being studied in further work.

7.2 Distillation from the ensemble of features

Distillation from the ensemble of feature representations is more flexible and advantageous than from the ensemble of logits since they can provide more affluent and diverse information to the student. However, distillation from the ensemble of features [140], [165], [187], [217], [247], [279], [290] is more challenging since each teacher's feature representation at specific layer is different from that of the other. Hence, how to transform the features and form an ensemble of teachers' feature-map-level representations becomes the key problem, as illustrated in Fig. 7(b).

Regarding this knotty problem, Park *et al*. [187] propose to feed the student's feature map into some non-linear layers (called transformer), and the output is trained to mimic the final feature map of the teacher network. In this way, the advantages of general model ensemble and feature-based KD methods, as mentioned in Sec. 4.2, can both be incorporated. The loss function is depicted as:

$$\mathcal{L}_{Ens}^{fea} = \sum_{i}^{m} || \frac{x_{T_i}}{||x_{T_i}||_2} - \frac{TF_i(x_S)}{||TF_i(x_S)||_2} ||_1 \tag{25}$$

where x_{T_i} and x_S are i-th teacher's and student's feature maps respectively, and TF is the transformer (*e.g.*, 3×3

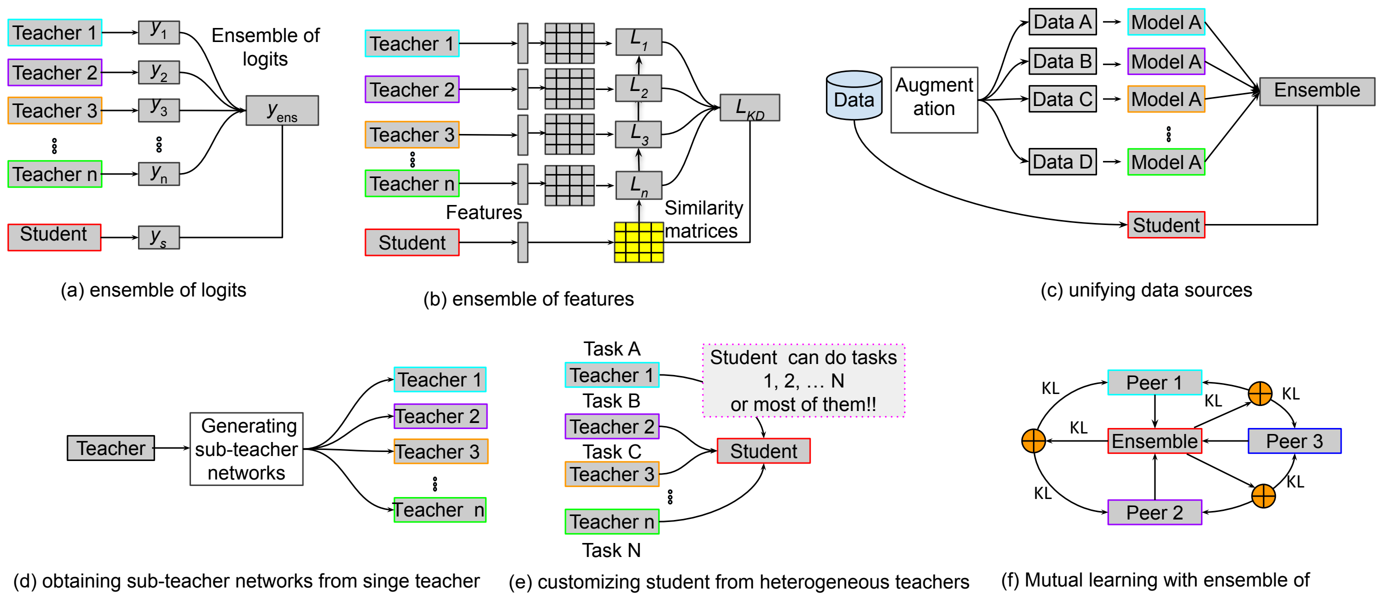


Fig. 7. Graphical illustration for KD with multiple teachers. The KD methods can be categorized into five types: (a) KD from the ensemble of logits; (b) KD from the ensemble of feature representations via some similarity matrices; (c) unifying various data sources from the same network (teacher) model A to generate various teacher models; (d) obtaining hierarchical or stochastic sub-teacher networks given one teacher network; (e) training a versatile student network from multiple heterogeneous teachers; (f) online KD from diverse peers via ensemble of logits.

convolution layer) used for the adaptation of student's feature with that of teacher.

Differently, Wu *et al*. [247] and Liu *et al*. [140] propose to let the student model imitate the learnable transformation matrices of the teacher models. This approach is, in fact, an updated version of a single teacher model [229]. For i-th teacher and student network in [247], the similarity between feature maps is computed based on Euclidean metric as:

$$\mathcal{L}_{Ens}^{fea} = \sum_{i}^{m} \alpha_i || \log(A_S) - log(A_{T_i})||_F^2 \qquad (26)$$

where α_i is the teacher weight for controlling the contribution of i-th teacher and α_i should satisfy $\sum_i^m \alpha_i = 1$, and A_S and A_{T_i} are the similarity matrices of the student and i-th teacher, which can be computed by $A_S = x_S^\intercal x_S$ and $A_T = x_{T_i}^\intercal x_{T_i}$, respectively.

Open challenges: Based on our review, there are just a few works that propose to distill knowledge from the ensemble of feature representations. Although [187], [248] propose to let the student directly mimic the ensemble of feature maps of the teachers via either non-linear transformation or similarity matrices with weighting mechanisms, there still exist some challenges. First of all, how can we know which teacher's feature representation is more reliable or more influential among the ensemble? Second, how to determine the weighting parameter α_i for each student in an adaptive way? Third, instead of summing all feature information together, is there any mechanism of selecting the best feature map of one teacher from the ensemble as the representative knowledge?

7.3 Distillation by unifying data sources

Although above-mentioned KD methods using multiple teachers are good in some aspects, however, they assume that the target classes of all teacher and student models are the same. Besides, the dataset used for training is often scarce, and the teacher models with high capacity are also limited. To tackle these problems, some recent works [73], [194], [203], [232], [247], [250] propose data distillation by unifying data sources from multiple teachers as illustrated in Fig. 7(c). The goal of these methods is to generate labels for the unlabelled data via various data processing approaches (*e.g.*, data augmentation) to train a student model.

Vongkulbhisal *et al*. [232] propose to unify an ensemble of *heterogeneous classifiers* (teachers) which may be trained to classify different sets of the target classes and can share the same network architecture. To generalize distillation, a probabilistic relationship connecting the outputs of the heterogeneous classifiers with that of the unified (ensemble) classifier is proposed. Similar to [232], Wu *et al*. [247] and Gong *et al*. [73] also explore to transfer the knowledge from teacher models trained in existing data to a student model by using unlabelled data to form a decision function.

Besides, some works utilize the potential of data augmentation approaches to build multiple teacher models from a trained teacher model. Radosavovic *et al*. [194] propose a distillation method via *multiple transformations* on the unlabeled data to build diverse teacher models sharing the same network structure. The technique follows four steps. First, a single teacher model is trained on manually labeled data. Second, the trained teacher model is applied to multiple transformations on the unlabelled data. Third, the predictions on the unlabelled data are converted to an ensemble of numerous predictions. Fourth, the student model is trained on the union of the manually labeled data and automatically labeled data. Sau *et al*. [203] propose an approach to simulate the effect of multiple teachers by injecting noise to the training data and perturbing the logit outputs of a teacher. In such a way, the perturbed outputs not only simulate the setting of multiple teachers but also result in noise in the softmax layer, thus regularizing the

TABLE 6
A taxonomy of KD with multiple teachers. L_{CE} is for cross-entropy loss; L_{Ens} is for the KD loss between the ensemble teacher and the student, L_{KD} indicates the KD loss between individual teacher and the student; $L_{KD_{fea+logits}}$ means KD loss using feature and logits; KL indicates KL divergence loss for mutual learning; L_{GAN} is for adversarial loss; MMD means mean maximum discrepancy loss; L_{reg} is the regression loss; N/A means not available. *Note that the losses summarized are generalized terms, which may vary in individual work.*

Method	Ensemble Logits	Ensemble of Features	Unifying data sources	Customize student	Extending teacher	Online KD	Mutual learning	Major Loss functions
Anil [11]	✓	✗	✗	✗	✗	✓	✓	L_{CE}+L_{Ens}
Chen [32]	✓	✓	✗	✗	✗	✓	✓	L_{CE}+L_{Ens}+L_{KD}
Dvornik [56]	✓	✗	✗	✓	✗	✓	✓	L_{CE}+L_{Ens}+L_{KD}
Fukuda [62]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{Ens}
Furlanello [63]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{Ens}
He [89]	✗	✓	✗	✗	✓	✗	✗	L_1+L_{KD}
Jung [110]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{KD}
Lan [123]	✓	✗	✗	✗	✗	✓	✗	L_{CE}+L_{Ens}
Lee [127]	✓	✓	✗	✗	✓	✓	✓	N/A
Liu [140]	✗	✓	✗	✓	✗	✓	✗	L_{CE}+L_{KD}
Luo [157]	✓	✓	✗	✓	✗	✗	✗	L_{CE}+$L_{KD_{fea+logits}}$
Zhou [290]	✓	✓	✗	✓	✗	✗	✗	MMD+L_{KD}
Mirzadeh [171]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{KD}
Papernot [186]	✓	✗	✓	✗	✗	✓	✗	L_{KD}
Park [187]	✗	✓	✗	✗	✗	✗	✗	L_{CE}+L_{KD}
Radosavovic [194]	✓	✗	✓	✗	✗	✗	✗	N/A
Ruder [197]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{KD}
Ruiz [198]	✓	✗	✗	✗	✓	✗	✗	L_{CE}+L_{Ens}
Sau [203]	✓	✗	✓	✗	✗	✗	✗	L_2 (KD)
Shen [207]	✓	✓	✗	✓	✗	✗	✗	L_{KD}+L_{PL}
Shen [208]	✓	✓	✗	✓	✗	✗	✗	$L_{KD_{fea+logits}}$+L_{reg}
Song [213]	✓	✗	✗	✗	✓	✓	✗	L_{CE}+L_{KD}
Tarvaninen [222]	✓	✗	✗	✗	✗	✓	✗	L_{KD}
Tran [227]	✓	✗	✗	✗	✓	✗	✓	L_{CE}+L_{Ens}+KL
Vongkulbhisal [232]	✓	✗	✓	✗	✗	✗	✗	L_{CE}+L_{Ens}
Wu [247]	✗	✓	✓	✗	✗	✗	✗	L_{KD}
Wu [248]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{Ens}
Yang [265]	✓	✗	✗	✗	✗	✗	✗	L_{CE}+L_{KD}
Ye [268]	✓	✓	✗	✓	✗	✗	✗	L_{KD}+L_{KD}
You [274]	✓	✓	✗	✗	✗	✗	✗	L_{CE}+$L_{KD_{fea+logits}}$
Zhang [279]	✓	✓	✗	✗	✗	✗	✗	L_{CE}+$L_{KD_{fea+logits}}$
Zhang [283]	✓	✗	✗	✗	✗	✓	✓	L_{CE}+KL
Zhu [293]	✓	✗	✗	✗	✗	✓	✓	L_{CE}+KL+L_{Ens}
Chung [45]	✓	✓	✗	✗	✗	✓	✓	L_{CE}+L_{GAN}+KL
Kim [114]	✓	✓	✗	✗	✗	✓	✓	L_{CE}+L_{Ens}+KL
Hou [101]	✗	✓	✗	✗	✗	✓	✗	L_{CE}+L_{Ens}
Xiang [250]	✓	✗	✓	✗	✗	✗	✗	L_{CE}+L_{KD}

distillation loss.

Summary: Unifying data source using data augmentation techniques and unlabeled data from a single teacher model to build up multiple sub-teacher models are also valid for training a student model. However, it requires a high-capacity teacher with more generalized target classes, which could confine the application of these techniques. Meanwhile, for some low-level vision problems, whether these techniques are effective or not should be further studied based on feature representations.

7.4 From a single teacher to multiple sub-teachers

Up to now, it has been shown that the student could be further improved with multiple teachers, which are used as an ensemble or separately. However, using multiple teacher networks is resource-heavy and delays the training process. Following this, some methods [89], [127], [197], [213], [227], [248], [274] have been proposed to generate multiple sub-teachers from a single teacher network as shown in Fig. 7(d). Lee *et al.* [127] propose stochastic blocks and skip connections to a teacher network so that the effect of multiple teachers can be obtained in the same resource as a single teacher network. The sub-teacher networks still have reliable performances since there exists a valid path for each batch. By doing so, the student can be trained with multiple teachers in the entire training phase. Similarly, Ruiz *et al.* [198] introduce hierarchical neural ensemble by employing a *binary-tree* structure to share a subset of intermediate layers between different models. The scheme allows controlling the inference cost by one-the-fly deciding how many branches to evaluate. Tran *et al.* [227], Song *et al.* [213] and He *et al.* [89] introduced *multi-headed architectures* to build up multiple teacher networks while amortizing the computation through a shared heavy-body network. Each head is assigned to an ensemble member and tries to mimic the *individual predictions* of the ensemble member.

Open challenges: Although network ensemble using stochastic or deterministic methods can achieve the effect of multiple teachers and online KD, however, there still exist many uncertainties. First of all, how many teachers are sufficient for online distillation? Second, which structure is better from the ensemble of sub-teachers? Third, how to balance the training efficiency and accuracy of student network? These challenges are worth being explored in further works.

7.5 Customizing student form heterogeneous teachers

In many cases, the well-trained deep networks (teachers) are focused on different tasks and are optimized for different datasets. However, most works focus on training a student by distilling knowledge from teacher networks on the same task or one the same dataset. To tackle these problems, *knowledge amalgamation* has been initialized by recent works [56], [68], [140], [157], [199], [207], [208], [268], [269], [290] to learn a versatile student model by distilling knowledge from the expertise of all teachers as illustrated in Fig. 7(e). Shen *et al*. [208], Ye *et al*. [268], Luo *et al*. [157] and Ye *et al*. [269] propose to train a student network by customizing the tasks without accessing human-labelled annotations. These methods all rely on some schemes such as branch-out [6] or selective learning [64]. The merits of these methods lie in that they allow for reusing pretrained deep networks trained on various datasets for diverse tasks to build a tailored student model based on user's demand. The student inherits most of the capability of heterogeneous teachers and thus can perform multiple tasks at a time. Shen *et al*. [207] and Gao *et al*. [68] utilize a similar methodology but focus on the same task (classification) with two teachers specialized in different classification problems. In such a way, the student is capable of handling the comprehensive or fine-grained classification. Differently, Dvornik *et al*. [56] attempted to learn a student that can predict unseen classes by distilling knowledge from teachers via few-shot learning. Rusu *et al*. [199] proposed a multi-teacher single-student policy distillation methods that can distill multiple policies of reinforcement learning agents to a single student network for sequential prediction tasks.

Open challenges: The works, as mentioned above, have shown great potential for customizing a versatile student network for various tasks; however, there are some limitations in such methods. First, the student may not be compact since there are branch-out structures. Second, the current techniques mostly require teachers to share similar network structures (*e.g.*, encoder-decoder), which confines the generalization of such methods. Third, training might be complicated since some works adopt a dual-stage strategy and follow many steps with fine-tuning. These challenges also point out the future of knowledge amalgamation.

7.6 Mutual learning with ensemble of peers

One problem of the conventional KD methods using multiple teachers is about their computation cost and complexity since they require pre-trained high-capacity teachers with two-stage (also called offline) learning. To simplify the distillation process, one-stage (online) KD methods [11], [32], [45], [101], [114], [123], [279], [283], [293] have been developed, as shown in Fig. 7(f). Instead of pre-training a static teacher model, these methods train a set of student models simultaneously by learning from each other in a peer-teaching manner. There are some benefits of such methods. First, these approaches merge the training processes of teachers and student models and use peer networks to provide the teaching knowledge. Second, these online distilling strategies can improve the performance of any-capacity models, leading to a more generic application. Third, such a peer-distillation method can sometimes outperform the teacher-based two-stage KD methods. For the KD with mutual learning, the distillation loss of *two peers* is based on KL divergence, which can be formulated as:

$$\mathcal{L}_{Peer}^{KD} = KL(z_1, z_2) + KL(z_2, z_1) \tag{27}$$

where KL is KL divergence function, and z_1 and z_2 are the predictions of peer one and peer two.

Besides, Lan *et al*. [123] and Chen *et al*. [32] also construct a multi-branch variant of a given target (student) network by adding auxiliary branches to create a local ensemble teacher (also called group leader) model from all branches. Each branch is trained with a distillation loss, which aligns the prediction of that branch with the teacher's prediction. Mathematically, it distillation loss can be formulated by minimizing the KL divergence of z_e (prediction of the ensemble teacher) and prediction z_i of i-th branch peer:

$$\mathcal{L}_{Ens}^{KD} = \sum_{i=1}^{m} KL(z_e, z_i) \tag{28}$$

where the prediction $z_e = \sum_{i=1}^{m} g_i z_i$. g_i is the weighting score or attention-based weights [32] of i-th branch peer z_i.

Although most of these methods only consider using logits information, some works also exploit feature information. Chung *et al*. [45] propose feature-map-level distillation by employing adversarial learning (discriminators). Kim *et al*. [114] introduce a feature fusion module to form an ensemble teacher; however, the fusion is based on concatenation of the features (output channels) from the branch peers. Moreover, Liu *et al*. [140] present a knowledge flow framework which moves the knowledge from the features of multiple teacher networks to a student.

Summary: Compared to two-stage KD methods using pre-trained teachers, distillation from student peers has many merits. The methods are built based on mutual learning of peers and sometimes on the ensemble of peers. Most works rely on logits information; however, some works also exploit feature information via adversarial learning or feature fusion. There is still room for improvement in this direction. For instance, how many peers are most optimal for the KD process? Besides, when the teacher is available, is it possible to use both online and offline methods simultaneously? Lastly, how to reduce the computation cost without the sacrifice of accuracy and generalization? We will discuss the advantages and disadvantages of online and offline KD in following Sec. 8.

7.7 Potentials

Table. 6 summarizes the KD methods with multiple teachers. Overall, most methods rely on the ensemble of logits; however, the knowledge of feature representations has not been excavated too much. Therefore, it is the potential to

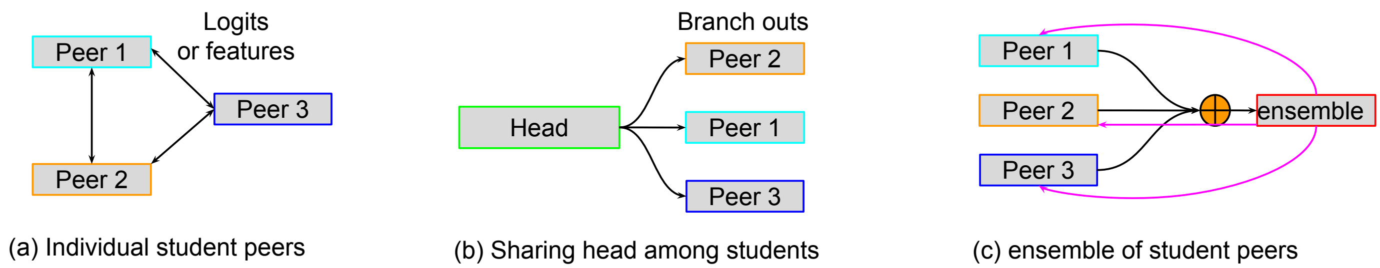


Fig. 8. An illustration of online KD methods. (a) online KD with individual student peer learning from each other; (b) online KD with student peers sharing trunk (head) structure; (c) online KD by assembling the weights of each student to form a teacher or group leader.

exploit the knowledge of the ensemble of feature representations by designing better gating mechanisms. Unifying data sources and extending teacher models are two effective methods for reducing individual teacher models; however, the performances are degraded. Thus, how to overcome this issue deserves more research. Third, customizing a versatile student is a valuable idea; however, existing methods are limited by network structures, diversity, and computation cost, which need to be improved in the following works.

8 Online Distillation

Overall insight: *With the absence of a pre-trained powerful teacher, simultaneously training a group of student models by learning from peers' predictions is an effective substitute for two-stage (offline) KD.*

In this section, we provide deeper analysis for online (one-stage) KD methods in contrast to previously mentioned offline (two-stage) KD methods. The offline KD methods often require pre-trained high-capacity teacher models to perform one-way transfer [7], [62], [69], [94], [101], [115], [139], [176], [274]. However, it is sometimes hard to get these 'good' teachers, and the performance of the student is degraded when the network capacity between teacher and student is huge. Besides, two-stage KD requires many parameters with high computation cost. To overcome these difficulties, some works focus more on *online* KD, which simultaneously trains a group of student peers by learning from peers' predictions.

8.1 Individual student peers

Zhang *et al*. [283], Gao *et al*. [69] and Anil *et al*. [11] focus on online *mutual learning* [283] (also called codistilation) in which a pool of untrained student networks with the same network structure simultaneously learn the target task together. In such a peer-teaching environment, each student learns the average class probabilities from the other (see Fig. 8(a)). Although Chung *et al*. [45] also employ individual student, they additionally design a feature map-based KD loss via adversarial learning. Hou *et al*. [101] propose Dual-Net, where two individual student classifiers are fused into a fused classifier. During training, the two student classifiers are locally optimized, while the fused classifier is globally optimized as a way of mutual learning. Other methods, such as [46], [176], focus on online video distillation by periodically update the weights of the student based on the output of the teacher. Although codistillation achieves parallel learning of students, [11], [45], [101], [283] do not consider the ensemble of peers' information as done in other works such as [32], [69].

8.2 Sharing blocks among student peers

Considering the training cost of employing individual students, some works propose to share some network structures (*e.g.*, head sharing) of the students with branches as shown in Fig. 8(b). In Song *et al*. [213], Lan *et al*. [123], the student peers are built upon the multi-branch architectures [219]. In such a way, all structures together with the shared trunk layers (often use head layers) can construct individual student peers, and *any* target student peer network in the whole multi-branch can be optimized.

8.3 Ensemble of student peers

While using codistillation and multi-architectures can facilitate online distillation, the knowledge from all student peers is missing. To this end, some works [32], [69], [114], [123], [139] propose to assemble knowledge (logits information) of all student peers to build a one-the-fly teacher or group leader, which is in turn distilled back to all student peers to enhance the student learning in a closed-loop form as shown in Fig. 8(c). Note that in ensemble distillation, the student peers can either be independent or share the same head structure (trunk). The ensemble distillation loss is mentioned in Eqn. 24 of Sec. 7 where a gating component g_i is added to balance the contribution of each student. In Chen *et al*. [32], the gating component g_i is obtained based on self-attention mechanism [230].

8.4 Summary and open challenges

Summary: Based on the above analysis, we have figured out that codistillation, muti-architectures, and ensemble learning are the three main techniques for online distillation. There are some advantages of online KD compared with offline KD. First of all, it removes the pre-training of large teachers. Second, online learning provides a simple but effective way to improve the learning efficiency and generalization ability of the network by training together with other student peers. Third, online learning with student peers often achieves better performances than offline learning.

Open challenges: There are some challenges to online KD. First, there lacks theoretical analysis for why online learning is sometimes better than offline learning. Second, in online ensemble KD, simply aggregating students' logits to form an ensemble teacher restrains the diversity of student peers, thus limiting the effectiveness of online learning. Third, existing methods are confined to the problem in which ground truth (GT) labels exist (*e.g.*classification), however for some problems(*e.g.*, low-level vision problems), how to ensemble the student peers to form an effective ensemble teacher still needs to be exploited.

9 DATA-FREE DISTILLATION

Oberal insight: *Can we achieve KD when the original data for teacher or (un)labelled data for training student is not available?*

One major limitation of most KD methods such as [94], [187], [189], [196] is that they assume the training samples of the original networks (teachers) or of target networks (students) are available. However, the training dataset is sometimes unknown in real-world applications due to the privacy and transmission issues [153]. To handle this problem, data-free KD paradigms [24], [35], [57], [87], [121], [153], [167], [179], [267], [271], [273] are newly developed. A taxonomy of these methods are summarized in Table. 7 and detailed technical analysis is provided as follows.

9.1 Distillation based on metadata

To our knowledge, Lopes *et al*. [153] initially propose to reconstruct the original training dataset using only teacher model and its *metadata* recorded in the form of *precomputed activation statistics*. Thus, the goal is to find the set of images whose representation best matches the one given by the teacher network. Gaussian noise is randomly passed as input to the teacher, and the gradient descent (GD) is made to minimize the difference between the metadata and the representations of noise input. To better constrain the reconstruction, the metadata of *all* layers of the teacher model are used and recorded to train the student model with high accuracy. Bhardwaj *et al*. [24], however, demonstrate that metadata from a *single* layer (average-pooling layer) using k-means clustering is sufficient to achieve high student accuracy. In contrast to [24], [153] requiring sampling the activations generated by real data, Haroush *et al*. [87] propose to use the metadata (*e.g.*, channel-wise mean and standard deviation) from Batch Normalization (BN) [105] layer with synthetic samples. The objective of metadata-based distillation can be formulated as:

$$X^* = \arg \min_{X \sim R^{H \times W}} L(\Phi(X), \Phi_0) \tag{29}$$

where X^* is the image (with width W and height H) to be found, Φ is the representation of X, Φ_0 is the representation of metadata, and L is the loss function (*e.g.*, l_2).

9.2 Distillation based on class-similarities

Nayak *et al*. [179] argue that the metadata used in [24], [153] are actually not complete data-free approaches since the metadata is formed using training data itself. They instead propose a zero-shot KD approach in which no data samples and no metadata information are used. In particular, the approach obtains useful prior information about the underlying data distribution in the form of *class similarities* from the model parameters of the teacher. The prior information can further be utilized for crafting data samples (also called data impressions (DIs)) via modeling the output space of the teacher model as a Dirichlet distribution. The class similarity matrix, similar to [229], is calculated based on the softmax layer of the teacher model. The objective of data impression X_i^k can be formulated based on cross-entropy loss:

$$X_i^k = \arg \min_{X} L_{CE}(y_i^k, T(X, \theta_T, \tau)) \tag{30}$$

where y_i^k is sampled i-th softmax vector and k is certain class.

9.3 Distillation using generator

Considering the limitation of metadata and similarity-based distillation methods, some works [35], [57], [167], [267], [271], [273] propose novel data-free KD methods via adversarial learning [74], [237], [239]. Although the tasks and network structures vary in these methods, most are built on a common framework. That is, the pretrained teacher network is fixed as a discriminator, while a generator is designed to synthesize training samples given various input sources (*e.g.*, noise [35], [267], [271], [273]). However, slight differences exist in some works. Fang *et al*. [57] point out the problem of taking teacher as the discriminator since the information of student is ignored, and generated samples can not be customized without student. Following this, they initialize to take both teacher and student as the discriminator to reduce the discrepancy between them while a generator is trained to generate some samples to enlarge the discrepancy. In contrast, Ye *et al*. [267] focus more on strengthening the generator structure in which three generators are designed and subtly used. Specifically, first, a group-stack generator is trained to generate the images originally used for pre-training the teachers and also the intermediate activations. Then a dual generator takes the generated image as the input and the dual part is taken as the target network (student) and regrouped for multi-label classifications. To compute the adversarial loss for both the generated image and the intermediate activations, multiple group-stack discriminators (multiple teachers) are also designed to amalgamate multi-knowledge into the generator. In Yoo *et al*. [273], the generator takes two inputs: a sampled class label y and noise z. Meanwhile, a decoder is also applied to reconstruct the noise input z' and class label y' from the fake data x' generated by the generator from noise input z and class label y. Thus by minimizing the errors between y and y' and between z and z', the generator generates more reliable data. Although the adversarial loss is not used in [271], however, the generator (called DeepInversion) taking an image prior regularization term to synthesize images is modified from DeepDream [175].

9.4 Open challenges for data-free distillation

Although data-free KD methods have shown great potential and pointed out the new direction for KD, there still exist many challenges. First of all, the recovered images are still unrealistic and with low-resolution, which may not be utilized in some data-captious tasks (*e.g.*, semantic segmentation). Second, training and computation of the existing methods might be complicated due to the utilization of many modules. Third, diversity and generalization of the

TABLE 7
A taxonomy of data-free knowledge distillation.

Method	Original data needed	Metadata or prior info.	Number of generators	Inputs	Discriminator	Multi-task distillation
Lopes [153]	✓	Activations of all layers	✗	Image shape	✗	✗
Bhardwaj [24]	✓	Activations of pooling layer	✗	Image shape	✗	✗
Haroush [87]	✓	Batch normalization layer	✗	Image shape	✗	✗
Nayak [179]	✗	Class similarities	✗	Class label+ Number of DIs	✗	✗
Chen [35]	✗	✗	One	Noise	Teacher	✗
Fang [57]	✗	✗	One	Noise/images	Teacher + student	✗
Ye [267]	✗	✗	Three	Noise	Teachers	✓
Yoo [273]	✗	✗	One	Noise + class labels	Teacher	✗
Yin [271]	✗	✗	One	Noise	Teacher	✗
Micaelli [167]	✗	✗	One	Noise	Teacher	✗

recovered data are still limited compared with the methods of data-driven distillation. Forth, whether such methods are effective for low-level tasks (*e.g.*, image super-resolution) needs to be further studied.

10 DISTILLATION WITH A FEW DATA SAMPLES

Overall insight: *How to perform efficient knowledge distillation with only a small amount of training data?*

Most KD methods with S-T structures, such as [45], [94], [115], [187], are based on matching information (*e.g.*, logits, hints) and optimizing the KD loss with the fully annotated large-scale training dataset. As a result, the training is still data-heavy and processing-inefficient. To enable efficient learning of student while using small amount of training data, some works [19], [116], [121], [136], [146] propose few-sample KD strategies. The technical highlight of these methods is generating pseudo training examples or aligning the teacher and the student with layer-wise estimation metrics.

10.1 Distillation via pseudo examples

Insight: *If training data is insufficient, try to create pseudo examples for training student.*

Under the condition that a large amount of training data is scarce, easily leading to overfitting of student network, [116], [121], [146] focus on creating pseudo training examples. Specifically, Kimura *et al*. [116] adopt the idea of inducing points [212] to generate pseudo training examples, which are then updated by applying adversarial examples [75], [219] and further optimized by an imitation loss. Liu *et al*. [146] generate pseudo ImageNet [49] labels from a teacher model (trained with ImageNet) and also utilize the semantic information (*e.g.*, words) to add supervision signal for the student. Interestingly, Kulkarni *et al*. [121] create a 'mismatched' unlabeled stimulus (*e.g.*, soft labels of MNIST dataset [125] provided by the teacher trained on CIFAR dataset [118]), which are used as for augmenting a small amount of training data to train the student.

10.2 Distillation via layer-wise estimation

Insight: *Layer-wise distillation from the teacher network via estimating the accumulated errors on the student network can also achieve the purpose of few-example KD.*

In Bai *et al*. [19] and Li *et al*. [136], the teacher network is first compressed to create student via network pruning [292], and then layer-wise distillation losses are applied to reduce the estimation error on given limited samples. To conduct layer-wise distillation, Bai *et al*. [19] add a 1×1 layer after each pruned layer block in the student and estimate the least-squared error to align the parameters with the student. A little differently, Li *et al*. [136] employ cross distillation losses to mimic the behavior of the teacher network, given its current estimations.

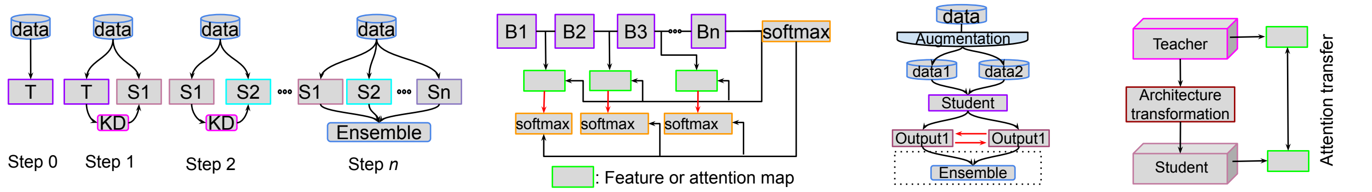
10.3 Challenges and potentials

Although KD methods with a small number of examples inspired by the techniques of data augmentation and layer-wise learning are convincing, these techniques are still confined by the structures of teacher networks since most methods rely on network pruning from teacher networks to create student networks. Besides, the performance of the student is heavily dependent on the amount of the crafted pseudo labels, which may impede the effectiveness of these methods. Lastly, most works focus on generic classification tasks, and it is unclear whether these methods are effective for the tasks without class labels (*e.g.*, low-level vision tasks).

11 SELF-DISTILLATION

Overal insight: *Is it possible to enable the student to distill knowledge by itself to achieve plausible performance?*

The conventional KD approaches [94], [114], [196], [229], [270] still have many setbacks to be tackled although significant performance boost has been achieved. First of all, these approaches are of low efficiency since student models scarcely exploit all knowledge from the teacher models. Second, designing and training high-capacity teacher models still face up with many obstacles. Third, two-stage distillation requires high computation and storage costs. To tackle these challenges, several novel self-distillation frameworks [47], [48], [63], [81], [102], [126], [155], [174], [254], [262], [281] have been proposed recently. The goal of self-distillation is to learn a student model by distilling knowledge in itself without referring to other models. We now provide a detailed analysis of the technical details for self-distillation.



(a) Born-again KD (b) KD via 'deep' supervision (c) data augmented KD (d) KD with architecture transform

Fig. 9. An illustration of self-distillation methods. (a) born-again distillation. Note that T and $S_1, \cdots, S_n$ can be multi-tasks. (b) distillation via 'deep' supervision where deepest branch (B_n) is used to distill knowledge to shallower branches. (c) distillation via data augmentation (*e.g.*, rotation, cropping). (d) distillation with network architecture transformation (*e.g.*, changing convolution filters).

TABLE 8
A taxonomy of self-distillation methods. Logits and hints indicate the knowledge to be distilled. 'Deep' supervision is for self-distillation from deepest branch of student. One-stage KD is checking whether self-distillation is achieved in one step. ✓/ ✗is for yes/no.

Method	Logits	Hints	Data augmentation	'Deep' supervision	One-stage KD	Multi-task KD	Architecture transformation
Clarm [47]	✓	✗	✗	✗	✗	✓	✗
Chowley [48]	✗	Attention map	✗	✗	✗	✗	✓
Furlanello [63]	✓	✗	✗	✗	✗	✗	✗
Hahn [81]	✓	✗	✗	✗	✗	✗	✗
Hou [102]	✗	Attention maps	✗	✓	✓	✗	✗
Luan [155]	✓	Feature maps	✗	✓	✓	✓	✗
Lee [126]	✓	✗	✓	✗	✗	✗	✗
Xu [254]	✓	Feature maps	✓	✗	✓	✗	✗
Zhang [281]	✓	Feature maps	✗	✓	✓	✗	✗
Yang [262]	✓	✗	✗	✗	✗	✗	✗

11.1 Born-again distillation

Insight: *Sequential teaching of students themselves enables them to become masters and outperform their teachers significantly.*

Furlanello *et al*. [63] in fact initialize the concept of self-distillation in which the students are parameterized identically to their teachers as shown in Fig. 9(a). Through sequential teaching, the student is consecutively updated, and at the end of the procedure, additional performance gains are achieved by an ensemble of multiple student generations. Hahn *et al*. [81] then apply born-again distillation [63] to natural language processing. Yang *et al*. [262] observe that it remains unclear how S-T optimization works, and they then focus on putting strictness (add an extra term to the standard cross-entropy loss) to the teacher model such that the student can better learn inter-class similarity and potentially prevent over-fitting. Instead of learning a single task, Clark *et al*. [47] extend [63] to multi-task setting where single-task models are distilled sequentially to teach a multi-task model. Since the born-again distillation approach is based on the multi-stage training, it is less efficient and computation-heavy compared to the following methods.

11.2 Distillation via 'deep' supervision

Insight: *The deeper layer (or branch) in the student model contains more useful information than those of shallower layers.*

Among the methods, Hou *et al*. [102], Luan *et al*. [155] and Zhang *et al*. [281] propose the similar approaches where the target network (student) is divided into several shallow sections (branches) according to its depth and original structure (see Fig. 9(b)). As the deepest section may contain more useful and discriminative feature information than those of shallower sections, the deeper branches can be used to distill knowledge to the shallower branches. A little differently, in Hou [102], instead of directly distilling features, attention-based methods used in [276] are adopted to force shallower layers to mimic the attention maps of deeper layers. Luan *et al*. [155] make each layer branch (ResNet block) as a classifier; thus, the deepest classifier is used to distill earlier classifiers' feature maps and logits.

11.3 Distillation based on data augmentation

Insight: *Data augmentation (e.g., rotation, flipping, cropping, etc) during training forces student network to be invariant to the augmentation transformations via self-distillation.*

Although most methods focus on how to better supervise student in self-distillation, data representations for training the student are not fully excavated and utilized. To this end, Xu *et al*. [254] and Lee *et al*. [126] focus on self-distillation via data augmentation of the training samples as shown in Fig. 9(c). There are some advantages to such a framework. First, it is efficient and effective to optimize a single student network without branching or the assistance of other models. Second, with data-to-data self-distillation, the student learns more inherent representations for generalization. Third, the performance of the student model is greatly enhanced with relatively low computation cost and memory load.

Xu *et al*. [254] apply random mirror and cropping to the batch images from the training data. Besides, inspired by mutual learning [283], the last feature layers and softmax outputs of the original batch image and distorted batch images are mutually distilled via MMD loss [103] and KL divergence loss, respectively. In contrast, Lee *et al*. [126] consider two types of data augmentation (rotation and color permutation to the same image), and the ensemble method used in [32], [123], [293] is employed to aggregate all logits of the student model to one, which is in turn used to transfer the knowledge to itself.

11.4 Distillation with architecture transformation

Insight: *A student model can be derived by changing convolution operators in the teacher model with any architecture change.*

Different from all the above-mentioned self-distillation methods, Crowley *et al*. [48] propose structure model distillation for memory reduction using a strategy of replacing standard convolution blocks with cheaper convolutions as shown in Fig. 9(d). In such a way, a student model is produced, which is a simple transformation of the teacher's architecture. Then attention transfer (AT) [103] is applied to align the teacher's attention map with that of the student's.

11.5 Summary and open challenges

Summary: In Table. 8, we summarize and compare different self-distillation approaches. Overall, using logits/feature information and two-stage training for self-distillation with 'deep' supervision from the deepest branch are the main stream. Besides, data augmentation and attention-based self-distillation approaches are promising. Lastly, it is shown multi-task learning with self-distillation is also a valuable direction deserving more research.

Open challenges: There still exist many challenges to tackle. First, there lacks theoretical support explaining why self-distillation works better. Mobahi *et al*. [174] provide theoretical analysis for born-again distillation [63] and find out that self-distillation may reduce over-fitting by loop-over training, thus leading to good performance. However, it is still unclear why other self-distillation methods (*e.g*., online 'deep' supervision [102], [155], [281]) even work better.

Besides, existing methods focus on self-distillation with a certain type of group-based network structures (*e.g*., ResNet group); thus, the generalization/flexibility of such self-distillation methods still need to be further probed. Lastly, all existing methods focus on classification-based tasks, and it is less clear whether self-distillation is effective for other tasks (*e.g*., low-level vision tasks).

12 CROSS-MODAL DISTILLATION

Overall insight: *KD for cross-modal learning is typically performed with network architectures containing modal-specific representations or shared layers, utilizing the training images in correspondence of different domains.*

One natural question we ask is whether it is possible to transfer knowledge from a pretrained teacher network for one task to a student learning another task while the training examples are in correspondence across domains. *Note that KD for cross-modal learning is essentially different from that for domain adaptation in which data are drawn independently from different domains, but the tasks are the same.*

Compared to previously-mentioned KD methods focused on transferring supervision within the same modality between teacher and student, cross-modal KD deals with the problem of using the teacher's representation as a supervision signal to train the student learning another task. In this problem setting, the student needs to rely on the visual input of the teacher to accomplish its task. Following this, many novel cross-modal KD methods [3], [8], [13], [16], [17], [53], [55], [78], [80], [96], [177], [178], [184], [200], [216], [223], [287] have been proposed. We now provide a systematic analysis for the technical details, meanwhile, point the challenges and potentials for cross-domain distillation.

12.1 Supervised cross-modal distillation

Using the ground truth labels for the data used in the student network is the common way of cross-modal KD, as shown in Fig. 10(a). Do *et al*. [53], Su *et al*. [216], Nagrani *et al*. [177], Nagrani *et al*. [178] and Hoffman *et al*. [96] rely on supervised learning for cross-modal transfer. Several works [3], [177], [178] leverage the synchronization of visual and audio information in the video data and learn a joint embedding between the two modalities. Afouras *et al*. [3] and Nagrani *et al*. [178] transfer the voice knowledge to learn a visual detector, while Nagrani *et al*. [177] utilize visual knowledge to learn a voice detector (student). Differently, Hoffman *et al*. [96], Do *et al*. [53] and Su *et al*. [216] focus on different modalities in only visual domain. In particular, Hoffman *et al*. [96] learn a depth network by transferring the knowledge from RGB network and fuse the information across modalities, which improves the object recognition performance at test time. Su *et al*. [216] utilize the knowledge from high-quality images to learn a classifier with better generalization on low-quality image (paired).

12.2 Unsupervised cross-modal distillation

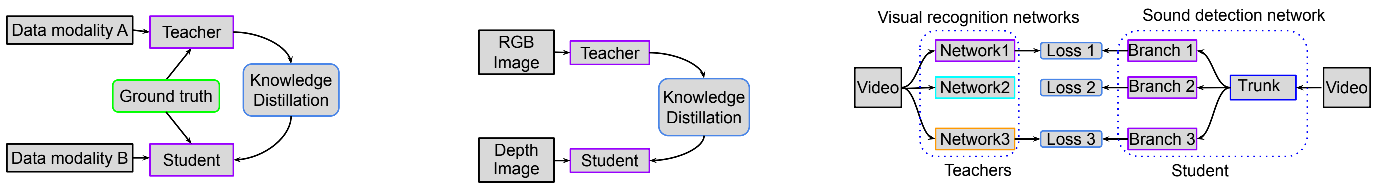
In contrast, most cross-modal KD methods exploit unsupervised learning due to the reason the labels in target domain are hard to get. Thus, these methods are also called distillation '*in the wild*'. In this setting, the knowledge from the teacher's modality provides *supervision* for the student network. [3], [8], [13], [16], [17], [55], [78], [80], [113], [184], [200], [223], [287] all aimed for cross-modal distillation in an unsupervised manner.

12.2.1 Learning from one teacher

Afouras *et al*. [3], Albanie *et al*. [8], Gupta *et al*. [78], Thoker *et al*. [223], Zhao *et al*. [287], Owens *et al*. [184], Kim *et al*. [113], Arandjelovic *et al*. [13], Gan *et al*. [67], Tang *et al*. [221] and Hafner *et al*. [80] focus on distilling knowledge from one teacher (see Fig. 10(b)), and mostly learn a single student network except Thoker *et al*. [223], Zhao *et al*. [287] learning two students. Specially, Thoker *et al*. [223] refer to mutual learning [283] where two students also learn from each other based on two KL divergence losses. Besides, Zhao *et al*. [287] exploit feature fusion strategy, similar to [112], [114] to learn a more robust decoder. Do *et al*. [53] focus on unpaired images of two modalities and learn a semantic segmentation network (student) using the knowledge from the other modality (teacher).

12.2.2 Learning from multiple teachers

Aytar *et al*. [16], Salem *et al*. [200], Aytar *et al*. [17] and Do *et al*. [53] exploit the potential of distilling from multiple teachers as mentioned in Sec. 7. Most methods rely on the concurrent knowledge among visual, sound, and textual information, as shown in Fig. 10(c). However, Salem *et al*.



(a) Supervised cross-modal KD

(b) Unsupervised cross-modal KD with one teacher

(c) cross-modal KD from multiple teachers

Fig. 10. Graphical illustration of cross-modal KD methods. (a) supervised cross-modal KD from teacher with one modality to the student with the other. (b) unsupervised cross-modal KD with one teacher. (c) unsupervised cross-modal KD with multiple teachers, each of which is transferring the discriminative knowledge to the student.

TABLE 9
A taxonomy of cross-modal knowledge distillation methods.

Method	Use GT	Source modality	Target modality	Number of teachers	Online KD	Knowledge	Model compression
Ayter [16]	✗	RGB frames	Sound	Two	✗	Logits	✗
Su [216]	✓	HR image map	LR image	One	✗	Soft labels	✓
Nagrani [177]	✓	RGB frames	Voice	One	✓	Soft labels	✗
Nagrani [178]	✓	Voice/face	Face/voice	Multiple	✓	Features	✗
Hoffman [96]	✓	RDG images	Depth images	One	✗	Features	✗
Afouras [3]	✗	Audio	Video	One	✗	Soft labels	✗
Albanie [8]	✗	Video frames	Sound	One	✗	Logits	✗
Gupta [78]	✗	RGN images	Depth images	One	✗	Soft labels	✗
Salem [200]	✗	Scene classification, object detection	Localization	Three	✗	Soft labels	✗
Thoker [223]	✗	RGB video	Skeleton data	One	✗	Logits	✗
Zhao [287]	✗	RGB frames	Heatmaps	One	✗	Confidence maps	✗
Owens *et al.* [184]	✗	Sound	Video frames	One	✗	Soft labels	✗
Arandjelovic [13]	✗	Video frames	Audio	One	✗	Features	✗
Do [53]	✓	Image, Questions, Answer info.	Image questions	Three	✗	Logits	✓
Aytar [17]	✓	Image	Sound, Image, Text	Three	✗	Features	✗
Kim [113]	✗	Sound/images	Images/sound	One	✗	Features	✗
Dou [55]	✗	CT images	MRI images	One	✓	Logits	✓
Hafner [80]	✗	Depth images	RGB images	One	✗	Embeddings	✗
Gan [67]	✓	Video frame	Sound	One	✗	Feature soft labels	✗
Tang [221]	✗	Texts	Video frame	One	✗	✗	

[200] focus on only visual modality, where teachers learn the information of object detection, image classification, and scene categorization via multi-task approach, and distill the knowledge to a single student.

12.3 Potentials and open challenges

Potentials: Based on the analysis of the existing cross-modal KD techniques in Table. 9, we can see that cross-modal KD expands the generalization capability of the knowledge learned from teacher models. The great potential of cross-domain KD is *relieving the dependence* for a large amount of labelled data in one modality or both. Besides, cross-domain KD is more *scalable* and can be *easily* applied to a *new* form of distillation task. Moreover, it is advantageous for learning multiple modalities of data 'in the wild' since it is relatively easy to get data with one modality based on other data. In vision fields, cross-modal KD provides the potential for distilling knowledge among images taken from different types of cameras. For instance, we can distill knowledge from an RGB image to event streams (stacked event images from event cameras) [216], [238].

Open challenges: Since the knowledge are the transferred representations (*e.g.*, logits, features) of teacher models, how to ensure the robustness of the transferred knowledge is crucial. We hope to transfer the good representations and however, negative representations do exist; thus, it is imperative that the supervision provided by the teachers is complementary to the target modality. Besides, existing cross-modal KD methods are highly dependent on data sources (*e.g.*, video, images), however, finding the data with paired (*e.g.*, RGB image with depth pair) or multiple modalities (class labels, bounding boxes and segmentation labels) is not always an easy task. Thus, is it possible to come up with a way for data-free distillation or distillation with a few examples? Or is it possible to just learn a student model with the data from the target modality based on the knowledge of the teacher without referencing the source modality?

On the other hand, the existing cross-modal KD methods are mostly offline methods, which are computation-heavy and memory-intensive; thus, it would be better if an online KD strategy is considered. Lastly, some works (*e.g.*, [17], [200]) learn a student model using the knowledge from multiple teachers; however, the student is less versatile or modality-dependent. Inspired by the analysis of Sec. 7.5, we open a research question, whether it is possible to learn a versatile student that can perform tasks from multiple modalities?

13 KD FOR SEMI-/SELF-SUPERVISED LEARNING

Overall insight: *KD with S-T learning is to learn a rich representation by training a model with a large number of unlabelled dataset and limited amount of labelled data.*

Semi-supervised learning usually handles the problem of over-fitting due to the lack of high-quality labels of training data. To this end, most methods apply S-T learning that assumes a dual role as a teacher and a student. The student model is to learn the given data as before, and the teacher learns from the noisy data and generates predicted targets, which are transferred to the student model via consistency cost. In self-supervised learning, the student itself generates knowledge to be learned via various approaches, and the knowledge is then transferred to itself via distillation losses. We now provide a detailed analysis of the technical details of the existing methods.

13.1 Semi-supervised learning

The baseline S-T frameworks for semi-supervised learning are initialized by Laine *et al*. [122] and Tarvainen *et al*. [222] as illustrated in Fig. 1(b). That is, the student and teacher models have the same structures, and the teacher learns from noise and transfers knowledge to the student via consistency cost. Interestingly, in [222], the teacher's weights are updated using the earth moving average (EMA) of student's weights. Inspired by [222], Luo *et al*. [158], Zhang *et al*. [282] French *et al*. [61] Choi *et al*. [43], Cai *et al*. [29] and Xu *et al*. [257] all employ the similar frameworks where the teacher's weights are updated using EMA of those of student. However, Ke *et al*. [112] mention that using a coupled EMA teacher is not sufficient for the student since the degree of coupling increases as the training goes on. To tackle this problem, the teacher is replaced with another student, and two students are optimized individually during training while a stabilization constraint is provided for knowledge exchange (similar to mutual learning [283]).

Instead of taking independent weights between teacher and student, Hailat *et al*. [83] employ weight-sharing, in which the last two fully connected layers of teacher and student are kept independent. The teacher model plays the role of teaching the student, stabilizing the overall model, and attempting to clean the noisy labels in the training dataset. In contrast, Gong *et al*. [73] and Xie *et al*. [252] follow the conventional distillation strategy proposed by [94], where a pretrained teacher is introduced to generate learnable knowledge using unlabelled data and utilizes it as privileged knowledge to teach the student on labelled data. However, during learning of the student, Xie *et al*. inject noise (*e.g.*, dropout) to the student such that it learns better than the teacher. Papernot *et al*. [186] propose to distill from multiple teachers (an ensemble of teachers) on a disjoint subset of sensitive data (augmented with noise) and to aggregate the knowledge of teachers to guide the student on query data.

13.2 Self-supervised learning

Distilling knowledge for self-supervised learning aims to preserve the learned representation for the student itself, as depicted in Fig. 1(c). Using pseudo labels is the most common approach, as done in [128], [181]. Specifically, Lee *et al*. [128] adopt self-supervised learning for KD, which not only ensures the transferred knowledge does not vanish but also provides an additional performance improvement. In contrast, Noroozi *et al*. [181] propose to transfer knowledge by reducing the learned representation (from a pretrained teacher model) to *pseudo-labels* (via clustering) on the unlabelled dataset, which are then utilized to learn a smaller student network. Another approach is based on data augmentation (*e.g.*, rotation, cropping, color permutation) [126], [254], which has been mentioned in Sec. 11.3.

13.3 Potentials and open challenges

Based on the technical analysis for the KD methods in semi-/self-supervised learning, it is noticeable that online distillation is the mainstream. However, there are several challenges. First, as pointed by [112], using EMA for updating teacher's weights might lead to less optimal learning of knowledge. Second, no methods attempt to exploit the rich feature knowledge from teacher models. Third, data augmentation methods in these distillation methods are less effective compared to those proposed in Sec. 11, in which the advantages of adversarial learning are distinctive. Fourth, the representations of knowledge in these methods are limited and less effective. It is the potential to exploit a better-structured data representation approach, such as GNNs. With these challenges, the future directions of KD for semi-/self-supervised learning could gain inspirations from exploiting feature knowledge and more sophisticated data augmentation methods together with more robust representation approaches.

14 KD WITH NOVEL LEARNING METRICS

14.1 Few-shot learning

Insight:*Is it possible possible to learn an effective student model to classify unseen classes (query set) by distilling knowledge from a teacher model with the support set?*

Different from the methods discussed in Sec. 10 focusing on distillation with a few samples for training a student network (without learning to generalize to new classes), this section stresses on analyzing the technical details of few-shot learning with KD. We first briefly introduce the definition of few-shot learning. Few-shot learning is to classify new data, having seen from only a few training examples. Few-shot learning itself is a meta-learning problem in which the DNN learns how to learn to classify given a set of training tasks and evaluate using a set of test tasks. Here, the goal is to discriminate between N classes with K examples of each (so-called N-way-K-shot classification). In this setting, these examples are known as the *support set*. In addition, there are further examples of the same classes, known as a *query set*. The approaches for learning prior knowledge of a few-shot are usually based on *three* types: prior knowledge about similarity, prior knowledge about learning procedure, and prior knowledge about data. We now analyze the KD methods for few-shot learning [56], [60], [107], [146], [189] that have been recently proposed.

Prior knowledge about similarity: Park *et al*. [189] propose distance-wise and angle-wise distillation losses. The aim is to penalize the structural differences in relations of the learned representations between teacher and student for few-shot learning.

Prior knowledge about learning procedure: [60], [107] tackle the second type of prior knowledge, namely learning procedure. To be specific, Flennerhag *et al*. [60] focus on transferring knowledge across the learning process, in which the information from previous tasks is distilled to facilitate the learning on new tasks. However, Jin *et al*. [107] address the problem of learning a meta-learner that can automatically learn what knowledge to transfer from the source network to where in the target network.

Prior knowledge about data: Dvornik *et al*. [56] and Liu *et al*. [146] address the third type of prior knowledge, namely data variance. In [56], The distillation approach is based on the ensemble of several teacher networks to leverage the variance of the classifier and a new strategy to encourage the networks to cooperate while encouraging diversity of predict on. However, in [146], the goal is to preserve the knowledge of the teacher (*e.g*., intra-class relationship) learned at the pretraining stage by generating pseudo labels for training samples in the fine-tuning set.

14.1.1 What's challenging?

The existing techniques based on our analysis actually expose crucial challenges. First, the overall performance of KD-based few-shot learning is convincing; however, the power of meta-learning is somehow degraded or exempted. Second, transferring knowledge from multi-source networks is potential, however identifying what to learn and where to transfer is heavily based on the meta-learner, and selecting which teacher to learn is computation-complex. Third, all approaches focus on a task-specific distillation; however, the performance drops as domain shifts. Thus, future works may focus more on handling these problems.

14.2 Incremental Learning

Overall insight: *KD for incremental learning mainly deals with two challenges: maintaining the performance on old classes and balancing between old and new classes.*

Incremental learning investigates how to learn the new knowledge continuously to update the model's knowledge while maintaining existing knowledge [249]. Many attempts [30], [102], [168], [211], [249], [278], [290] have been taken to utilize KD to address the challenge of maintaining the old knowledge. Based on the number of teacher networks used for distillation, these methods can be categorized into two types: distillation from a single teacher and distillation from multiple teachers.

14.2.1 Distillation from a single teacher

Shmelkov *et al*. [211], Wu *et al*. [249], Michieli *et al*. [168] and Hou *et al*. [102] focus on learning student networks for the new classes by distilling knowledge (logits information) from pretrained teachers on old-class data. Even though these methods vary in the form of tasks and distillation process, they follow similar S-T structure. That is, usually, the pretrained model is taken as the teacher, and the same network or different network is employed to adapt for new classes. A little differently, Michieli *et al*. exploit the intermediate feature representations and transfer them to the student.

14.2.2 Distillation from multiple teachers

Castro *et al*. [30], Zhou *et al*. [290] and Ammar *et al*. [10] concentrate on the learning an incremental model with multiple teachers. Specifically, Castro *et al*. share the same feature extractor between teachers and the student. The teachers contain old classes, and their logits are used for distillation and classification. Interestingly, Zhou *et al*. propose a multi-model and multi-level KD strategy in which all previous model snapshots are leveraged to learn the last model (student). This approach is similar to born-again KD methods, as mentioned in Sec. 11, where the student model at the last step, is updated using the assembled knowledge from all previous steps. However, the assembled knowledge here also depends on the intermediate feature representations. Ammar *et al*. develop a cross-domain incremental RL framework in which the transferable knowledge is shared and then projected to different task domains of the task-specific student peers.

14.2.3 Open challenges

The existing methods rely on multi-step training (offline); however, it is more significant if the online (one-step) distillation approaches can be excavated to improve the learning efficiency and performance. Besides, it is required to access the previous data in existing methods to avoid the ambiguities between the update steps. However, it remains a challenge on whether data-free distillation methods can be possible or not. Furthermore, existing methods only tackle the incremental learning of new classes in the same data domain; however, it is more significant if cross-domain distillation methods can be applied in this direction.

14.3 Reinforcement learning

Overall insight: *KD in reinforcement learning is to encourage polices (as students) in the ensemble to learn from the best policy (as a teacher), thus enabling the rapid improvement and continuous optimization for the optimal policy.*

Reinforcement learning (RL) is a learning problem that trains a policy to interact with the environment, such that the policy yields maximal reward. To use the best policy to guide other policy, KD has been employed in [15], [27], [60], [99], [138], [140], [199], [261]. Based on the specialties of these methods, we divide them into three categories and provide explicit analysis as follows. *Note that we assume one has some basic familiarity with RL and skip the definitions of Deep Q-network and A3C.*

14.3.1 Collaborative distillation

Xue *et al*. [261], Hong *et al*. [99] and Lin *et al*. [138] are all focused on collaborative distillation, which is similar to the way done in mutual learning [283]. In Xue *et al*., the agents teach each other based on the reinforcement rule, and teaching occurs between the value function of one agent (teacher) and that of another (student). Note that the knowledge is provided by a group of student peers periodically and assembled to enhance the learning speed and stability, which is similar to [99]. However, Hong *et al*. [99] periodically distill the best-performing policy to the rest of the ensemble. Lin *et al*. stress on collaborative learning among heterogeneous learning agents and incorporate the knowledge into online training.

14.3.2 Model compression with RL-based distillation

Ashok *et al*. [15] tackle the problem of model compression via RL. The method takes a larger teacher network and outputs a compressed student network, which is derived from the teacher network. In particular, two recurrent policy networks are employed to aggressively remove layers from the teacher network and to carefully reduce the size of each remaining layer. The learned student network is evaluated by a reward, which is a score based on the accuracy and compression of the teacher.

14.3.3 Random network distillation

Burda *et al*. [27] focus on a different perspective where the prediction problem is randomly generated. The approach involves two networks, the target (student) network, which is fixed and randomly initialized, and a predictor (teacher) network trained on the data collected by the agent. With the knowledge distilled from the predictor, the target network tends to have lower prediction errors. Rusu *et al*. [199] also apply random initialization for the target network; however, they focus more on online learning of action policies, which can be either single-task or muti-task.

14.3.4 Potentials of RL-based KD

We have analyzed existing RL-based KD methods in detail. Especially, we notice that model compression via RL-based KD is promising due to its extraordinary merits. First, RL-based KD better addresses the problem of *scalability* of network models, which is similar to what neural architecture search (NAS) does. Besides, the reward functions in RL-based KD *better balance* the accuracy-size trade-off. Moreover, with RL, it is also possible to transfer knowledge from a *smaller model to a larger model*, which shows distinctive advantages than other KD methods.

15 Applications for Visual Intelligence

15.1 Semantic and motion segmentation

Insight: *Semantic segmentation is a structured problem, thus distilling knowledge for semantic segmentation networks has to consider the structure information (e.g., spatial context structures).*

Semantic segmentation is a special classification problem of predicting the category label in a pixel-wise manner. Since the existing SOTA methods such as FCNs [152] are with large model size and with high computation cost, some methods [38], [55], [57], [88], [149], [168], [176], [206], [251] have been proposed to train lightweight networks via KD. Although these methods vary in learning methods, most of them share the same distillation frameworks. Specially, Xie *et al*. [251], Shan [206] and Michieli *et al*. [168] are all built upon pixel-wise, feature-based distillation methods. Besides, Liu *et al*. [149] and He *et al*. [88] both exploit affinity-based distillation strategy using intermediate features; however, Liu *et al*.also employ pixel-wise and holistic KD losses via adversarial learning. In contrast, Dou *et al*. [55] focus on unpaired multi-modal segmentation and propose an online KD method via mutual learning [283]. Chen *et al*. [38] propose a target-guided KD approach to learn the real image style by training the student to imitate a teacher trained with real images. Mullapudi *et al*. [176] train a compact video segmentation model via online distillation in which a teacher's output is used as a learning target to adapt the student and to select the next frame for supervision.

15.2 KD for visual detection and tracking

Insight: *Distilling visual detectors has to consider challenges of regression, region proposals and less voluminous labels.*

Visual detection is a very crucial high-level task in computer vision. Speed and accuracy are the two key factors for visual detectors. To achieve impressive speed-up and lightweight network models, KD is of potential choice. However, applying distillation methods to detection is more challenging than those designed for classification. First, the performance of detection degrades seriously after compression. Second, detection classes are not equally important; thus special considerations for distillation have to be taken into account. Third, domain and data generalization has to be considered for a distilled detector. To tackle these challenges, many impressive KD methods [33], [36], [37], [59], [71], [86], [98], [107], [109], [119], [131], [156], [201], [221], [240], [253] have been proposed for compressing visual detection networks. We categorize these methods according to their specialties (*e.g.*, pedestrian detection).

15.2.1 Generic object detection

[33], [36], [58], [86], [98], [109], [150], [221], [240] all aim to learn lightweight object detectors with KD. Among these works, Chen *et al*. [36], Hao *et al*. [86] highlight learning a class-incremental student detector by following the generic KD framework, namely from a pretrained teacher, however, novel object detection losses are adopted as strong impetus for learning new classes. These losses handle classification results, location results, the detected region of interest and all intermediate region proposals. On the other hand, Chen *et al*. [33] learns a student detector by distilling knowledge from the intermediate layer, logits, and regressor of the teacher, in contrast to [240] in which only the intermediate layer of the teacher is utilized based on fine-grained imitation mask to identify informative locations. Jin *et al*. [109], Tang *et al*. [221] and Hong *et al*. [98] all exploit multiple intermediate layers as useful knowledge. Jin *et al*.design

an uncertainty-aware distillation loss to learn the multiple-shot features from the teacher network. However, Hong *et al.*and Tang *et al.* are based on one-stage KD (online) via adversarial learning and semi-supervised learning, respectively. Differently, Liu *et al.* [150] combine single S-T learning and mutual learning of students for learning lightweight tracking network.

15.2.2 Pedestrian detection

While person detection is based on generic object detection, it is more challenged by various sizes and aspect ratios of pedestrians under extreme illumination conditions. To learn effective lightweight detector, Chen *et al.* [37] suggest using the unified hierarchical knowledge via multiple intermediate supervisions, in which not only the feature pyramid (from low-level to high-level features) and region features but also the logits information are distilled. Kruthiventi *et al.* [119] learns an effective student detector in challenging illumination condition by extracting dark knowledge (both RGB and thermal-like hint features) from a multi-modal teacher network.

15.2.3 Face detection

Ge *et al.* [71] and Karlekar *et al.* [111] both compress face detectors to recognize low-resolution faces via selective KD (last hidden layer) from teachers which are initialized to recognize high-resolution faces. In contrast, Jin *et al.* [107], Luo *et al.* [156] and Feng *et al.* [59] only use single type of images. Jin *et al.*focus on compressing face detectors by fully using the supervisory signal from classification maps of teacher models and regression maps of the ground truth. They point out an important aspect that the classification map of a larger model is worth learning than that of smaller models. Feng *et al.*present a triplet KD method to transfer knowledge from a teacher model to a student model in which a triplet of samples, the anchor image, the positive image, and the negative image, is used. The triplet loss aims to minimize the feature similarity between the anchor and positive images while maximizing that between the anchor and negative images. Luo *et al.*address the importance of neurons at the higher hidden layer of the teacher, and a neuron selection method is applied to choose neurons that are crucial for teaching the student. Dong *et al.* [54], on the other hand, concentrate on the interaction between the teacher and the students. That is, two students learn to generate pseudo facial landmark labels, which are filtered and selected as the qualified knowledge by the teacher.

15.2.4 Vehicle detection and driving learning

Lee *et al.* [131], Saputra *et al.* [201] and Xu *et al.* [259] focus more on detection tasks for autonomous driving. In particular, Lee *et al.*focus on compressing a vehicle maker classification system based on a cascaded CNNs (teacher) into a single CNN structure (student). The proposed distillation method uses the feature map as the transfer medium, and the teacher and student are trained in parallel (online distillation). Although the detection task is different, Xu *et al.* build a binary weight Yolo vehicle detector by also mincing the feature maps of the teacher network from easy tasks to difficult ones progressively. On the other hand, Zhao *et al.* [286] exploit a S-T framework to encourage the student to learn the teacher's sufficient, invariant representation knowledge (based on semantic segmentation) for driving.

15.2.5 Pose detection

Distilling human pose detectors has several challenges. First, the lightweight detectors have to deal with arbitrary person images/videos to determine joint locations with unconstrained human appearances. Second, the detectors must be robust to viewing conditions and background noises. Third, the detectors should have fast inference speed and be memory-efficient. To this end, [104], [164], [180], [223], [233], [255], [280] have formulated various distillation methods. Zhang *et al.* [280] achieve effective knowledge transfer by distilling the joint confidence maps from a pre-trained teacher model while Huang *et al.* [104] exploit the heat map and location map of a pretrained teacher as the knowledge to be distilled. Besides, Xu *et al.* [255], Thoker *et al.* [223] and Martinez *et al.* [164] all focus on multi-person pose estimation, however, Thoker *et al.*address cross-modality distillation problem in which a novel framework based on mutual learning [283] of two students supervised one teacher is initialized. Xu *et al.* [255] learn the integral knowledge, namely the feature, logits, and structured information via a discriminator, under standard S-T framework while Martinez *et al.* [164] train the student to mimic the confidence maps, feature maps and inner-stage predictions of a pre-trained teacher with depth images. Wang *et al.* [233] train a 3D pose estimation network by distilling knowledge from non-rigid structure from motion using only 2D landmark annotations. In contrast, Nie *et al.* [180] introduce online KD in which the pose kernels in videos are distilled by leveraging the temporal cues from the previous frame in a one-shot learning manner.

15.3 Domain adaptation

Insight: *Is it possible to distill knowledge of a teacher in one domain to a student another domain?*

Domain adaptation (DA) address the problem of learning a target domain with the help of a different but related source domain [12]. Since Lopez *et al.* [154] and Gupta *et al.* [78] initially propose the technique of transferring knowledge between images from different modalities (called generalized distillation), it is natural to ask the question: can this novel technique be used to address the problem of DA? The challenge of DA usually comes with transferring knowledge from the source model (usually with labels) to the target domain with unlabelled data. To address the issue, recently many KD methods based on S-T frameworks [12], [29], [39], [43], [50], [97], [166], [228], [257] have been proposed. Although these methods are focused on diverse tasks, technically, they can be categorized into two types: unsupervised and semi-supervised DA via KD.

15.3.1 Semi-supervised DA

French *et al.* [61] Choi *et al.* [43], Cai *et al.* [29] and Xu *et al.* [257], Cho *et al.* [40] all propose similar S-T frameworks for either semantic segmentation or object detection. Specially, these frameworks are the updated methods of Mean-Teacher [222], which is based on self-ensemble of the student networks (teacher models are with the same structure as the students). Note that the weights of the teacher models in these methods are the exponential moving average (EMA) of the weights of student models. A little differently, Choi *et al.*add a target-guided generator to produce augmented images, instead of stochastic augmentation as in [29], [61], [257]. Cai *et al.*also exploit the feature knowledge from the teacher model and apply region-level and intra-graph consistency losses instead of mean square error loss.

In contrast, Ao *et al.* [12] propose a generalized distillation DA method by applying the generalized distillation information [154] to multiple teachers to generate soft labels which are then used to supervise the student model (this framework is similar to online KD from multiple teachers as mentioned in Sec. 7). Cho *et al.* [40] propose a S-T learning framework in which a smaller depth prediction network is trained based on the supervision of the auxiliary information (ensemble of multiple depth predictions) obtained from a larger stereo matching network (teacher).

15.3.2 Unsupervised DA

Some methods such as [39], [97] distill the knowledge from source domain to target domain based on adversarial learning [74] and image translation [106], [237], [239]. Technically, images in the source domain are translated to images in the target domain as data augmentation, and cross-domain consistency losses are adopted to force the teacher and student models to produce consistent predictions. Differently, Tsai *et al.* [228] and Deng *et al.* [50] focus on aligning the feature similarities between teacher and student models, compared to Meng *et al.* [166] focusing on aligning softmax outputs.

15.4 Depth and scene flow estimation

Insight: *The challenges for distilling depth and flow estimation tasks come with transferring the knowledge of data and labels.*

Depth and optical flow estimations are low-level vision tasks aiming to estimate the 3D structure and motion of the scene. There are several challenges. First, different from some tasks (*e.g.*, semantic segmentation), depth and flow estimations do not have don't have class labels, thus directly applying existing KD techniques may not work well. Besides, learning a lightweight student model usually requires a large amount of labelled data to achieve robust generalization capability; however, acquiring these data is very costly.

With these challenges, Guo *et al.* [77], Pilzer *et al.* [193] and Tosi *et al.* [225] all propose distillation-based approaches to learn monocular depth estimation. These methods are focused on handling the second challenge, namely, data distillation. Specifically, Pilzer *et al.* [193] propose an unsupervised distillation approach where the left image is translated to right via image translation framework [106], [237], and the inconsistencies between left and right images are used to improve depth estimation, which is finally used to improve the student network via KD. In contrast, Guo *et al.*and Tosi *et al.*focus on cross-domain KD, which aims to distill the *proxy* labels obtained from the stereo network (teacher) to learn a student depth estimation network. Choi *et al.* [40] learns a student network for monocular depth inference by distilling the knowledge of depth predictions from a *stereo* teacher network via data ensemble strategy.

On the other hand, Liu *et al.* [145] and Aleotti *et al.* [9] propose data-distillation methods for scene flow estimation. Liu *et al.* distill reliable predictions from a teacher network with unlabelled data and use these predictions (for non-occluded pixels) as annotations to guide a student network to learn the optical flow. However, Liu *et al.*propose to leverage on the knowledge learned by the teacher networks specialized in stereo to distill proxy annotations, which is similar to the KD method for depth estimation in [77], [225]. Strikingly, Tosi *et al.* [226] learn a compact network for predicting holistic scene understanding tasks jointly, including depth, optical flow and motion segmentation, based on distillation of proxy semantic labels and semantic-aware self-distillation of optical information.

15.5 Image translation

Insight: *Distilling GAN frameworks for image translation has to consider three factors: large number of parameters of the generators, no ground truth labels for training data and complex framework (both generator and discriminator).*

Several works also attempt to compress GANs for image translation with KD. Aguinaldo *et al.* [5] focus on unconditional GANs and first propose to learn a smaller student generator by distilling knowledge from the generated images of a larger teacher generator using mean squared error (MSE), however, the knowledge incorporated in the teacher discriminator is not excavated. In contrast, Chen *et al.* [34] and Li *et al.* [135] focus on conditional GANs and also exploit the knowledge from the teacher discriminator. Specifically, Chen *et al.*include a student discriminator to measure the distances between real images and images generated by student and teacher generators, and the student GAN is trained under the supervision of the teacher GAN. In particular, Li *et al.* [135] adopt the same discriminator of the teacher as the student discriminator and fine-tune the discriminator together with the compressed generator, which is automatically found with significantly fewer computation costs and parameters by using neural architecture search (NAS). Differently, Wang *et al.* [235] focus on compressing encoder-decoder based neural style transfer network via collaborative distillation (between the encoder and its decoder), where the student is restricted to learn the linear embedding of the teacher's output.

15.6 KD for Video understanding

15.6.1 Video classification and recognition

Bhardwaj *et al.* [25] and Wang *et al.* [243] employ the general S-T learning framework for video classification; however, the student is trained with only processing a few frames of the video and can produce a representation similar to that of the teacher. Gan *et al.* [66] focus on video concept learning for action recognition and event detection by jointly using web videos and images. That is, the learned knowledge from teacher network (so-called Lead network) by utilizing web videos is used to filter out the noisy images that are then used to fine-tune the teacher network to obtain a student network (so-called Exceeding network). Gan *et al.* [65], on the other hand, explore geometry, as a new type of practical auxiliary knowledge, for self-supervised learning of video representations.

15.6.2 Video captioning

[185], [285] both apply graph-based S-T learning for image captioning. Specifically, Zhang *et al*. [285] leverage the object-level information (teacher) to learn the scene feature representation (student) via a spatio-temporal graph. However, Pan *et al*. [185] highlight the importance of the relational graph connecting for all the objects in the video and force the caption model to learn the abundant linguistic language via teacher-recommended learning.

16 Discussions

In this section, we take into account some fundamental questions and challenges that are crucial for better understanding and improvement of KD.

16.1 Bigger models are better teachers?

The early assumption and idea behind KD are that soft labels (probabilities) from a trained teacher reflect more about the distribution of data than the ground truth labels [94]. If this is true, then it is expected that, as the teacher becomes more robust, the knowledge (soft labels) provided by the teacher would be more reliable and better captures the distribution of classes. That is, a more robust teacher provides more constructive knowledge as supervision to the student. Thus, the intuitive approach for learning more accurate student is to employ a bigger and more robust teacher. However, based on the experimental results in [41], it is found out that a bigger and more robust model does not always make a better teacher. As the teacher's capacity continuously grows up, the student's accuracy rises to some extent and then begins to drop. As there lack theoretical support for KD, we summarize two crucial reasons based on the studies in [41], [192].

- The student is able to follow the teacher; however, it can not absorb more useful knowledge from the teacher. This indicates that there is a mismatch between the KD losses and evaluation methods of accuracy. As pointed in [192], the optimization method used could have a large impact on the distillation risk. Thus, optimization methods might be crucial for significant KD to the student.
- Another reason comes with the situation that the student is unable to follow the teacher due to the large model capacity between the teacher and the student. This is stated in [91], [94] that S-T similarity is highly related to how well the student can mimic the teacher. If the student is more similar to the teacher, the student will produce outputs similar to the teacher.

On the other hand, the intermediate feature representations are also effective knowledge that can be used to learn the student [115], [196]. The common approach for feature-based distillation is to transfer the features into a type of representation such that the student can easily learn. In such a case, bigger models are better teachers? As pointed in [196], feature-based distillation is better than the distillation of soft labels and deeper student performs better than shallower one. Besides, increasing the number of layers (features representations), the performance of the student also increases [115]. However, when the student is fixed, a bigger teacher does not always teach the better student. When the similarity between the teacher and student is relatively high, the student tends to achieve plausible results.

16.2 Is pretrained teacher important?

While most works focus on learning a smaller student based on the pretrained teacher, the distillation is not always efficient and effective. When the model capacity between the teacher and the student is large, it is hard for the student to follow the teacher, thus inducing the difficulty of optimization. Is pretrained teacher really important for learning a compact student with plausible performance? [123], [283] propose to learn from student peers, each of which has the same model complicity. The greatest advantage of this distillation approach is efficiency since the pretraining of a high capacity teacher is exempted. Besides, instead of teaching, the student peers learn to cooperate with each other to obtain an optimal learning solution. Surprisingly, learning without the teacher even enables improving the performance. Why learning without the teacher is even better? This question has been studied in [222], which turns out that the compact student may have less chance of overfitting. Besides, [192] suggests that early-stopping of training on ImageNet [49] achiever better performance. Moreover, the ensemble of students pools their collective predictions together, thus helping to converge at more robust minima as pointed in [283].

16.3 Is born-again self-distillation better?

Born-again network [63] is the initial self-distillation method, in which the student is trained sequentially, and the later step is supervised by the earlier generation. At the end of the procedure, all the multiple student generations are assembled together to get additional gain. So is self-distillation in *generations* better? [192] finds that network architecture heavily determines the success of KD in generations. Besides, although the ensemble of the student models from the entire generations outperforms a single model trained from scratch, the ensemble does not outperform an ensemble of an equal number of models trained from scratch.

Instead, recent works [174], [254], [281] shift the focus from sequential self-distillation (multiple stages) to one-stage (online) manner. Namely, the student distills knowledge to itself without resort to the teacher and heavy computation. These methods show more efficiency, less computation, and higher accuracy. What's the reason why they are better? As pointed in [174], [281], the reason is that online self-distillation can help student models converge to flat minima. Second, self-distillation prevents student models from vanishing gradient problems. Lastly, self-distillation helps to extract more discriminative features. In summary, online self-distillation shows significant advantages than sequential distillation methods and would be more generalizable.

16.4 Single teacher vs multiple teachers

It is noticeable that recent distillation methods turn to exploit the potential of learning from multiple teachers. Is learning from multiple teachers really better than learning from a single teacher? To answer this question, [274] intuitively points out that the student can fuse different predictions from multiple teachers to establish its own comprehensive understanding of the knowledge. The intuition behind this is that by unifying the knowledge from the ensemble of teachers, the relative similarity relationship among teachers is maintained, thus providing more integrated dark knowledge for the student. Besides, similar to mutual learning [123], [283], the ensemble of teachers collect the individual predictions (knowledge) together, thus converging at more robust minima. Lastly, learning from multiple teachers relieves training difficulty, especially vanishing gradient problems.

16.5 Is data-free distillation effective enough?

Another discussion point comes to the situation when training data is unavailable. While some novel methods [35], [153], [267], [273] have been proposed to tackle this problem and achieve plausible results, there lacks a theoretical explanation for why such methods are robust enough for learning a portable student. Besides, these methods are only focused on classification; the generalization capability of such methods is still low. Lastly, most works employ generators to generate the 'latent' images from noise via adversarial learning [75], [237], however such methods are relatively hard to train and computationally expensive.

16.6 Logits vs features

The general knowledge defined in existing KD methods is from three aspects: logits, feature maps (intermediate layers), and both. However, it is still controversial concerning which one presents better knowledge. While works such as [91], [103], [115], [196], [229] focus on the better interpretation of feature representations and claim features might contain richer information, some other works [94], [179], [246], [283] mention that softened labels (logits) could represent each sample by class distribution and student can easily learn the intra-class variation. However, it is noticeable that KD via logits has obvious drawbacks. First, its effectiveness only limits to softmax loss function and relies on the number of classes (can not be applied to low-level vision tasks). Second, when the capacity between teacher and student is big, it is hard for the student to follow the teacher's class probabilities [41]. Besides, as studied in [229], semantically similar inputs tend to elicit similar activation patterns in teacher networks, indicating the similarity-preserving knowledge from intermediate features express not only the representation space but also the activations of object category (similar to class distributions). Thus, we can clearly see that features provide more affluent knowledge than logits and generalize better to the problems without class labels.

17 NEW OUTLOOKS AND PERSPECTIVES

In this section, we provide some ideas and discuss future directions of knowledge distillation. We take the latest deep learning methods (*e.g.*, neural architecture search (NAS), graph neural network (GNN)), novel non-euclidean distances (*e.g.*, hypersphere), better feature representation approaches, and potential vision applications, such as 360° vision [130] and event-based vision [238], etc, into account.

17.1 Potential of NAS

In recent years, NAS has become a popular topic in deep learning. NAS is the potential for automating the design of neural networks. The potential of NAS is recently demonstrated in [21], [135] for GAN compression and is shown to be effective for finding efficient student architecture from the teacher model with fewer computation costs and parameters. It turns out that NAS improves the compression ratio and accelerate the distillation process. A similar approach is taken by [15] who leans to remove layers of teacher network based on reinforcement learning (RL). Thus, we prospect that NAS with RL can be the good direction of knowledge distillation for model compression. This might significantly relieve the complexity and enhance the learning efficiency of existing methods in which the student is manually designed based on the teacher.

17.2 Potential of GNN

Although GNN has brought progress for the learning of KD under the S-T frameworks, there still remain some challenges. This is because most methods rely on finding structured data such that graph-based algorithms can be applied. Although [140] consider the instance features and instance relationships as instance graph, and [160] build input graph representation for multi-task knowledge distillation. However, in knowledge distillation, there exists *non-structural* knowledge in addition to the structural knowledge (*e.g.*, training data, logits, intermediate features, and outputs of teacher), thus it is potential to construct a flexible knowledge graph to tackle the *non-structural* distillation process.

17.3 Non-euclidean distillation measure

To date, the existing KD losses are mostly dependent on the euclidean loss (*e.g.*, l_1). However, such methods have their own limitations. [51] has shown that algorithms that regularize with euclidean distance, *e.g.*, MSE loss, are easily confused by random features. The difficulty happens when the model capacity between the teacher and the student is large. Besides, l_2 regularization does not punish small weight enough. Inspired by recent work [188] for GAN training, we conjecture that is the potential to exploit the information of higher-order statistics of data in non-euclidean space (*e.g.*, hypersphere). This is because geometric constraints induced by the non-euclidean distance might make training more stable, thus improving the efficiency of KD.

17.4 Better feature representations

Existing methods that focus on KD with multiple teachers show more potential for handling cross-domain problems or other problems where ground truth is not available. However, the ensemble of feature representations [45], [101], [187] is still challenging in some aspects. One critical challenge is how to fuse the feature representations and balance each with more robust gating mechanisms. Manually assigning weight to each component may hurt the diversity and flexibility of individual feature representation, thus impairing the effectiveness of ensemble knowledge. One possible solution is attention gates, as demonstrated in some detection tasks [134], [205]. which aims to highlight the important feature dimensions and prune feature responses to preserve the only the activations relevant to the specific task. Another one is inspired by the gating mechanism used in LSTM [95], [288]. Different from LSTM, gate unit in KD is elaborately designed to remember to features across different image regions, and to control the pass of each region feature as a whole by their contribution to the task (*e.g.*, classification) with the weight of importance.

17.5 More constructive theoretical analysis

While KD shows impressive performance improvements in many tasks, the intuition behind it is still less clear. Recently, [41] explains conventional KD [94] using linear models, and [7], [90], [224] focus on explaining feature-based KD. Mobahi *et al*. [174] provides theoretical analysis for self-distillation. However, concerning data-free KD and KD from multiple teachers, the mystery behind them is still unknown. Therefore, further theoretical studies on explaining the principle of these methods should be followed.

17.6 Potentials for special vision problems

While existing KD techniques are mostly developed based on vision problems (*e.g.*, classification), however, they are rarely exploited in some special vision fields, such as 360° vision [130] and event-based vision [238], [239]. The biggest challenge for both vision fields is the lack of labelled data, and learning in these vision needs a special change of inputs for neural networks. Thus, the potential of KD, especially cross-modal KD, for these two fields is promising. By distilling knowledge from the teacher trained with RGB images or frames to the student network specialized in learning to predict 360° images or stacked event images, it not only handles the problem of lack of data but also achieves desirable results in the prediction tasks.

18 Conclusion

This review of KD and S-T learning has covered major technical details and applications for visual intelligence. We provide a formal definition of the problem and introduce the taxonomy methods for existing KD approaches. Drawing connections among these approaches provide a new active area of research and is likely to create new methods that take advantage of the strength of each paradigm. Each taxonomy of the KD methods shows the current technical status regarding its advantages and disadvantages. Based on the explicit analysis, we then discuss how to overcome the challenges and break the bottleneck by exploiting new deep learning methods, new KD losses, and new vision application fields.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (NRF-2018R1A2B3008640) and the Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT (NRF2017M3C4A7069369).

References

[1] https://github.com/FLHonker/Awesome-Knowledge-Distillation.
[2] https://github.com/dkozlov/awesome-knowledge-distillation.
[3] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Asr is all you need: cross-modal distillation for lip reading. *arXiv preprint arXiv:1911.12747*, 2019.
[4] Gustavo Aguilar, Yuan Ling, Yu Zhang, Benjamin Yao, Xing Fan, and Edward Guo. Knowledge distillation from internal representations. *arXiv preprint arXiv:1910.03723*, 2019.
[5] Angeline Aguinaldo, Ping-Yeh Chiang, Alex Gain, Ameya Patil, Kolten Pearson, and Soheil Feizi. Compressing gans using knowledge distillation. *arXiv preprint arXiv:1902.00159*, 2019.
[6] Karim Ahmed, Mohammad Haris Baig, and Lorenzo Torresani. Network of experts for large-scale image categorization. In *European Conference on Computer Vision*, pages 516–532. Springer, 2016.
[7] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D. Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[8] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Emotion recognition in speech using cross-modal transfer in the wild. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 292–301, 2018.
[9] Filippo Aleotti, Matteo Poggi, Fabio Tosi, and Stefano Mattoccia. Learning end-to-end scene flow by distilling single tasks knowledge. *arXiv preprint arXiv:1911.10090*, 2019.
[10] Haitham Bou Ammar, Eric Eaton, José Marcio Luna, and Paul Ruvolo. Autonomous cross-domain knowledge transfer in lifelong policy gradient reinforcement learning. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.
[11] Rohan Anil, Gabriel Pereyra, Alexandre Passos, Robert Ormandi, George E Dahl, and Geoffrey E Hinton. Large scale distributed neural network training through online distillation. *arXiv preprint arXiv:1804.03235*, 2018.
[12] Shuang Ao, Xiang Li, and Charles X Ling. Fast generalized distillation for semi-supervised domain adaptation. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
[13] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 609–617, 2017.
[14] Siddhartha Arora, Mitesh M Khapra, and Harish G Ramaswamy. On knowledge distillation from complex networks for response prediction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3813–3822, 2019.
[15] Anubhav Ashok, Nicholas Rhinehart, Fares Beainy, and Kris M Kitani. N2n learning: Network to network compression via policy gradient reinforcement learning. *arXiv preprint arXiv:1709.06030*, 2017.
[16] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in neural information processing systems*, pages 892–900, 2016.
[17] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. See, hear, and read: Deep aligned representations. *arXiv preprint arXiv:1706.00932*, 2017.
[18] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662, 2014.
[19] Haoli Bai, Jiaxiang Wu, Irwin King, and Michael Lyu. Few shot network compression via cross distillation. *arXiv preprint arXiv:1911.09450*, 2019.

[20] David Barber and Felix V Agakov. The im algorithm: a variational approach to information maximization. In *Advances in neural information processing systems*, page None, 2003.

[21] Pouya Bashivan, Mark Tensen, and James J DiCarlo. Teacher guided architecture search. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5320–5329, 2019.

[22] Vasileios Belagiannis, Azade Farshad, and Fabio Galasso. Adversarial network compression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[23] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[24] Kartikeya Bhardwaj, Naveen Suda, and Radu Marculescu. Dream distillation: A data-independent model compression framework. *arXiv preprint arXiv:1905.07072*, 2019.

[25] Shweta Bhardwaj, Mukundhan Srinivasan, and Mitesh M Khapra. Efficient video classification using fewer frames. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 354–363, 2019.

[26] Cristian Bucilua, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541. ACM, 2006.

[27] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

[28] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1616–1637, 2018.

[29] Qi Cai, Yingwei Pan, Chong-Wah Ngo, Xinmei Tian, Lingyu Duan, and Ting Yao. Exploring object relation in mean teacher for cross-domain detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11457–11466, 2019.

[30] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 233–248, 2018.

[31] Shu Changyong, Li Peng, Xie Yuan, Qu Yanyun, Dai Longquan, and Ma Lizhuang. Knowledge squeezed adversarial network compression. *arXiv preprint arXiv:1904.05100*, 2019.

[32] Defang Chen, Jian-Ping Mei, Can Wang, Yan Feng, and Chun Chen. Online knowledge distillation with diverse peers, 2019.

[33] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017.

[34] Hanting Chen, Yunhe Wang, Han Shu, Changyuan Wen, Chunjing Xu, Boxin Shi, Chao Xu, and Chang Xu. Distilling portable generative adversarial networks for image translation. *arXiv preprint arXiv:2003.03519*, 2020.

[35] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3514–3522, 2019.

[36] Li Chen, Chunyan Yu, and Lvcai Chen. A new knowledge distillation for incremental object detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2019.

[37] Rui Chen, Haizhou Ai, Chong Shang, Long Chen, and Zijie Zhuang. Learning lightweight pedestrian detector with hierarchical knowledge distillation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1645–1649. IEEE, 2019.

[38] Yuhua Chen, Wen Li, and Luc Van Gool. Road: Reality oriented adaptation for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7892–7901, 2018.

[39] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1791–1800, 2019.

[40] Jaehoon Cho, Dongbo Min, Youngjung Kim, and Kwanghoon Sohn. A large rgb-d dataset for semi-supervised monocular depth estimation. *arXiv preprint arXiv:1904.10230*, 2019.

[41] Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4794–4802, 2019.

[42] Jungchan Cho and Minsik Lee. Building a compact convolutional neural network for embedded intelligent sensor systems using group sparsity and knowledge distillation. *Sensors*, 19(19):4307, 2019.

[43] Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6830–6840, 2019.

[44] LI Chongxuan, Taufik Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. In *Advances in neural information processing systems*, pages 4088–4098, 2017.

[45] Inseop Chung, SeongUk Park, Jangho Kim, and Nojun Kwak. Feature-map-level online adversarial knowledge distillation, 2020.

[46] Anthony Cioppa, Adrien Deliege, Maxime Istasse, Christophe De Vleeschouwer, and Marc Van Droogenbroeck. Arthus: Adaptive real-time human segmentation in sports through online distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[47] Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D Manning, and Quoc V Le. Bam! born-again multi-task networks for natural language understanding. *arXiv preprint arXiv:1907.04829*, 2019.

[48] Elliot J Crowley, Gavin Gray, and Amos J Storkey. Moonshine: Distilling with cheap convolutions. In *Advances in Neural Information Processing Systems*, pages 2888–2898, 2018.

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

[50] Zhijie Deng, Yucen Luo, and Jun Zhu. Cluster alignment with a teacher for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9944–9953, 2019.

[51] Michal Derezinski and Manfred KK Warmuth. The limits of squared euclidean distance regularization. In *Advances in Neural Information Processing Systems*, pages 2807–2815, 2014.

[52] Qianggang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Shu-Tao Xia. Adaptive regularization of labels, 2019.

[53] Tuong Do, Thanh-Toan Do, Huy Tran, Erman Tjiputra, and Quang D Tran. Compact trilinear interaction for visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 392–401, 2019.

[54] Xuanyi Dong and Yi Yang. Teacher supervises students how to learn from partially labeled images for facial landmark detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 783–792, 2019.

[55] Qi Dou, Quande Liu, Pheng Ann Heng, and Ben Glocker. Unpaired multi-modal segmentation via knowledge distillation. *arXiv preprint arXiv:2001.03111*, 2020.

[56] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3723–3731, 2019.

[57] Gongfan Fang, Jie Song, Chengchao Shen, Xinchao Wang, Da Chen, and Mingli Song. Data-free adversarial distillation. *arXiv preprint arXiv:1912.11006*, 2019.

[58] Heitor Felix, Walber M Rodrigues, David Macêdo, Francisco Simões, Adriano LI Oliveira, Veronica Teichrieb, and Cleber Zanchettin. Squeezed deep 6dof object detection using knowledge distillation. *arXiv preprint arXiv:2003.13586*, 2020.

[59] Yushu Feng, Huan Wang, Roland Hu, and Daniel T Yi. Triplet distillation for deep face recognition. *arXiv preprint arXiv:1905.04457*, 2019.

[60] Sebastian Flennerhag, Pablo G Moreno, Neil D Lawrence, and Andreas Damianou. Transferring knowledge across learning processes. *arXiv preprint arXiv:1812.01054*, 2018.

[61] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*, 2017.

[62] Takashi Fukuda, Masayuki Suzuki, Gakuto Kurata, Samuel Thomas, Jia Cui, and Bhuvana Ramabhadran. Efficient knowledge distillation from an ensemble of teachers. In *Interspeech*, pages 3697–3701, 2017.

[63] Tommaso Furlanello, Zachary C Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. *arXiv preprint arXiv:1805.04770*, 2018.

[64] Inés María Galván, Pedro Isasi, Ricardo Aler, and José María Valls. A selective learning method to improve the generalization of multilayer feedforward neural networks. *International journal of neural systems*, 11(02):167–177, 2001.

[65] Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J Guibas. Geometry guided convolutional neural networks for self-supervised video representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5589–5597, 2018.

[66] Chuang Gan, Ting Yao, Kuiyuan Yang, Yi Yang, and Tao Mei. You lead, we exceed: Labor-free video concept learning by jointly exploiting web videos and images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 923–932, 2016.

[67] Chuang Gan, Hang Zhao, Peihao Chen, David Cox, and Antonio Torralba. Self-supervised moving vehicle tracking with stereo sound. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7053–7062, 2019.

[68] Jiyang Gao, Zhen Li, Ram Nevatia, et al. Knowledge concentration: Learning 100k object classifiers in a single cnn. *arXiv preprint arXiv:1711.07607*, 2017.

[69] Liang Gao, Xu Lan, Haibo Mi, Dawei Feng, Kele Xu, and Yuxing Peng. Multistructure-based collaborative online distillation. *Entropy*, 21(4):357, 2019.

[70] Liang Gao, Haibo Mi, Boqing Zhu, Dawei Feng, Yicong Li, and Yuxing Peng. An adversarial feature distillation method for audio classification. *IEEE Access*, 7:105319–105330, 2019.

[71] Shiming Ge, Shengwei Zhao, Chenyu Li, and Jia Li. Low-resolution face recognition in the wild via selective knowledge distillation. *IEEE Transactions on Image Processing*, 28(4):2051–2062, 2018.

[72] Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. *arXiv preprint arXiv:1905.09747*, 2019.

[73] Chen Gong, Xiaojun Chang, Meng Fang, and Jian Yang. Teaching semi-supervised classifier via generalized distillation. In *IJCAI*, pages 2156–2162, 2018.

[74] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[75] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[76] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017.

[77] Xiaoyang Guo, Hongsheng Li, Shuai Yi, Jimmy Ren, and Xiaogang Wang. Learning monocular depth by distilling cross-domain stereo networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 484–500, 2018.

[78] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2827–2836, 2016.

[79] Steven Gutstein, Olac Fuentes, and Eric Freudenthal. Knowledge transfer in deep convolutional neural nets. *International Journal on Artificial Intelligence Tools*, 17(03):555–567, 2008.

[80] Frank Hafner, Amran Bhuiyan, Julian FP Kooij, and Eric Granger. A cross-modal distillation network for person re-identification in rgb-depth. *arXiv preprint arXiv:1810.11641*, 2018.

[81] Sangchul Hahn and Heeyoul Choi. Self-knowledge distillation in natural language processing. *arXiv preprint arXiv:1908.01851*, 2019.

[82] Md Akmal Haidar and Mehdi Rezagholizadeh. Textkd-gan: text generation using knowledge distillation and generative adversarial networks. In *Canadian Conference on Artificial Intelligence*, pages 107–118. Springer, 2019.

[83] Zeyad Hailat and Xue-Wen Chen. Teacher/student deep semi-supervised learning for training with noisy labels. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 907–912. IEEE, 2018.

[84] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, pages 1024–1034, 2017.

[85] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

[86] Yu Hao, Yanwei Fu, Yu-Gang Jiang, and Qi Tian. An end-to-end architecture for class-incremental object detection with knowledge distillation. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2019.

[87] Matan Haroush, Itay Hubara, Elad Hoffer, and Daniel Soudry. The knowledge within: Methods for data-free model compression. *arXiv preprint arXiv:1912.01274*, 2019.

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

[89] Xiaoxi He, Zimu Zhou, and Lothar Thiele. Multi-task zipping via layer-wise neuron sharing. In *Advances in Neural Information Processing Systems*, pages 6016–6026, 2018.

[90] Srinidhi Hegde, Ranjitha Prasad, Ramya Hebbalaguppe, and Vishwajith Kumar. Variational student: Learning compact and sparser networks in knowledge distillation framework. *arXiv preprint arXiv:1910.12061*, 2019.

[91] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation, 2019.

[92] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons, 2018.

[93] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge distillation with adversarial samples supporting decision boundary. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3771–3778, 2019.

[94] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[95] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[96] Judy Hoffman, Saurabh Gupta, Jian Leong, Sergio Guadarrama, and Trevor Darrell. Cross-modal adaptation for rgb-d detection. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5032–5039. IEEE, 2016.

[97] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

[98] Wei Hong and Jingke Yu. Gan-knowledge distillation for one-stage object detection. *arXiv preprint arXiv:1906.08467*, 2019.

[99] Zhang-Wei Hong, Prabhat Nagarajan, and Guilherme Maeda. Periodic intra-ensemble knowledge distillation for reinforcement learning. *arXiv preprint arXiv:2002.00149*, 2020.

[100] Franziska Horn and Klaus-Robert Müller. Learning similarity preserving representations with neural similarity and context encoders. 2016.

[101] Saihui Hou, Xu Liu, and Zilei Wang. Dualnet: Learn complementary features for image recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 502–510, 2017.

[102] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019.

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

[104] Dong-Hyun Hwang, Suntae Kim, Nicolas Monet, Hideki Koike, and Soonmin Bae. Lightweight 3d human pose estimation network training using teacher-student learning. *arXiv preprint arXiv:2001.05097*, 2020.

[105] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[106] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[107] Haibo Jin, Shifeng Zhang, Xiangyu Zhu, Yinhang Tang, Zhen Lei, and Stan Z Li. Learning lightweight face detector with knowledge distillation. In *2019 International Conference on Biometrics (ICB)*, pages 1–7. IEEE, 2019.

[108] Xiao Jin, Baoyun Peng, Yichao Wu, Yu Liu, Jiaheng Liu, Ding Liang, Junjie Yan, and Xiaolin Hu. Knowledge distillation via route constrained optimization. *arXiv preprint arXiv:1904.09149*, 2019.

[109] Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Uncertainty-aware multi-shot knowledge distillation for image-based object re-identification. *arXiv preprint arXiv:2001.05197*, 2020.

[110] Jee-weon Jung, HeeSoo Heo, Hye-jin Shim, and Ha-Jin Yu. Distilling the knowledge of specialist deep neural networks in acoustic scene classification. 2019.

[111] Jayashree Karlekar, Jiashi Feng, Zi Sian Wong, and Sugiri Pranata. Deep face recognition model compression via knowledge transfer and distillation. *arXiv preprint arXiv:1906.00619*, 2019.

[112] Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy Ren, and Rynson WH Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6728–6736, 2019.

[113] Changil Kim, Hijung Valentina Shin, Tae-Hyun Oh, Alexandre Kaspar, Mohamed Elgharib, and Wojciech Matusik. On learning associations of faces and voices. In *Asian Conference on Computer Vision*, pages 276–292. Springer, 2018.

[114] Jangho Kim, Minsung Hyun, Inseop Chung, and Nojun Kwak. Feature fusion for online mutual knowledge distillation. *arXiv preprint arXiv:1904.09058*, 2019.

[115] Jangho Kim, SeongUk Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. In *Advances in Neural Information Processing Systems*, pages 2760–2769, 2018.

[116] Akisato Kimura, Zoubin Ghahramani, Koh Takeuchi, Tomoharu Iwata, and Naonori Ueda. Few-shot learning of neural networks from scratch by pseudo example optimization. *arXiv preprint arXiv:1802.03039*, 2018.

[117] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[118] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[119] Srinivas SS Kruthiventi, Pratyush Sahay, and Rajesh Biswal. Low-light pedestrian detection from rgb images using multi-modal knowledge distillation. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 4207–4211. IEEE, 2017.

[120] Akshay Kulkarni, Navid Panchi, and Shital Chiddarwar. Stage-wise knowledge distillation, 2019.

[121] Mandar Kulkarni, Kalpesh Patil, and Shirish Karande. Knowledge distillation using unlabeled mismatched images. *arXiv preprint arXiv:1703.07131*, 2017.

[122] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.

[123] Xu Lan, Xiatian Zhu, and Shaogang Gong. Knowledge distillation by on-the-fly native ensemble. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 7528–7538. Curran Associates Inc., 2018.

[124] Carlos Lassance, Myriam Bontonou, Ghouthi Boukli Hacene, Vincent Gripon, Jian Tang, and Antonio Ortega. Deep geometric knowledge distillation with graphs. *arXiv preprint arXiv:1911.03080*, 2019.

[125] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[126] Hankook Lee, Sung Ju Hwang, and Jinwoo Shin. Rethinking data augmentation: Self-supervision and self-distillation. *arXiv preprint arXiv:1910.05872*, 2019.

[127] Kwangjin Lee, Luong Trung Nguyen, and Byonghyo Shim. Stochasticity and skip connections improve knowledge transfer. 2019.

[128] Seung Hyun Lee, Dae Ha Kim, and Byung Cheol Song. Self-supervised knowledge distillation using singular value decomposition. In *European Conference on Computer Vision*, pages 339–354. Springer, 2018.

[129] Seunghyun Lee and Byung Cheol Song. Graph-based knowledge distillation by multi-head self-attention network. *arXiv preprint arXiv:1907.02226*, 2019.

[130] Yeonkun Lee, Jaeseok Jeong, Jongseob Yun, Wonjune Cho, and Kuk-Jin Yoon. Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9181–9189, 2019.

[131] Yunsoo Lee, Namhyun Ahn, Jun Ho Heo, So Yeon Jo, and Suk-Ju Kang. Teaching where to see: Knowledge distillation-based attentive information transfer in vehicle maker classification. *IEEE Access*, 7:86412–86420, 2019.

[132] Cheng Li, Xiaoxiao Guo, and Qiaozhu Mei. Deepgraph: Graph structure predicts network growth. *arXiv preprint arXiv:1610.06251*, 2016.

[133] Hao-Ting Li, Shih-Chieh Lin, Cheng-Yeh Chen, and Chen-Kuo Chiang. Layer-level knowledge distillation for deep neural network learning. *Applied Sciences*, 9(10):1966, 2019.

[134] Kunpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Attention bridging network for knowledge transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5198–5207, 2019.

[135] Muyang Li, Ji Lin, Yaoyao Ding, Zhijian Liu, Jun-Yan Zhu, and Song Han. Gan compression: Efficient architectures for interactive conditional gans. *arXiv preprint arXiv:2003.08936*, 2020.

[136] Tianhong Li, Jianguo Li, Zhuang Liu, and Changshui Zhang. Few sample knowledge distillation for efficient network compression. *arXiv preprint arXiv:1812.01839*, 2018.

[137] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1910–1918, 2017.

[138] Kaixiang Lin, Shu Wang, and Jiayu Zhou. Collaborative deep reinforcement learning. *arXiv preprint arXiv:1702.05796*, 2017.

[139] Rongcheng Lin, Jing Xiao, and Jianping Fan. Mod: A deep mixture model with online knowledge distillation for large scale video temporal concept localization. *arXiv preprint arXiv:1910.12295*, 2019.

[140] Iou-Jen Liu, Jian Peng, and Alexander G Schwing. Knowledge flow: Improve upon your teachers. *arXiv preprint arXiv:1904.05878*, 2019.

[141] Jian Liu, Yubo Chen, and Kang Liu. Exploiting the ground-truth: An adversarial imitation based knowledge distillation approach for event detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6754–6761, 2019.

[142] Jingen Liu, Mubarak Shah, Benjamin Kuipers, and Silvio Savarese. Cross-view action recognition via view knowledge transfer. In *CVPR 2011*, pages 3209–3216. IEEE, 2011.

[143] Linqing Liu, Huan Wang, Jimmy Lin, Richard Socher, and Caiming Xiong. Attentive student meets multi-task teacher: Improved knowledge distillation for pretrained models. *arXiv preprint arXiv:1911.03588*, 2019.

[144] Peiye Liu, Wu Liu, Huadong Ma, Tao Mei, and Mingoo Seok. Ktan: knowledge transfer adversarial network. *arXiv preprint arXiv:1810.08126*, 2018.

[145] Pengpeng Liu, Irwin King, Michael R Lyu, and Jia Xu. Ddflow: Learning optical flow with unlabeled data distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8770–8777, 2019.

[146] Qing Liu, Lingxi Xie, Huiyu Wang, and Alan L Yuille. Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3662–3671, 2019.

[147] Ruishan Liu, Nicolo Fusi, and Lester Mackey. Teacher-student compression with generative adversarial networks. *arXiv preprint arXiv:1812.02271*, 2018.

[148] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Improving multi-task deep neural networks via knowledge distillation for natural language understanding. *arXiv preprint arXiv:1904.09482*, 2019.

[149] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019.

[150] Yuanpei Liu, Xingping Dong, Wenguan Wang, and Jianbing Shen. Teacher-students knowledge distillation for siamese trackers. *arXiv preprint arXiv:1907.10586*, 2019.

[151] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. Knowledge distillation via

instance relationship graph. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7096–7104, 2019.

[152] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[153] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017.

[154] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

[155] Yunteng Luan, Hanyu Zhao, Zhi Yang, and Yafei Dai. Msd: Multi-self-distillation learning via multi-classifiers within deep neural networks. *arXiv preprint arXiv:1911.09418*, 2019.

[156] Ping Luo, Zhenyao Zhu, Ziwei Liu, Xiaogang Wang, and Xiaoou Tang. Face model compression by distilling knowledge from neurons. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

[157] Sihui Luo, Xinchao Wang, Gongfan Fang, Yao Hu, Dapeng Tao, and Mingli Song. Knowledge amalgamation from heterogeneous networks by common feature learning. *arXiv preprint arXiv:1906.10546*, 2019.

[158] Yucen Luo, Jun Zhu, Mengxi Li, Yong Ren, and Bo Zhang. Smooth neighbors on teacher graphs for semi-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8896–8905, 2018.

[159] Zelun Luo, Jun-Ting Hsieh, Lu Jiang, Juan Carlos Niebles, and Li Fei-Fei. Graph distillation for action detection with privileged modalities. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 166–183, 2018.

[160] Jiaqi Ma and Qiaozhu Mei. Graph representation learning via multi-task knowledge distillation. *arXiv preprint arXiv:1911.05700*, 2019.

[161] Andrey Malinin, Bruno Mlodozeniec, and Mark Gales. Ensemble distribution distillation. *arXiv preprint arXiv:1905.00076*, 2019.

[162] Karttikeya Mangalam and Mathieu Salzamann. On compressing u-net using knowledge distillation. *arXiv preprint arXiv:1812.00249*, 2018.

[163] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2017.

[164] Angel Martínez-González, Michael Villamizar, Olivier Canévet, and Jean-Marc Odobez. Efficient convolutional neural networks for depth-based multi-person pose estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[165] Mehak Mehak and Vineeth N Balasubramanian. *Knowledge Distillation from Multiple Teachers using Visual Explanations*. PhD thesis, Indian Institute of Technology Hyderabad, 2018.

[166] Zhong Meng, Jinyu Li, Yashesh Gaur, and Yifan Gong. Domain adaptation via teacher-student learning for end-to-end speech recognition. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 268–275. IEEE, 2019.

[167] Paul Micaelli and Amos J Storkey. Zero-shot knowledge transfer via adversarial belief matching. In *Advances in Neural Information Processing Systems*, pages 9547–9557, 2019.

[168] Umberto Michieli and Pietro Zanuttigh. Knowledge distillation for incremental learning in semantic segmentation. *arXiv preprint arXiv:1911.03462*, 2019.

[169] Soma Minami, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. Knowledge transfer graph for deep collaborative learning, 2019.

[170] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[171] Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant: Bridging the gap between student and teacher. *arXiv preprint arXiv:1902.03393*, 2019.

[172] Asit Mishra and Debbie Marr. Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. *arXiv preprint arXiv:1711.05852*, 2017.

[173] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

[174] Hossein Mobahi, Mehrdad Farajtabar, and Peter L Bartlett. Self-distillation amplifies regularization in hilbert space. *arXiv preprint arXiv:2002.05715*, 2020.

[175] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. 2015.

[176] Ravi Teja Mullapudi, Steven Chen, Keyi Zhang, Deva Ramanan, and Kayvon Fatahalian. Online model distillation for efficient video inference. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3573–3582, 2019.

[177] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Learnable pins: Cross-modal embeddings for person identity. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 71–88, 2018.

[178] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Seeing voices and hearing faces: Cross-modal biometric matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8427–8436, 2018.

[179] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, R Venkatesh Babu, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. *arXiv preprint arXiv:1905.08114*, 2019.

[180] Xuecheng Nie, Yuncheng Li, Linjie Luo, Ning Zhang, and Jiashi Feng. Dynamic kernel distillation for efficient pose estimation in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6942–6950, 2019.

[181] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9359–9367, 2018.

[182] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2642–2651. JMLR. org, 2017.

[183] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106(5):808–828, 2018.

[184] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *European conference on computer vision*, pages 801–816. Springer, 2016.

[185] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. *arXiv preprint arXiv:2003.13942*, 2020.

[186] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*, 2016.

[187] SeongUk Park and Nojun Kwak. Feed: Feature-level ensemble for knowledge distillation, 2019.

[188] Sung Woo Park and Junseok Kwon. Sphere generative adversarial network based on geometric moment matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4292–4301, 2019.

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

[190] Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 268–284, 2018.

[191] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5007–5016, 2019.

[192] Mary Phuong and Christoph H Lampert. Distillation-based training for multi-exit architectures. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1355–1364, 2019.

[193] Andrea Pilzer, Stephane Lathuiliere, Nicu Sebe, and Elisa Ricci. Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9768–9777, 2019.

[194] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised

learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4119–4128, 2018.

[195] Siddharth Roheda, Benjamin S Riggan, Hamid Krim, and Liyi Dai. Cross-modality distillation: A case for conditional generative adversarial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2926–2930. IEEE, 2018.

[196] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[197] Sebastian Ruder, Parsa Ghaffari, and John G Breslin. Knowledge adaptation: Teaching to adapt. *arXiv preprint arXiv:1702.02052*, 2017.

[198] Adrià Ruiz and Jakob Verbeek. Distilled hierarchical neural ensembles with adaptive inference cost. *arXiv preprint arXiv:2003.01474*, 2020.

[199] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. *arXiv preprint arXiv:1511.06295*, 2015.

[200] Tawfiq Salem, Connor Greenwell, Hunter Blanton, and Nathan Jacobs. Learning to map nearly anything. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 4803–4806. IEEE, 2019.

[201] Muhamad Risqi U Saputra, Pedro PB de Gusmao, Yasin Almalioglu, Andrew Markham, and Niki Trigoni. Distilling knowledge from a deep pose regressor network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 263–272, 2019.

[202] Fahad Sarfraz, Elahe Arani, and Bahram Zonooz. Noisy collaboration in knowledge distillation. 2019.

[203] Bharat Bhusan Sau and Vineeth N Balasubramanian. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*, 2016.

[204] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

[205] Jo Schlemper, Ozan Oktay, Michiel Schaap, Mattias Heinrich, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention gated networks: Learning to leverage salient regions in medical images. *Medical image analysis*, 53:197–207, 2019.

[206] Yuhu Shan. Distilling pixel-wise feature similarities for semantic segmentation. *arXiv preprint arXiv:1910.14226*, 2019.

[207] Chengchao Shen, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. Amalgamating knowledge towards comprehensive classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3068–3075, 2019.

[208] Chengchao Shen, Mengqi Xue, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. Customizing student networks from heterogeneous teachers via adaptive knowledge amalgamation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3504–3513, 2019.

[209] Zhiqiang Shen, Zhankui He, Wanyun Cui, Jiahui Yu, Yutong Zheng, Chenchen Zhu, and Marios Savvides. Adversarial-based knowledge distillation for multi-model ensemble and noisy data refinement. *arXiv preprint arXiv:1908.08520*, 2019.

[210] Zhiqiang Shen, Zhankui He, and Xiangyang Xue. Meal: Multi-model ensemble via adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4886–4893, 2019.

[211] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3400–3409, 2017.

[212] Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264, 2006.

[213] Guocong Song and Wei Chai. Collaborative learning for deep neural networks. In *Advances in Neural Information Processing Systems*, pages 1832–1841, 2018.

[214] Suraj Srinivas and François Fleuret. Knowledge transfer with jacobian matching. *arXiv preprint arXiv:1803.00443*, 2018.

[215] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[216] Jong-Chyi Su and Subhransu Maji. Adapting models to signal degradation using distillation. *arXiv preprint arXiv:1604.00433*, 2016.

[217] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*, 2019.

[218] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018.

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

[220] Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. Multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:1902.10461*, 2019.

[221] Shitao Tang, Litong Feng, Wenqi Shao, Zhanghui Kuang, Wei Zhang, and Yimin Chen. Learning efficient detector with semi-supervised adaptive distillation. *arXiv preprint arXiv:1901.00366*, 2019.

[222] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017.

[223] Fida Mohammad Thoker and Juergen Gall. Cross-modal knowledge distillation for action recognition. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 6–10. IEEE, 2019.

[224] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation, 2019.

[225] Fabio Tosi, Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia. Learning monocular depth estimation infusing traditional stereo knowledge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9799–9809, 2019.

[226] Fabio Tosi, Filippo Aleotti, Pierluigi Zama Ramirez, Matteo Poggi, Samuele Salti, Luigi Di Stefano, and Stefano Mattoccia. Distilled semantics for comprehensive scene understanding from videos. *arXiv preprint arXiv:2003.14030*, 2020.

[227] Linh Tran, Bastiaan S Veeling, Kevin Roth, Jakub Swiatkowski, Joshua V Dillon, Jasper Snoek, Stephan Mandt, Tim Salimans, Sebastian Nowozin, and Rodolphe Jenatton. Hydra: Preserving ensemble diversity for model distillation. *arXiv preprint arXiv:2001.04694*, 2020.

[228] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7472–7481, 2018.

[229] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1365–1374, 2019.

[230] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[231] Jayakorn Vongkulbhisal, Phongtharin Vinayavekhin, and Marco Visentini-Scarzanella. Unifying heterogeneous classifiers with distillation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[232] Jayakorn Vongkulbhisal, Phongtharin Vinayavekhin, and Marco Visentini-Scarzanella. Unifying heterogeneous classifiers with distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3175–3184, 2019.

[233] Chaoyang Wang, Chen Kong, and Simon Lucey. Distill knowledge from nrsfm for weakly supervised 3d pose learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 743–752, 2019.

[234] Donghui Wang, Yanan Li, Yuetan Lin, and Yueting Zhuang. Relational knowledge transfer for zero-shot learning. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[235] Huan Wang, Yijun Li, Yuehai Wang, Haoji Hu, and Ming-Hsuan Yang. Collaborative distillation for ultra-resolution universal style transfer. *arXiv*, pages arXiv–2003, 2020.

[236] Junjie Wang, Xiangfeng Wang, Bo Jin, Junchi Yan, Wenjie Zhang, and Hongyuan Zha. Heterogeneous graph-based knowledge transfer for generalized zero-shot learning, 2019.

[237] Lin Wang, Wonjune Cho, and Kuk-Jin Yoon. Deceiving image-to-image translation networks for autonomous driving with adversarial perturbations. *arXiv preprint arXiv:2001.01506*, 2020.

[238] Lin Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. *arXiv preprint arXiv:2003.07640*, 2020.

[239] Lin Wang, S. Mohammad Mostafavi I., Yo-Sung Ho, Kuk-Jin Yoon, et al. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10081–10090, 2019.

[240] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4933–4942, 2019.

[241] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv preprint arXiv:2002.10957*, 2020.

[242] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. Kdgan: knowledge distillation with generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 775–786, 2018.

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

[244] Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. Minegan: effective knowledge transfer from gans to target domains with few images. *arXiv preprint arXiv:1912.05270*, 2019.

[245] Yunhe Wang, Chang Xu, Chao Xu, and Dacheng Tao. Adversarial learning of portable student networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[246] Tiancheng Wen, Shenqi Lai, and Xueming Qian. Preparing lessons: Improve knowledge distillation with better supervision. *arXiv preprint arXiv:1911.07471*, 2019.

[247] Ancong Wu, Wei-Shi Zheng, Xiaowei Guo, and Jian-Huang Lai. Distilled person re-identification: Towards a more scalable system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1187–1196, 2019.

[248] Meng-Chieh Wu and Ching-Te Chiu. Multi-teacher knowledge distillation for compressed video action recognition based on deep learning. *Journal of Systems Architecture*, 103:101695, 2020.

[249] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019.

[250] Liuyu Xiang and Guiguang Ding. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. *arXiv preprint arXiv:2001.01536*, 2020.

[251] Jiafeng Xie, Bing Shuai, Jian-Fang Hu, Jingyang Lin, and Wei-Shi Zheng. Improving fast segmentation with teacher-student learning. *arXiv preprint arXiv:1810.08476*, 2018.

[252] Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Self-training with noisy student improves imagenet classification. *arXiv preprint arXiv:1911.04252*, 2019.

[253] Jiaolong Xu, Yiming Nie, Peng Wang, and Antonio M López. Training a binary weight object detector by knowledge transfer for autonomous driving. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2379–2384. IEEE, 2019.

[254] Ting-Bing Xu and Cheng-Lin Liu. Data-distortion guided self-distillation for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5565–5572, 2019.

[255] Xixia Xu, Qi Zou, Xue Lin, Yaping Huang, and Yi Tian. Integral knowledge distillation for multi-person pose estimation. *IEEE Signal Processing Letters*, 2020.

[256] Yixing Xu, Yunhe Wang, Hanting Chen, Kai Han, XU Chunjing, Dacheng Tao, and Chang Xu. Positive-unlabeled compression on the cloud. In *Advances in Neural Information Processing Systems*, pages 2561–2570, 2019.

[257] Yonghao Xu, Bo Du, Lefei Zhang, Qian Zhang, Guoli Wang, and Liangpei Zhang. Self-ensembling attention networks: Addressing domain shift for semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5581–5588, 2019.

[258] Zheng Xu, Yen-Chang Hsu, and Jiawei Huang. Learning loss for knowledge distillation with conditional adversarial networks. *arXiv preprint arXiv:1709.00513*, 2017.

[259] Zheng Xu, Yen-Chang Hsu, and Jiawei Huang. Training shallow and thin networks for acceleration via knowledge distillation with conditional adversarial networks. *arXiv preprint arXiv:1709.00513*, 2017.

[260] Zheng Xu, Yen-Chang Hsu, and Jiawei Huang. Training student networks for acceleration with conditional adversarial networks. In *BMVC*, page 61, 2018.

[261] Zeyue Xue, Shuang Luo, Chao Wu, Pan Zhou, Kaigui Bian, and Wei Du. Transfer heterogeneous knowledge among peer-to-peer teammates: A model distillation approach. *arXiv preprint arXiv:2002.02202*, 2020.

[262] Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan L Yuille. Training deep neural networks in generations: A more tolerant teacher educates better students. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5628–5635, 2019.

[263] Chenglin Yang, Lingxi Xie, Chi Su, and Alan L Yuille. Snapshot distillation: Teacher-student optimization in one generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2859–2868, 2019.

[264] Ze Yang, Linjun Shou, Ming Gong, Wutao Lin, and Daxin Jiang. Model compression with two-stage multi-teacher knowledge distillation for web question answering system. *arXiv preprint arXiv:1910.08381*, 2019.

[265] Ze Yang, Linjun Shou, Ming Gong, Wutao Lin, and Daxin Jiang. Model compression with two-stage multi-teacher knowledge distillation for web question answering system. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 690–698, 2020.

[266] Huaxiu Yao, Chuxu Zhang, Ying Wei, Meng Jiang, Suhang Wang, Junzhou Huang, Nitesh V Chawla, and Zhenhui Li. Graph few-shot learning via knowledge transfer. *arXiv preprint arXiv:1910.03053*, 2019.

[267] Jingwen Ye, Yixin Ji, Xinchao Wang, Xin Gao, and Mingli Song. Data-free knowledge amalgamation via group-stack dual-gan, 2020.

[268] Jingwen Ye, Yixin Ji, Xinchao Wang, Kairi Ou, Dapeng Tao, and Mingli Song. Student becoming the master: Knowledge amalgamation for joint scene parsing, depth estimation, and more. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2829–2838, 2019.

[269] Jingwen Ye, Xinchao Wang, Yixin Ji, Kairi Ou, and Mingli Song. Amalgamating filtered knowledge: learning task-customized student from multi-task teachers. *arXiv preprint arXiv:1905.11569*, 2019.

[270] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4133–4141, 2017.

[271] Hongxu Yin, Pavlo Molchanov, Zhizhong Li, Jose M Alvarez, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. *arXiv preprint arXiv:1912.08795*, 2019.

[272] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems*, pages 4800–4810, 2018.

[273] Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. In *Advances in Neural Information Processing Systems*, pages 2701–2710, 2019.

[274] Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294, 2017.

[275] Lu Yu, Vacit Oguz Yazici, Xialei Liu, Joost van de Weijer, Yongmei Cheng, and Arnau Ramisa. Learning metrics from teachers: Compact networks for image embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2907–2916, 2019.

[276] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural

networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.
[277] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.
[278] Mengyao Zhai, Lei Chen, Frederick Tung, Jiawei He, Megha Nawhal, and Greg Mori. Lifelong gan: Continual learning for conditional image generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2759–2768, 2019.
[279] Chenrui Zhang and Yuxin Peng. Better and faster: knowledge transfer from multiple self-supervised learning tasks via graph distillation for video classification. *arXiv preprint arXiv:1804.10069*, 2018.
[280] Feng Zhang, Xiatian Zhu, and Mao Ye. Fast human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3517–3526, 2019.
[281] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3713–3722, 2019.
[282] Shifeng Zhang, Jianmin Li, and Bo Zhang. Pairwise teacher-student network for semi-supervised hashing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.
[283] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4320–4328, 2018.
[284] Zhi Zhang, Guanghan Ning, and Zhihai He. Knowledge projection for effective design of thinner and faster deep neural networks. 2017.
[285] Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zhengjun Zha. Object relational graph with teacher-recommended learning for video captioning. *arXiv preprint arXiv:2002.11566*, 2020.
[286] Albert Zhao, Tong He, Yitao Liang, Haibin Huang, Guy Van den Broeck, and Stefano Soatto. Lates: Latent space distillation for teacher-student driving policy learning. *arXiv preprint arXiv:1912.02973*, 2019.
[287] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018.
[288] Rui-Wei Zhao, Jianguo Li, Yurong Chen, Jia-Ming Liu, Yu-Gang Jiang, and Xiangyang Xue. Regional gating neural networks for multi-label image classification. In *BMVC*, pages 1–12, 2016.
[289] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.
[290] Peng Zhou, Long Mai, Jianming Zhang, Ning Xu, Zuxuan Wu, and Larry S Davis. M2kd: Multi-model and multi-level knowledge distillation for incremental learning. *arXiv preprint arXiv:1904.01769*, 2019.
[291] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in neural information processing systems*, pages 465–476, 2017.
[292] Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression. *arXiv preprint arXiv:1710.01878*, 2017.
[293] Xiatian Zhu, Shaogang Gong, et al. Knowledge distillation by on-the-fly native ensemble. In *Advances in neural information processing systems*, pages 7517–7527, 2018.

Update information: April 15th, 2020

- Added a new section: KD for video understanding
- Added latest CVPR 2020 papers and other resources.