UniKD: Universal Knowledge Distillation for Mimicking Homogeneous or Heterogeneous Object Detectors

Shanshan Lao[1*] Guanglu Song[2] Boxiao Liu[2] Yu Liu[2†] Yujiu Yang[1†]
[1] Tsinghua Shenzhen International Graduate School, Tsinghua University
[2] Sensetime Research
laoss21@mails.tsinghua.edu.cn, {songguanglu, liuboxiao}@sensetime.com
liuyuisanai@gmail.com, yang.yujiu@sz.tsinghua.edu.cn

Abstract

Knowledge distillation (KD) has become a standard method to boost the performance of lightweight object detectors. Most previous works are feature-based, where students mimic the features of homogeneous teacher detectors. However, distilling the knowledge from the heterogeneous teacher fails in this manner due to the serious semantic gap, which greatly limits the flexibility of KD in practical applications. Bridging this semantic gap now requires case-by-case algorithm design which is time-consuming and heavily relies on experienced adjustment. To alleviate this problem, we propose Universal Knowledge Distillation (UniKD), introducing additional decoder heads with deformable cross-attention called Adaptive Knowledge Extractor (AKE). In UniKD, AKEs are first pretrained on the teacher's output to infuse the teacher's content and positional knowledge into a fixed-number set of knowledge embeddings. The fixed AKEs are then attached to the student's backbone to encourage the student to absorb the teacher's knowledge in these knowledge embeddings. In this query-based distillation paradigm, detection-relevant information can be dynamically aggregated into a knowledge embedding set and transferred between different detectors. When the teacher model is too large for online inference, its output can be stored on disk in advance to save the computation overhead, which is more storage efficient than feature-based methods. Extensive experiments demonstrate that our UniKD can plug and play in any homogeneous or heterogeneous teacher-student pairs and significantly outperforms conventional feature-based KD.

1. Introduction

Object detection is a fundamental computer vision task that has been widely applied to many practical applications.

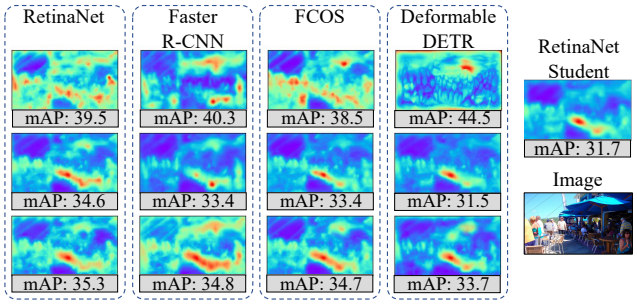


Figure 1: The feature density of different models. First row: four teachers. Second row: students distilled with FitNet. Third row: students distilled with UniKD.

In recent years, various frameworks for object detection have been proposed to improve tdetection performance such as Faster R-CNN [19], RetinaNet [13], FCOS [21], and Deformable DETR [37]. In practical applications, different detectors often favor different devices due to constraints on parameter number, inference latency, and even the detector framework. For example, the high-performance Faster R-CNN is not friendly to some edge devices due to the RoI pooling operator but is often deployed in many cloud devices. Therefore, boosting the accuracy of deployable models in different detector frameworks as much as possible is the core problem in practical applications.

Knowledge Distillation (KD) [20, 6, 31] methods have significantly boosted the performance of lightweight student via learning from a high-capacity teacher model. This learning paradigm explored by researchers on various visual tasks including high-level visual understanding tasks [19, 13], as well as low-level visual tasks [25, 8]. Current detection KD methods are mostly feature-based [20, 28, 29, 26]. For example, FGD [28] improves the RetinaNet-Res50 from 37.4 mAP to 39.6 mAP by mimicking the RetinaNet-Res101. However, these methods are not general enough because they only consider homogeneous pairs of teachers and students with the same framework. **How to flex-**

*Work done during internship at SenseTime Research.
†Corresponding authors.

ibly transfer the knowledge in heterogeneous teacher-student pairs, such as from RetineNet-Res101 to Deformable DETR-Res18, is a practical and challenging topic for object detection. Unfortunately, conventional feature-based distillation fails in directly distilling knowledge from the heterogeneous teacher due to a serious semantic gap. As shown in Fig. 1, activation maps from different detection frameworks indicate significantly different semantics. We further show the feature maps of students after feature-based KD in the second row, which demonstrates that it's hard for students to imitate the teachers' output completely and this even impairs the students' performance. To alleviate this dilemma, several works [18, 23] have tried to bridge this semantic gap by introducing assistants to guide the optimization of the student detector. However, these assistants require a case-by-case algorithm design to adapt to different teacher-student pairs, which heavily relies on experienced adjustment and greatly limits its flexibility.

Additionally, to obtain supervision from the teacher in each iteration, large amounts of computing resources and training time are consumed when training data pass through the giant teacher networks, which is inefficient and costly. A straightforward solution, called offline KD, is to store the multi-scale teacher features of each training image in advance and then reuse them during distillation to avoid the storage-consuming online inference of teachers. However, the storage cost of the current feature-based distillation method is unacceptable.

In this paper, we propose a query-based distillation paradigm called Universal Knowledge Distillation (UniKD) to flexibly transfer knowledge in any homogeneous or heterogeneous teacher-student pairs. The advantages of such a query-based paradigm are threefold: **(1)** Given a high-capacity teacher model trained in any popular detection frameworks, we can directly boost the performance of lightweight detectors, whether they're homogeneous or heterogeneous. **(2)** UniKD is a general knowledge distillation paradigm with zero-cost algorithm adjustment in different practical applications without time-consuming case-by-case design. **(3)** In contrast to distilling the whole feature map, query-based UniKD extracts the teachers' knowledge into a small number of knowledge embeddings, which requires significantly less storage than feature-based methods in offline KD and even performs better. See Tab. 4.

In UniKD, we introduce Adaptive Knowledge Extractor (AKE) modules, which are additional transformer decoder heads with deformable cross-attention. The AKE modules use content queries (q_{ct}) and positional queries (q_{pos}) as probes to extract the detection-relevant knowledge from the network's intermediate output for distillation. Specifically, this paradigm has two stages. In the first stage, given a high-capacity teacher model, q_{ct} and q_{pos} are first generated to absorb the teacher's knowledge via the AKE modules. AKE modules are pre-trained at this stage to be capable of extracting detection-relevant knowledge. In the second stage, the AKEs with frozen q_{ct}, q_{pos} and parameter weights are attached to the output multi-scale features of the student to imitate the teacher's output extracted by the same AKEs. This encourages the student to absorb the teacher's knowledge in knowledge embeddings. The proposed AKE module is detector-agnostic and can be applied to arbitrary detectors for extracting detection-relevant knowledge, which can be transferred to any student detectors to boost their performance.

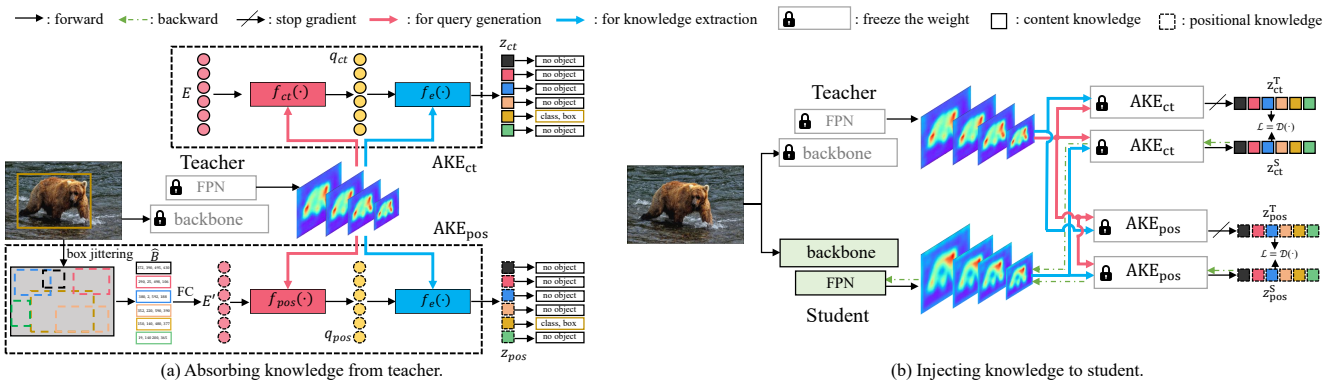
We conduct extensive experiments to demonstrate the generalization of our method, which achieves better or comparable results on homogeneous or heterogeneous teacher-student pairs compared to existing works. To our best knowledge, we are the first to implement the knowledge transferring between traditional detectors and end-to-end Deformable DETR. We boost the performance of RetinaNet by 2.0 mAP on MS-COCO 2017 dataset by mimicking Deformable DETR, and in turn, Deformable DETR can also obtain 2.0 mAP improvement by imitating the RetinaNet. Even in homogeneous teacher-student pairs, the proposed UniKD still outperforms the previous state-of-the-art methods and establishes new advanced results. In summary, the contributions of this paper are as follows:

- We propose the query-based Universal Knowledge Distillation, which is a new knowledge distillation paradigm for transferring information in homogeneous or heterogeneous teacher-student pairs.
- We introduce AKE modules with content queries and positional queries to extract detection-relevant knowledge. It requires zero-cost algorithm adjustment when applied to different detectors and has high storage efficiency for offline KD.
- We conduct extensive experiments on various teacher-student pairs and model architectures to verify the effectiveness and universality of UniKD. Especially, to our best knowledge, we are the first to effectively transfer the detection-relevant knowledge between conventional detectors and end-to-end Deformable DETR.

2. Related Works

2.1. Object Detection

Currently, modern object detectors can be roughly divided into three types: Two-stage, One-stage, and End-to-end detectors. Two-stage detectors generate region proposals at first and refine them in the second stage, with Faster R-CNN [19] as a typical example. Some one-stage detectors use anchors prior to detecting objects, such as SSD [15] and RetinaNet [13], called anchor-based detectors. Other



(a) Absorbing knowledge from teacher.

(b) Injecting knowledge to student.

Figure 2: (a) Pretraining of AKE modules, including query generation and detection-relevant knowledge extraction. In this stage, the parameter of teacher is fixed, and only the AKE modules are optimized. (b) In this distillation stage, the AKE modules are fixed. The gradients generated by $\mathcal{D}(\cdot)$ only update the parameter of the student. Note that the query generation in all AKEs is based on the teacher's output. The original detection losses are omitted for clear visualization.

one-stage detectors make box predictions directly on spatial points [21, 30] or group keypoints on a heatmap to model objects [9, 36], called anchor-free detectors. End-to-end detectors [1, 37] introduce object queries and decode them with transformers. These queries are optimized by bipartite matching with ground truths. Due to the different optimization processes in different detection paradigms, a large semantic gap exists between heterogeneous detectors, reducing the benefits brought by pixel-to-pixel feature mimicking. Our UniKD learns how to absorb the detection-relevant knowledge from the teacher and encode it into fixed-length queries so that the extracted features are detector-agnostic.

2.2. Knowledge Distillation for Detection

As object detection is a fine-grained visual understanding task, only mimicking the response of the teacher, as in the vanilla KD, is insufficient. An intuitive way of knowledge distillation is to learn the intermediate features map of the teacher[2, 11, 22, 5], which is commonly used for object detection. Some works try to assign different weights for different parts of the feature map to improve distillation performance. FGFI [24] attaches more importance to regions near objects, while DeFeat [4] decouples the foreground and background parts and distills them separately. ICD [7] determines the weight in an instance-conditional manner. Moreover, other researchers propose different types of knowledge to transfer, such as attention maps[33] and boundary distribution[35].

Beyond homogeneous detectors, some recent works explore the possibility of distillation between heterogeneous pairs. MimicDet [18] proposes to let a one-stage detection head imitates the feature learned by an R-CNN head in a teacher-free manner. HEAD [23] extends this idea and introduces an assistant head to reduce the semantic gap. G-DetKD [32] performs soft matching between features with different resolutions called SGFI. However, these feature-based methods rely on case-by-case algorithm adjustment to adapt to different teacher-student pairs and require huge storage consumption in offline KD. In this paper, we propose a unified query-based distillation paradigm, called UniKD, that achieves plug-and-play in any homogeneous or heterogeneous teacher-student pair by absorbing and distilling detection-relevant information.

3. Method

In this section, we briefly summarize the pipeline of the conventional feature-based methods. Then we introduce the pipeline of our proposed UniKD, including the architecture of the AKE modules and the details in the two stages of UniKD to improve the performance of lightweight detectors. Finally, we discuss the difference between the query-based UniKD and other feature-based KD algorithms and demonstrate the advantages of our method.

3.1. Review of feature-based Detection KD

Recent works for the detection KD focus on feature-based methods that distill intermediate features to preserve spatial information. These feature-mimicking methods are conducted in a spatially pixel-to-pixel matching manner, which can be generally formulated as:

$$\mathcal{L} = \mathcal{D}(F^T, \phi(F^S)), \tag{1}$$

where F^T and F^S are the latent feature representation generated by the teacher and the student, respectively. $\mathcal{D}(\cdot)$ is a distillation loss, *e.g.*, MSE loss. ϕ is an adaptation layer to align the feature dimension between student and teacher. This learning manner of pixel-to-pixel alignment between student and teacher can promote knowledge transfer in homogeneous teacher-student pairs since they have the same detection head and label assignment method. However, this manner has the following problems: (1) direct feature

mimicking fails to distill the knowledge from the heterogeneous teacher due to the serious semantic gap; (2) when the teacher model is too large for real-time inference during distillation, using offline KD which dumps all the feature maps will result in unacceptably huge storage consumption.

3.2. Universal Knowledge Distillation

We propose a general method called Universal Knowledge Distillation (UniKD) to transfer knowledge in any homogeneous or heterogeneous teacher-student pair. The main idea is that we propose AKE modules to first absorb the general knowledge from the feature maps of the teacher network into a fixed-number set of embeddings. Then the same AKEs are applied to the student network and the distillation loss minimizes the distance between the two sets. The AKE modules are implemented by transformer decoder heads attaching to the output multi-scale features of Feature Pyramid Networks (FPN) [12]. As shown in Fig. 2, UniKD can be seen as a two-stage learning paradigm: (1) **absorbing knowledge from teacher**; (2) **injecting knowledge to student**.

In the first stage, as shown in Fig. 2(a), AKEs are pretrained only using the teacher features to absorb the teacher's knowledge into the knowledge embeddings. Specifically, given an image I, we extract the output of FPN in the frozen teacher as F^T. And then send it into content AKE (AKE_{ct}) and positional AKE (AKE_{pos}) respectively. Both AKEs firstly generate queries q^T, and then perform deformable cross-attention between q^T and F^T to obtain knowledge embeddings z^T. The final z^T_{ct} and z^T_{pos} are supervised by the classification and localization loss to ensure that detection-relevant knowledge can be extracted.

In the second stage illustrated in Fig. 2(b), F^S and F^T will be sent into the fixed pre-trained AKEs to obtain the outputs $\{z^S_{ct}, z^S_{pos}, z^T_{pos}, z^T_{pos}\}$. Injecting knowledge to the student is implemented by reducing the difference between z^T and z^S, which will be described in detail in Sec. 3.2.3.

3.2.1 Architecture of AKE module

AKE is the core module in UniKD, which converts the feature maps into a fixed set of embeddings by using a deformable transformer decoder. The overall architecture of AKE is shown in Fig. 3, which consists of two components: query generator (f_{ct} or f_{pos}) and detection-relevant knowledge extractor (f_e). The final distillation loss of UniKD is reducing the difference between the knowledge embeddings z of teacher and student, rather than direct feature maps. Note that there are two types of AKE modules, AKE_{ct} to extract content knowledge embedding z_{ct} and AKE_{pos} to extract positional knowledge embedding z_{pos}. They share the same structure, but the input queries are different, and the parameters are not shared. The pipeline of AKEs can be

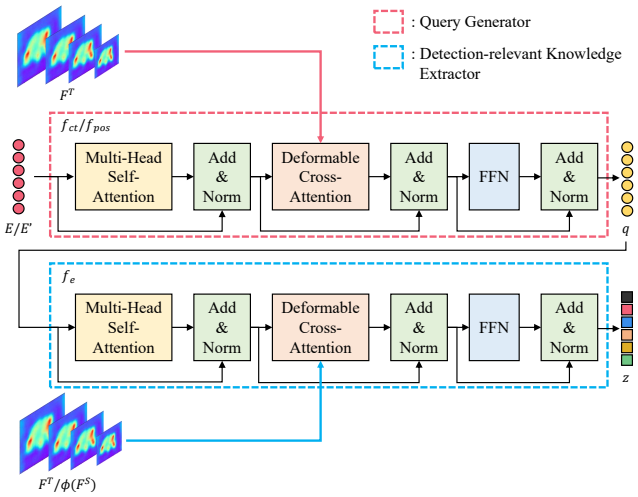


Figure 3: The detailed architecture of AKE modules. The parameter weights of f_{ct}, f_{pos} and their corresponding f_e are not shared.

formulated as:

$$\begin{cases} \text{AKE}_{\text{ct}}: \underset{N\times C}{\text{E}} \xrightarrow{f_{ct}(\cdot)} \underset{N\times C}{q_{ct}} \xrightarrow{f_e(\cdot)} \underset{N\times C}{z_{ct}} \\ \text{AKE}_{\text{pos}}: \underset{N\times 4}{\hat{B}} \xrightarrow{FC} \underset{N\times C}{\text{E}'} \xrightarrow{f_{pos}(\cdot)} \underset{N\times C}{q_{pos}} \xrightarrow{f_e(\cdot)} \underset{N\times C}{z_{pos}}, \end{cases} \tag{2}$$

where N signifies the number of queries and C indicates the channel dimension.

Query Generator. The query generator is illustrated in the red box in Fig. 3. We first initialize two kinds of input embeddings, E for content knowledge extraction and E' for positional knowledge extraction. $E \in \mathbb{R}^{N\times C}$ denotes the initialized learnable decoder input. $E' \in \mathbb{R}^{N\times C}$ denotes the latent embedding generated through $\hat{B} \in \mathbb{R}^{N\times 4}$, which indicates the jittered boxes around the annotated ground-truth B and the randomly sampled background boxes. Specifically, given the annotated $B \in \mathbb{R}^{N'\times 4}$ where N' is the box number, we first sample one jittered box for each ground-truth box. Then a batch of background boxes is randomly sampled around the ground truths and ones whose maximum IoU with ground truths is lower than β will be kept. Repeat it until the number of background boxes meets $N - N'$. The generation of $\hat{B}$ is demonstrated in Eq.(3).

$$\hat{B}_i = \begin{cases} \text{jitter}(B_i) \text{ s.t. } \text{IoU}_{(\hat{\text{B}}_\text{i},\text{B}_\text{i})} > \alpha, & i \in \{1,\dots,N'\} \\ \text{Rand s.t. } \max\{\text{IoU}_{(\hat{\text{B}}_\text{i},\text{B})}\} < \beta, & i \in \{N'+1,\dots,N\} \end{cases} \tag{3}$$

As a rule of thumb, α and β are set to be 0.6 and 0.4, respectively. After that, we adopt a single fully connected layer to transform $\hat{B}$ to E', *i.e.*, $E' = \text{FC}(\hat{B})$.

With E or E' as the input, two kinds of queries are then generated: content queries q_{ct} and positional queries q_{pos}. We represent the query generation process as:

$$\begin{aligned} q_{ct} &= f_{ct}(E, F^T) \\ q_{pos} &= f_{pos}(E', F^T), \end{aligned} \tag{4}$$

where f_{ct} and f_{pos} both consist of one transformer decoder layer shown in the red box in Fig. 3. Deformable cross-attention on F^T is introduced here to help the generated queries to be dynamic and dependent on the input images.

Detection-relevant Knowledge Extractor. Given the queries q_{ct} and q_{pos}, we then use them as probes to extract the detection-relevant knowledge by deformable cross-attention with the multi-scale feature F^*. The module that performs this process is called the detection-relevant knowledge extractor, whose architecture is shown in the blue box in Fig. 3. We represent the operations of the knowledge extraction in AKE modules as follows:

$$\begin{aligned} z_{ct} &= f_e(q_{ct}, F^*) \\ z_{pos} &= f_e(q_{pos}, F^*), \end{aligned} \tag{5}$$

where F^* can be F^T or $\phi(F^S)$. The two f_e in Eq.(5) are both conducted by the stacked transformer decoder layers, but their parameters are not shared. The total parameter in AKE is only 1.56M, which is negligible compared to the backbone and FPN. Note that the AKE modules are discarded in the inference stage.

3.2.2 Absorbing knowledge from teacher

As shown in Fig. 2(a), in the first stage, we only use the teacher model to pre-train the AKE modules. This pre-training process promotes knowledge embeddings z to learn what knowledge is critical for detecting an instance, which we call absorbing knowledge from the teacher.

For the content knowledge, we first obtain the content knowledge embedding z^T_{ct} via Eq.(5) with the input F^T. By attaching an extra FFN layer, z^T_{ct} predicts a fixed-length set of N predictions. Due to the undefined matching relation between z_{ct} and ground-truth boxes B, inspired by DETR [1], we adopt a bipartite matching between these two sets. After matching, output embedding z^T_{ct} are divided into positive and negative sets. We use $\mathcal{P}^{ct}$ and $\mathcal{N}^{ct}$ to denote the index of positive and negative z_{ct}, respectively. $\sigma^{ct}(i)$ is defined as the target index of the i-th query. Specifically, if the i-th query belongs to the positive set, $B_{\sigma^{ct}(i)}$ is its ground truth and $y_{\sigma^{ct}(i)}$ is the corresponding label. Otherwise, it has no target box and $y_{\sigma^{ct}(i)}$ indicates a background label (no object). For the positional knowledge, we first obtain the positional knowledge embedding z^T_{ct}, and the relation between $\hat{B}$ and B is already known. Thus, following the same definition of $\mathcal{P}$, $\mathcal{N}$ and $\sigma(i)$, we have $\mathcal{P}^{pos} = \{1, \ldots, N'\}$, $\mathcal{N}^{pos} = \{N'+1, \ldots, N\}$ and $\sigma^{pos}(i) = i$. The loss function in the first stage can be formulated as:

Methods	Homo.	Hetero.
FitNet, FKD, FGD, *.etc*	✓	x
HEAD, *.etc*	x	✓
G-DetKD, our UniKD, *.etc*	✓	✓

Table 1: Comparison with other KD paradigms.

$$\begin{aligned} \mathcal{L}^T_* =& \frac{1}{N'} \sum_{i \in \mathcal{P}^*} \mathcal{L}_{box}(\mathrm{FFN}(z^T_{*_i}), B_{\sigma^*(i)}) + \\ & \frac{1}{N} \sum_{i \in \mathcal{P}^* \cup \mathcal{N}^*} \mathcal{L}_{cls}(\mathrm{FFN}(z^T_{*_i}), y_{\sigma^*(i)}), \end{aligned} \tag{6}$$

where $*$ can be ct or pos.

3.2.3 Injecting knowledge to student

In the second stage, illustrated in Fig. 2(b), we freeze all the parameters in the AKEs which have been optimized in the stage of absorbing knowledge from teacher. Note that the AKEs for the teacher and the student share the same query generator with F^T as input, and get the same q_{ct} and q_{pos} as shown in Eq.(4). According to Eq.(5), q_{ct} and q_{pos} are directly used as probes to extract knowledge from F^T and F^S to obtain the z^T_{ct}, z^S_{ct}, z^T_{pos} and z^S_{pos} via detection-relevant knowledge extractor. The distillation loss is calculated by summarizing losses from both extractors as follows:

$$\begin{aligned} \mathcal{L}_{kd} &= \lambda_1 \mathcal{D}(\mathrm{AKE_{ct}}(F^T, \phi(F^S)), \mathrm{AKE_{ct}}(F^T, F^T)) + \\ &\quad \lambda_2 \mathcal{D}(\mathrm{AKE_{pos}}(F^T, \phi(F^S)), \mathrm{AKE_{pos}}(F^T, F^T)) \\ &= \lambda_1 \mathcal{D}(z^S_{ct}, z^T_{ct}) + \lambda_2 \mathcal{D}(z^S_{pos}, z^T_{pos}), \end{aligned} \tag{7}$$

where we define $\mathcal{D}(\cdot)$ as the loss function to measure the distance between the knowledge embeddings z^T and z^S. λ_1 and λ_2 are the corresponding loss weights. $\mathrm{AKE}_*(\cdot,\cdot)$ takes F^T as the first input to generate queries q, which are then adopted in the cross-attention with the second input (F^T or $\phi(F^S)$) for detection-relevant knowledge extraction. Note that $\mathrm{AKE_{ct}}$ and $\mathrm{AKE_{pos}}$ are not shared. We introduce the combination of fully connected layers and deformable self-attention layers as the ϕ to align the feature dimension and enhance the distilling ability of the student. During the computation of $\mathcal{D}(\cdot)$, we decouple the positive and negative predictions as:

$$\begin{aligned} \mathcal{D}(z^S, z^T) =& \frac{1}{N'} \sum_{i \in \mathcal{P}} \mathrm{MSE}(\mathrm{z^S_i}, \mathrm{z^T_i}) + \\ & \frac{1}{N - N'} \sum_{i \in \mathcal{N}} \mathrm{MSE}(\mathrm{z^S_i}, \mathrm{z^T_i}). \end{aligned} \tag{8}$$

In addition to the distillation loss, the object detection losses such as classification loss and localization loss are still adopted which is the same as the baseline.

Student (ResNet-18)	Method	Teachers (ResNet-50)					
		RetinaNet (39.5)	ATSS (39.4)	Faster R-CNN (40.3)	FCOS (38.5)	RepPoints (38.6)	Deformable DETR (44.5)
RetinaNet (31.7)	FitNet	34.6 (+2.9)	31.7 (+0.0)	33.4 (+1.7)	33.4 (+1.7)	32.1 (+0.4)	31.5 (-0.2)
	UniKD	35.3 (+3.6)	34.4 (+2.7)	34.8 (+3.1)	34.7 (+3.0)	34.4 (+2.7)	33.7 (+2.0)
ATSS (34.7)	FitNet	36.8 (+2.1)	34.6 (-0.1)	36.6 (+1.9)	34.9 (+0.2)	34.9 (+0.2)	34.0 (-0.7)
	UniKD	37.8 (+3.1)	37.2 (+2.5)	37.5 (+2.8)	37.5 (+2.8)	37.1 (+2.4)	36.6 (+1.9)
Faster R-CNN (33.5)	FitNet	34.2 (+0.7)	32.7 (-0.8)	35.2 (+1.7)	32.6 (-0.9)	32.4 (-1.1)	32.1 (-1.4)
	UniKD	35.0 (+1.5)	34.5 (+1.0)	35.3 (+1.8)	34.8 (+1.3)	34.1 (+0.6)	34.3 (+0.8)
FCOS (32.3)	FitNet	34.6 (+2.3)	32.8 (+0.5)	34.2 (+1.9)	33.2 (+0.9)	32.5 (+0.2)	31.9 (-0.4)
	UniKD	35.6 (+3.3)	35.2 (+2.9)	35.5 (+3.2)	35.5 (+3.2)	34.9 (+2.6)	34.5 (+2.2)
RepPoints (31.9)	FitNet	35.4 (+3.5)	31.9 (+0.0)	35.0 (+3.1)	32.9 (+1.0)	32.5 (+0.6)	31.8 (-0.1)
	UniKD	35.8 (+3.9)	35.0 (+3.1)	35.7 (+3.8)	35.3 (+3.4)	34.9 (+3.0)	34.7 (+2.8)
Deformable DETR (37.2)	FitNet	38.0 (+0.8)	37.3 (+0.1)	39.2 (+2.0)	37.6 (+0.4)	37.6 (+0.4)	37.2 (+0.0)
	UniKD	39.2 (+2.0)	39.1 (+1.9)	39.9 (+2.7)	38.9 (+1.7)	39.0 (+1.8)	38.8 (+1.6)

Table 2: The universality of our methods across different detectors. Each column corresponds to one of the teacher models and each row represents one type of student.

Methods	Distillation Mode	Training time (1x)	mAP
UniKD	online	$\sim$ 73 h	49.2
FitNet	online	$\sim$ 71 h	46.7
UniKD	offline	$\sim$ 15 h	49.1

Table 3: The training time comparison between online and offline KD, with a giant model EVA [3](1074M parameters, 64.1 mAP) as the teacher and ATSS-Res50 (39.1 mAP) as the student.

Methods	Distillation Mode	Real cons. on disk	mAP
UniKD	offline	168.45 GB	49.1
FitNet (full)	offline	3.802 TB	46.7
FitNet (foreground)	offline	1.841 TB	46.8

Table 4: Different methods' offline KD results and their real consumption on disk with the same setting as Tab. 3. We compare UniKD with the full feature distillation and foreground feature distillation via FitNet.

3.3. Discussion

We first compare UniKD with previous KD paradigms on compatibility with homogeneous and heterogeneous teacher-student pairs. As shown in Tab. 1, existing methods are designed only for one type of pair, or need substantial efforts to transfer between different detectors. For instance, G-DetKD designed for RetinaNet to DETR is completely different from that for RetinaNet to FCOS. On the contrary, our UniKD with an absorbing-and-injecting paradigm makes it possible to extract general knowledge from the teacher regardless of its architecture, and directly mimicking the knowledge that integrates all the feature layers alleviates the semantic gap of the pixel-to-pixel matching manner. These merits help UniKD to achieve universality with zero adaptation cost across various teacher-student pairs.

Furthermore, the online inference of the teacher model during knowledge distillation is costly or even impracticable when huge teacher models are adopted. As shown in Tab. 3, if using GPUs with relatively small memory (V100-16G), online distillation with giant teacher results in out-of-memory. Therefore, we use 8 Tesla V100 (32G) GPUs in our experiment and offline KD to reduce the overall training time by $4.9\times$. In this case, offline KD is more applicable, *i.e.*, the teacher's features are first dumped on disk and then loaded without real-time inference during distillation. For offline KD in Tab. 4, previous feature-based methods require a huge amount of storage due to the need to preserve intermediate features from all the FPN levels. Even distilling the foreground feature alone results in unacceptable storage consumption. Compared with storing the feature maps, our query-based UniKD only needs to dump a fixed number of embeddings, greatly reducing storage consumption by $23.1\times$ and $11.2\times$. In addition to the significant improvement of storage efficiency, UniKD also achieves better performance, making it more applicable to operate offline KD.

4. Experiments

4.1. Datasets and Metrics

To verify the effectiveness of UniKD, we conduct extensive experiments on the challenging MS-COCO 2017 dataset [14]. The COCO dataset contains 80 object classes with 118k and 5k images for training and testing, respectively. The performance is evaluated by the mean Average Precision (mAP) metric across the IoU threshold from 0.5 to 0.95 over all classes.

Implementation Details. For the distillation process, we train the student models with a batch size of 16 for 12 epochs, known as the $1\times$ schedule. We use single-scale training by default, with the shorter side of the input im-

Student (RetinaNet)	Method	Teachers (Mask R-CNN)			
		ResNeXt-101 (44.5)	ConvNeXt-S (51.8)	Swin-S (48.2)	Uniformer-B (50.3)
ResNet-18 (31.7)	FitNet	33.8 (+2.1)	34.7 (+3.0)	32.2 (+0.5)	32.3 (+0.6)
	UniKD	35.2 (+3.5)	35.3 (+3.6)	34.4 (+2.7)	34.4 (+2.7)
ConvNeXt-T (43.0)	FitNet	40.1 (-2.9)	42.4 (-0.6)	41.3 (-1.7)	41.4 (-1.6)
	UniKD	43.4 (+0.4)	44.0 (+1.0)	44.1 (+1.1)	44.2 (+1.2)
Swin-T (41.4)	FitNet	42.0 (+0.6)	43.3 (+1.9)	43.2 (+1.8)	43.2 (+1.8)
	UniKD	43.0 (+1.6)	43.9 (+2.5)	44.2 (+2.8)	43.8 (+2.4)
Uniformer-S (44.3)	FitNet	43.7 (-0.6)	45.2 (+0.9)	45.6 (+1.3)	46.4 (+2.1)
	UniKD	45.1 (+0.8)	46.1 (+1.8)	46.5 (+2.2)	46.9 (+2.6)

Table 5: The universality of our methods across different backbone architectures. The students are RetinaNet, and the teachers are Mask R-CNN with different backbones, except for the teacher with ConvNeXt which is Cascade Mask R-CNN.

age scaled to 800 and the longer side limited up to 1333 pixels. We use the optimizer SGD with a momentum of 0.9 and a weight decay of 0.0001. The initial learning rate is 0.01 for one-stage detectors and 0.02 for two-stage detectors, and decays by a factor of 10 at the 8th and 11th epochs. For Deformable DETR [37] and models with backbones other than ResNet, we follow the recommended setting on the learning rate, optimizer, and data augmentation. For distillation, we use two deformable self-attention layers in the adaptor and set the loss weight of content queries and position-dependent queries by 10, and the number of two types of queries is 200 by default.

To pretrain AKE modules on the intermediate feature of teachers, we follow the settings as Deformable DETR. We train the decoders for 15 epochs by default, yet the training time is much less than fully training the teacher as the backbone and FPN are all fixed without propagating gradients. The number of transformer layers in the proposed f_{ct}, f_{pos} and f_e is set to 1. For Deformable DETR, the features after the encoder are used as the input to AKE modules.

4.2. Recent SOTAs in heterogeneous KD.

Recent SOTAs only consider homogeneous KD cases originally, but some of them can be applied to heterogeneous pairs after adjustment, *e.g.* FGD and MGD. Although applicable for hetero-KD, we show in Tab. 7 that SOTA methods for homogeneous KD perform badly in heterogeneous pairs due to the great semantic gap, leading to $-0.4 \sim 1.3$ mAP change. In contrast, UniKD performs uniformly well with $3.0 \sim 3.1$ mAP improvement.

4.3. The universality of UniKD.

Universality across detectors. We first demonstrate the universality of UniKD across different detectors. We choose six representative detectors, including RetinaNet [13], Faster R-CNN [19], FCOS [21], ATSS [34], RepPoints [30], and Deformable DETR [37], to construct homogeneous and heterogeneous teacher-student pairs. The same backbone of ResNet 50 and ResNet 18 is used by the teachers and students respectively. We compare our UniKD with FitNet [2], a representative of intermediate feature mimicking in a pixel-to-pixel manner. Note that in the results posted in this section, we modify the setting of FitNet, using a deformable self-attention layer as the adaptation ϕ and only distilling on the foreground area. These modifications improve the performance of FitNet and make it even comparable to the SOTA feature-based methods.

As shown in Tab. 2, our method consistently boosts the performance of all the student-teacher pairs, surpassing the counterpart in all cases. On the contrary, FitNet leads to unstable gains. For example, when we let RetinaNet learn from homogeneous RetinaNet, the FitNet improves the student by 2.9 mAP, which is still smaller than the 3.6 mAP of UniKD. When the teacher is Faster R-CNN, the gain of FitNet shrinks to 1.7 mAP due to the significant semantic gap. However, UniKD can still extract useful knowledge from the two-stage teacher and improve the student by 3.1 mAP, surpassing the results of FitNet using the homogeneous teacher. For the more challenging teacher of Deformable DETR, FitNet leads to a negative effect on the performance, yet UniKD enhances the performance of the student by 2.0 mAP. Moreover, we show that end-to-end detectors can also benefit from UniKD. By learning from Faster-R-CNN, the performance of the Deformable DETR student can be improved from 37.2 to 39.9 mAP. The superiority of our method can be verified by comparing other results in Tab. 2 which is omitted here.

Universality across backbone architectures. Here, we show UniKD is effective regardless of the backbone architectures. We use RetinaNet as the student and Mask R-CNN or Cascade Mask R-CNN as the teacher. Four types of backbones are used, including ResNeXt [27], modern variant of ResNet (ConvNeXt) [17], vision transformer (Swin-Transformer) [16] and hybrid architecture (Uniformer) [10]. As shown in Tab. 5, directly distilling features leads to undesirable results, which are inferior to the baseline in some cases such as ConvNeXt mimicking ResNeXt. This further verifies that different backbone architectures may result in

Content	Position	mAP
x	x	31.7
✓	x	34.9
x	✓	34.5
✓	✓	**35.3**

(a) Two types of queries.

UniKD	FitNet	mAP
x	x	31.7
x	✓	34.6
✓	x	**35.3**
✓	✓	34.9

(b) Compatibility between UniKD and FitNet.

Num. of queries	mAP
Baseline	31.7
100	35.1
200	**35.3**
300	35.2

(c) The number of queries.

Num. of layers	mAP
Baseline	31.7
1	**35.3**
2	35.2
3	34.9

(d) The number of cross-attention layers in f_e.

λ_1	λ_2	mAP
5	5	34.7
5	10	34.8
10	10	**35.3**
10	20	35.3
20	20	35.2

(e) Loss weights of queries.

Num. of epochs	Time (%)	mAP	
		Teacher	Student
5	7%	22.8	35.2
10	14%	26.1	35.2
15	21%	27.6	**35.3**
20	28%	28.4	**35.3**

(f) The number of epochs to pretrain AKE.

Num. of layers	FitNet	UniKD
0	31.8	31.9
1	31.9	33.4
2	31.7	**33.8**
3	31.3	**33.8**
4	31.6	**33.8**

(g) The number of self-attention layers in ϕ.

Table 6: Ablation studies on Retina ResNet-50 teacher and Retina ResNet-18 student except (g) with Deformable DETR.

Student	Method	mAP	mAP_{50}	mAP_{75}
RetinaNet R18 (31.7)	FGD	31.3	48.6	33.2
	MGD	32.5	49.9	35.2
	UniKD	**34.8**	**53.3**	**37.0**
FCOS R18 (32.5)	FGD	33.8	52.1	35.7
	MGD	33.4	50.6	35.9
	UniKD	**35.5**	**54.4**	**37.6**

Table 7: The performance of conventional feature-based distillation methods for homogeneous pairs.

the semantic gap between features, hurting the distillation process. In UniKD, universal knowledge can be extracted and distilled to enhance the student consistently. For example, learning from ResNeXt-101 can improve Uniformer-S by 0.8 mAP which is even better than the teacher.

4.4. Ablation Study

In this subsection, we conduct extensive experiments to analyze the components and hyper-parameters in UniKD.

The effectiveness of two types of queries. We first explore the effectiveness of two proposed types of queries, *i.e.*, content queries and positional queries. The results are shown in Tab. 6a. We can find that each of them can significantly boost the performance of the student model independently, and using both of them achieves the best. This shows that content queries and position-dependent queries capture different knowledge to some extent and are complementary to each other.

The compatibility between UniKD and FitNet. We try to adopt UniKD and FitNet simultaneously in Tab. 6b. It shows that using them together performs worse than using UniKD individually, but it is still better than FitNet. This indicates that the feature-mimicking mechanism is already contained in UniKD and using UniKD alone is enough.

The number of queries. The number of queries can limit the amount of knowledge extracted from the teacher, and we ablate the number of queries to examine its influence on the distillation performance, shown in Tab. 6c. We find that more queries are generally better, while the number of queries over 200 might decrease slightly. So we use 200 queries in all other experiments.

The number of cross-attention layers in f_e. The cross-attention layers in f_e transform the intermediate features from the teacher and student into a more detection-relevant feature space. However, more layers mean more complexity and may harm the distillation performance. As shown in Tab. 6d, using only one layer is the best, and can significantly reduce the computation complexity.

The loss weights of two types of queries. The loss weight λ_1 and λ_2, are chosen to achieve comparable loss values as detection losses, and we ablate the choices of two weights in Tab. 6e. The results show that UniKD is insensitive to λ_1 and λ_2, and setting both of them to 10 performs the best. All the other experiments in this paper use the same value of λ_1 and λ_2, demonstrating the robustness of our method.

The epochs of pretraining AKE. The cross-attention layers in AKE are trained by detection losses, and more training epochs lead to better performance of AKE itself. We report the distillation results and the pretraining time in percentage compared with the full training time of the teacher in Tab. 6f. Pretraining the AKE for 15 epochs is enough to achieve the best performance, which only accounts for 21% of the teacher training time with the ResNet 50 backbone. This is because the backbone and FPN are fixed during pretraining, thus the percentage will decrease further as the teacher network becomes larger. For example, the percentage becomes 8% when using the Uniformer-B backbone.

The number of self-attention layers in the adaptor. The self-attention layer in the adaptor helps to reduce the gap

between the teacher and the student. To show its effects, we replace the teacher with Deformable DETR where the semantic gap is significant. As shown in Tab. 6g, two self-attention layers boost the performance by 1.9% mAP combined with UniKD, while more layers do not bring further improvement. Also, we adopt the same adaptors to FitNet. Tabel 6g shows that it even hurts the performance, proving the effectiveness of UniKD is essential to the improvement.

The performance of modified FitNet. As mentioned in section 4.3, we modify the conventional FitNet method for better performance. The detailed modifications include: **(a)** distill on all the FPN levels, **(b)** we use a deformable self-attention layer as the adaptation layer ϕ, which is used to align the channel number of the student and teacher, **(c)** inspired by FGD, only foreground features are used to compute feature MSE loss. As shown in Tab. 8, the modified version is comparable with SOTA detection KD methods such as FGD (only $0 \sim 0.1$ mAP gap).

Student	Method	mAP	mAP_s	mAP_m	mAP_l
RetinaNet R50 (37.4)	FitNet	39.4	22.9	43.0	50.7
	FitNet (modified)	**40.4**	**23.5**	**44.8**	52.5
	FGD	**40.4**	23.4	44.7	**54.1**
Faster R-CNN R50 (38.4)	FitNet	41.1	22.8	45.6	56.4
	FitNet (modified)	41.9	**26.0**	**46.4**	53.9
	FGD	**42.0**	23.8	**46.4**	**55.5**

Table 8: The performance of original FitNet and our modified version compared with existing KD methods on homogeneous teacher-student pairs.

4.5. Comparison with State-of-the-art Methods

Here we compare our method with state-of-the-art on detection distillation under fair settings. For homogeneous teacher-student pairs, we choose RetinaNet and Cascade Mask R-CNN with ResNext 101 as the teachers to distill RetinaNet and Faster R-CNN with ResNet 50 as the students, respectively. As shown in Tabel 9, UniKD achieves better performance than other methods that are specifically designed for homogeneous detectors. As to the heterogeneous scenario, we let RetinaNet and FCOS with ResNet 18 mimic RepPoints with ResNet 50, shown in Tab. 10. We can find that UniKD surpasses the state-of-the-art by a significant margin, demonstrating the superiority of our method.

4.6. Visualization and Analysis

To analyze the knowledge extracted from the teacher model, we show the location and weight of the attention for two types of queries in Fig. 4. We can see that the queries learn to gather information from the salient and marginal parts of objects to fulfill the detection task, and the two types of queries are complementary to each other. This further verifies our initial motivation, i.e., mining and transferring universal knowledge related to detection itself.

Student	Method	mAP	mAP_s	mAP_m	mAP_l
RetinaNet R50 (37.4)	FKD	39.6	22.7	43.3	52.5
	FGD	40.4	23.4	**44.7**	54.1
	UniKD	**40.7**	**23.4**	44.6	**54.3**
Faster R-CNN R50 (38.4)	FKD	41.5	23.5	45.0	55.3
	FGD	42.0	23.8	**46.4**	55.5
	UniKD	**42.3**	**24.7**	46.0	**55.8**

Table 9: Comparison with existing KD methods on homogeneous teacher-student pairs.

Student	Method	mAP	mAP_{50}	mAP_{75}
RetinaNet R18 (31.7)	SGFI	31.6	49.5	33.2
	HEAD	34.2	52.4	36.6
	UniKD	**34.8**	**53.3**	**37.0**
FCOS R18 (32.5)	SGFI	32.6	50.9	34.4
	HEAD	35.0	53.8	36.8
	UniKD	**35.5**	**54.4**	**37.6**

Table 10: Comparison with existing KD methods on heterogeneous teacher-student pairs.

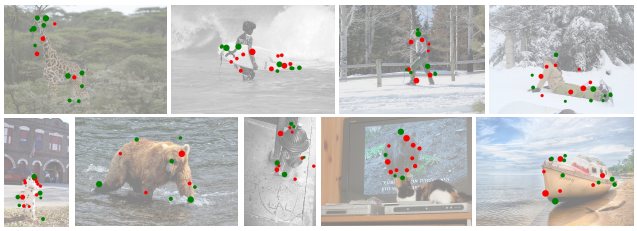


Figure 4: The visualization of learned offsets in the deformable cross-attention in f_e. The red dots indicates q_{ct} and green dots indicates q_{pos}.

5. Conclusion

In this paper, we proposed Universal Knowledge Distillation (UniKD) for training high-performance lightweight object detectors. It can be directly applied to homogeneous or heterogeneous teacher-student pairs without complex adjustments and is storage-efficient for offline KD. UniKD uses content and positional queries to extract detection-relevant knowledge and transfers it to arbitrary students. Extensive experiments on various detector pairs and model architectures demonstrate the effectiveness and universality of UniKD. In addition, we also observe that learning from stronger teachers with different architectures does not lead to more improvement, and we leave this issue in future works.

Acknowledgement

This work was partly supported by the National Natural Science Foundation of China (Grant No. U1903213) and the Shenzhen Science and Technology Program (JSGG20220831093004008).

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229, 2020. 3, 5

[2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017. 3, 7

[3] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. *arXiv preprint arXiv:2211.07636*, 2022. 6

[4] Jianyuan Guo, Kai Han, Yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2154–2164, 2021. 3

[5] Kunyang Han, Yong Liu, Jun Hao Liew, Henghui Ding, Yunchao Wei, Jiajun Liu, Yitong Wang, Yansong Tang, Yujiu Yang, Jiashi Feng, et al. Global knowledge calibration for fast open-vocabulary segmentation. *arXiv preprint arXiv:2303.09181*, 2023. 3

[6] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1

[7] Zijian Kang, Peizhen Zhang, Xiangyu Zhang, Jian Sun, and Nanning Zheng. Instance-conditional knowledge distillation for object detection. *Advances in Neural Information Processing Systems*, 34:16468–16480, 2021. 3

[8] Shanshan Lao, Yuan Gong, Shuwei Shi, Sidi Yang, Tianhe Wu, Jiahao Wang, Weihao Xia, and Yujiu Yang. Attentions help cnns see better: Attention-based hybrid image quality assessment network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1140–1149, 2022. 1

[9] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, pages 734–750, 2018. 3

[10] Kunchang Li, Yali Wang, Peng Gao, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. Uniformer: Unified transformer for efficient spatiotemporal representation learning. *arXiv preprint arXiv:2201.04676*, 2022. 7

[11] Quanquan Li, Shengying Jin, and Junjie Yan. Mimicking very efficient network for object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 6356–6364, 2017. 3

[12] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4

[13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 2, 7

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755, 2014. 6

[15] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2

[16] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 7

[17] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 7

[18] Xin Lu, Quanquan Li, Buyu Li, and Junjie Yan. Mimicdet: Bridging the gap between one-stage and two-stage object detection. In *European Conference on Computer Vision*, pages 541–557, 2020. 2, 3

[19] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 1, 2, 7

[20] 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. 1

[21] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 1, 3, 7

[22] Jiahao Wang, Mingdeng Cao, Shuwei Shi, Baoyuan Wu, and Yujiu Yang. Attention probe: Vision transformer distillation in the wild. In *ICASSP*, pages 2220–2224, 2022. 3

[23] Luting Wang, Xiaojie Li, Yue Liao, Zeren Jiang, Jianlong Wu, Fei Wang, Chen Qian, and Si Liu. Head: Hetero-assists distillation for heterogeneous object detectors. *arXiv preprint arXiv:2207.05345*, 2022. 2, 3

[24] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4933–4942, 2019. 3

[25] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 1

[26] Taiqiang Wu, Cheng Hou, Zhe Zhao, Shanshan Lao, Jiayi Li, Ngai Wong, and Yujiu Yang. Weight-inherited distillation for task-agnostic bert compression, 2023. 1

[27] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on*

computer vision and pattern recognition, pages 1492–1500, 2017. 7

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

[29] Zhendong Yang, Zhe Li, Ailing Zeng, Zexian Li, Chun Yuan, and Yu Li. Vitkd: Practical guidelines for vit feature knowledge distillation. *arXiv preprint arXiv:2209.02432*, 2022. 1

[30] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. Reppoints: Point set representation for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9657–9666, 2019. 3, 7

[31] Zhendong Yang, Ailing Zeng, Zhe Li, Tianke Zhang, Chun Yuan, and Yu Li. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. *arXiv preprint arXiv:2303.13005*, 2023. 1

[32] Lewei Yao, Renjie Pi, Hang Xu, Wei Zhang, Zhenguo Li, and Tong Zhang. G-detkd: Towards general distillation framework for object detectors via contrastive and semantic-guided feature imitation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3591–3600, 2021. 3

[33] Linfeng Zhang and Kaisheng Ma. Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors. In *International Conference on Learning Representations*, 2020. 3

[34] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020. 7

[35] Zhaohui Zheng, Rongguang Ye, Ping Wang, Dongwei Ren, Wangmeng Zuo, Qibin Hou, and Ming-Ming Cheng. Localization distillation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9407–9416, 2022. 3

[36] Xingyi Zhou, Jiacheng Zhuo, and Philipp Krahenbuhl. Bottom-up object detection by grouping extreme and center points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 850–859, 2019. 3

[37] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 1, 3, 7