LGD: Label-guided Self-distillation for Object Detection

Peizhen Zhang,[*1] **Zijian Kang,**[*2] **Tong Yang,**[1] **Xiangyu Zhang,**[†1]
Nanning Zheng,[2] **Jian Sun** [1]

[1]MEGVII Technology,
[2]Xi'an Jiaotong University
{zhangpeizhen, yangtong, zhangxiangyu, sunjian}@megvii.com,
kzj123@stu.xjtu.edu.cn, nnzheng@mail.xjtu.edu.cn

Abstract

In this paper, we propose the first self-distillation framework for general object detection, termed LGD (**L**abel-**G**uided self-**D**istillation). Previous studies rely on a strong pretrained teacher to provide instructive knowledge that could be unavailable in real-world scenarios. Instead, we generate an instructive knowledge by inter-and-intra relation modeling among objects, requiring only student representations and regular labels. Concretely, our framework involves sparse label-appearance encoding, inter-object relation adaptation and intra-object knowledge mapping to obtain the instructive knowledge. They jointly form an implicit teacher at training phase, dynamically dependent on labels and evolving student representations. Modules in LGD are trained end-to-end with student detector and are discarded in inference. Experimentally, LGD obtains decent results on various detectors, datasets, and extensive tasks like instance segmentation. For example in MS-COCO dataset, LGD improves RetinaNet with ResNet-50 under 2× single-scale training from 36.2% to 39.0% mAP (+ **2.8**%). It boosts much stronger detectors like FCOS with ResNeXt-101 DCN v2 under 2× multi-scale training from 46.1% to 47.9% (+ **1.8**%). Compared with a classical teacher-based method FGFI, LGD not only performs better without requiring pretrained teacher but also reduces **51**% training cost beyond inherent student learning.

1 Introduction

Knowledge distillation (KD) (Romero et al. 2015; Hinton, Vinyals, and Dean 2015) is initially proposed for image classification and obtains impressive results. Typically, it is about transferring instructive knowledge from a pretrained model (teacher) to a smaller one (student). Recently, KD applied to the fundamental object detection task, has aroused researchers' interests (Li, Jin, and Yan 2017; Wei et al. 2018; Wang et al. 2019; Zhang et al. 2020; Dai et al. 2021; Guo et al. 2021; Zhang and Ma 2021; Yao et al. 2021). Existing works achieve respectable performance but the choice of teacher is sophisticated and inconsistent among them. One common ground is that they all require a heavy pretrained teacher as it is discovered by recent works (Zhang and Ma 2021; Yao et al. 2021) that distillation efficacy could be enhanced with stronger teachers. Yet the pursuit for an ideal teacher could scarcely be satisfied in real-world applications, since it might take tons of efforts on trial and error (Peng et al. 2020). Instead, the issue that "***KD for generic detection without pretrained teacher***" is barely investigated.

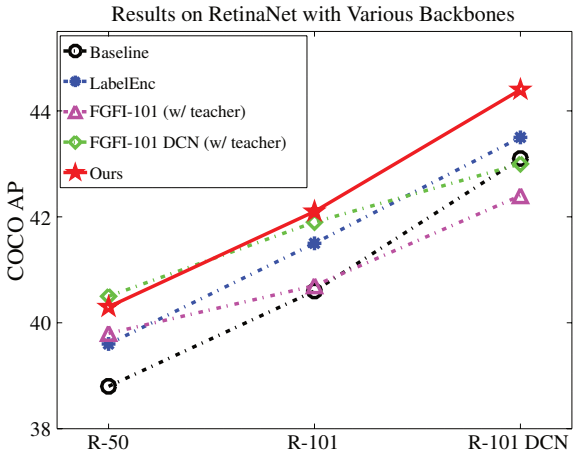

Figure 1: Results trending on RetinaNet 2× ms with backbones R-{50, 101, 101 DCN} respectively. FGFI-{101, 101 DCN} denote FGFI method using RetinaNet 2× ms with R-101 and R-101 DCN as teachers, respectively.

To alleviate the pretrained teacher dependence, teacher-free schemes are proposed like (a) *self-distillation*, (b) *collaborative learning* and (c) *label regularization*, where instructive knowledge could be cross-layer features (Zhang et al. 2019), competitive counterparts (Zhang et al. 2018) and modulated label distribution (Yuan et al. 2020), etc. However, these methods are designed for classification and are inapplicable to detection since the latter has to handle multiple objects with different locations and categories but singe image classification. Lately, LabelEnc (Hao et al. 2020) extends traditional label regularization by introducing location-category modeling with an isolated network. It produces label representations with which the student features are supervised. Though it obtains impressive results, we find the improvement saturates (Figure 3) as detector grows stronger, *e.g.*, with larger backbones and multi-scale training. We conjecture this is because labels themselves describe

[*]These authors contributed equally.
[†]Corresponding author.

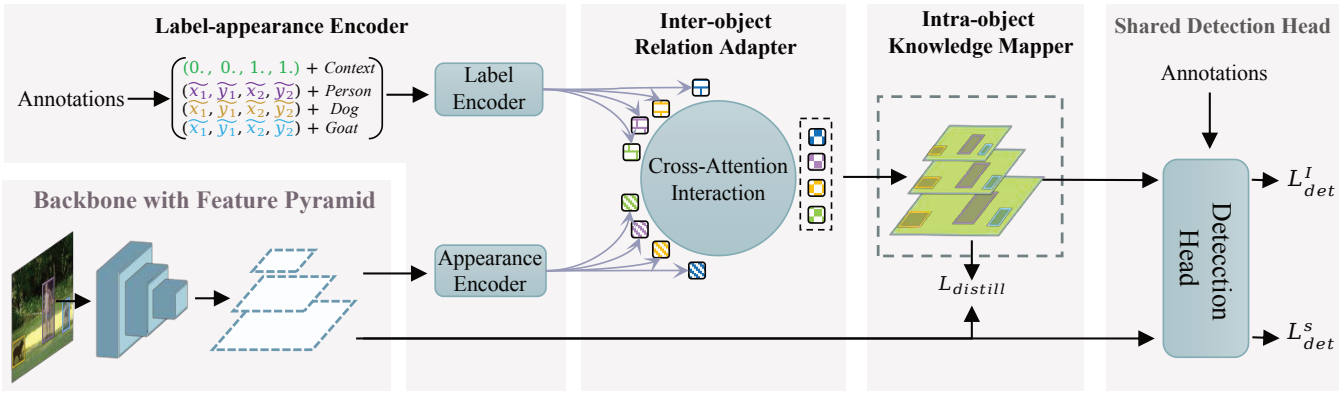

Figure 2: The proposed framework contains three modules: (1) Label-appearance encoder, (2) Inter-object relation adapter and (3) Intra-object knowledge mapper. For brevity, we omit the pyramid level indications which will be elaborated in Section 3. L^I_{det} / L^S_{det} denote detection losses upon instructive / student representations and $L_{distill}$ is the distillation loss. We denote by $(\widetilde{x}_1, \widetilde{y}_1, \widetilde{x}_2, \widetilde{y}_2)$ the ground-truth box location normalized by image size that $(0., 0., 1., 1.)$ refers to an entire context box.

only object-wise categories and locations, without considering the inter-object relationship which is also important (Hu et al. 2018; Cai et al. 2019). For detectors with limited capacity, LabelEnc provides strong complementary supervision, albeit without relation information. For stronger detectors which are able to extract abundant object-wise hints from default supervision, using LabelEnc becomes less beneficial or even detrimental (see the leftmost figure in Figure 3). This might result from semantic discrepancy by heterogeneous input (image *vs.* label) and isolated modeling.

Motivated by this, we propose **L**abel-**G**uided self-**D**istillation (LGD), a new teacher-free method for object detection as shown in Figure 2. In LGD, we devise an inter-object relation adapter and an intra-object knowledge mapper to collaboratively model the relation in forming instructive knowledge. The relation adapter computes interacted embeddings by a cross-attention interaction. Specifically, the interacted embedding of each object is calculated by first measuring the cross-modal similarity between its appearance embedding and every label embedding upon which a weighted-aggregation is then performed. The knowledge mapper maps the interacted embeddings onto feature map space as final instructive knowledge, considering intra-object representation consistency and localization heuristics. Owing to the above relation modeling, the final instructive knowledge is naturally adapted to the student representations, facilitating effective distillation for strong student detectors and semantic discrepancy mitigation. Beyond efficacy, our method is also efficient, it does not rely on a strong convolution network as teacher because we adopt efficient instance-wise embeddings design. The above efficient design allows LGD to train jointly with the student, simplify the pipeline, and reduce training cost (Table 7). During inference, only student detector is kept, bringing no extra cost. In short, our contributions are three-fold:

1. We propose a new self-distillation framework for general object detection. Unlike previous methods that use a convolution network as teacher, LGD generates instructive knowledge on-the-fly without pretrained teacher and improves the detection quality under limited training cost.
2. We introduce inter-and-intra relation to model a new instructive knowledge, rather than simply extract existent relation from student and teacher for distillation.
3. The proposed method outperforms previous teacher-free SOTA with higher upper limit and is better than classical teacher-based method FGFI in strong student settings. Beyond inherent student learning, it saves **51**% training time against the classical teacher-based distillation.

2 Related Work

2.1 Detection KD with Pretrained Teachers

Unlike classification, knowledge transfer for object detection is more challenging. In detection, models are asked to predict multiple instances with diversified categories distributed at different locations in the image. (Li, Jin, and Yan 2017) proposed Mimic to distill activations within the region proposals predicted by RPN (Ren et al. 2015). (Chen et al. 2017) introduced weighted cross-entropy and bounded regression loss for enhancing the performance. To further exploit the context information of the distilling regions around the objects, (Wang et al. 2019) extended the ground-truth box regions by anchor-assigned ones. For learning adapted sampling weight for different knowledge, (Zhang et al. 2020) proposed PAD with uncertainty modeling. Besides intermediate feature hints, (Dai et al. 2021) involved the prediction map distillation obeying the assignment rules and relation distillation (Park et al. 2019) upon their defined general instances. Instead of focusing on foreground regions only, (Guo et al. 2021) decoupled the fore/back-ground knowledge transfer. To facilitate region-agnostic distillation, (Zhang and Ma 2021) proposed feature-based knowledge transfer by spatial-channel-wise attention. To resolve the feature resolution mismatching in cross-layer distillation and mitigate the misaligned label assignment, (Yao et al. 2021) introduced G-DetKD. Above methods mainly conducted feature-based distillation which is followed in this

work. Whereas, they are designed for settings with strong pretrained teachers that could be unavailable or unaffordable in real-world scenarios. Recently, (Huang et al. 2020) proposed self-distillation for weakly supervised detection but the setting is much different from generic object detection.

2.2 Teacher-free Methods

Beyond traditional KD with pretrained teacher, there are teacher-free schemes that could be divided into three categories: (1) self-distillation (2) collaborative learning and (3) label regularization. (1) self-distillation excavates instructive knowledge from model itself. For instance, (Yang et al. 2019; Kim et al. 2020) used previously saved snapshots as teachers. In (Zhang et al. 2019), network was divided into sections that deeper layers were used to teach the shallower ones. In MetaDistiller (Liu et al. 2020), the knowledge stemmed from one-step predictions. (2) Collaborative learning involves multiple students to boost each other. (Zhang et al. 2018) proposed deep mutual learning (DML) where student networks with identical architecture learned collaboratively. (Lan, Zhu, and Gong 2018) proposed ONE by considering ensemble learning in branch-granularity. In KDCL (Guo et al. 2020), predictions were fused together as instructive knowledge. Likewise in (Chen et al. 2020a), ensemble logits of multiple students were aggregated to distill another. (Furlanello et al. 2018) proposed Born-Again Network (BAN) that leveraged information from last generations to distill the next. (3) For label regularization, (Yuan et al. 2020) proposed tf-KD for regularized label distribution beyond label smoothing (Szegedy et al. 2016). However, above methods were designed for classification only.

Recently, there have been newly-built label regularization methods (Mostajabi, Maire, and Shakhnarovich 2018; Hao et al. 2020) using an isolated network to explicitly model labels as features for supervision,*w.r.t.* semantic segmentation and detection. They obtained impressive results. In (Hao et al. 2020), dense color maps with category and location information were constructed and fed into an auto-encoder-like network to fetch label representations. However, they considered each object modeling separately which was suboptimal. Instead, we propose to generate instructive knowledge by inter-object and intra-object relation modeling to form a self-distillation scheme with higher upper limit.

3 Method

As shown in Fig. 2, we illustrate the modules in LGD as follows: (1) An encoder that computes label and appearance embeddings. (2) An inter-object relation adapter that generates interacted embeddings given label and appearance embeddings of objects. (3) An intra-object knowledge mapper that back-projects interacted embeddings onto feature map space to obtain instructive knowledge for distillation.

3.1 Label-appearance Encoder

(1) Label Encoding: For each object, we concatenate its normalized ground-truth box $(\widetilde{x_1}, \widetilde{y_1}, \widetilde{x_2}, \widetilde{y_2})$ and one-hot category vector to obtain a descriptor. The object-wise descriptors are passed into a label encoding module for refined label embeddings $\mathcal{L} = \{\mathbf{l}_i \in \mathbb{R}^C\}_{i=0}^N$, where i indicates object index, $C = 256$ is the intermediate feature dimension, and N is the object number. $i = 0$ indexes the context object. To introduce basic relation modeling among label descriptors and maintain a permutation-invariant property, we adopt the classical PointNet (Qi et al. 2017) as the label encoding module. It processes the descriptors by a multi-layer perceptron (Friedman et al. 2001) with local-global modeling by a spatial transformer network (Jaderberg et al. 2015). Also, the label descriptors are similar to point set that is accustomed to PointNet (bounding boxes could be viewed as points in 4-dimensional Cartesian space). Empirically, using PointNet as encoder behaves slightly better than MLP or transformer encoder (Vaswani et al. 2017) (Table 4). We further replace the BatchNorm (Ioffe and Szegedy 2015) with LayerNorm (Ba, Kiros, and Hinton 2016) to adapt the small-batch detection setting. Notably, the above 1D object-wise label encoding manner is more efficient than that in LabelEnc. The LabelEnc constructs an ad-hoc color map $\in \mathbb{R}^{H\times W\times K}$ to describe labels where (H, W) and K are input resolution and object category number respectively ($HWK \gg C$). The color map is processed by an extra CNN and pyramid network for 2D pixel-wise representations $\mathcal{L}' = \{\mathbf{l}'_i \in \mathbb{R}^{H_p\times W_p\times C}, 1 \le p \le P\}$. P refers to the number of pyramid scales (Lin et al. 2017a) that (H_p, W_p) denotes feature map resolution at scale p.

(2) Appearance Encoding: Beyond label encoding, we retrieve compact appearance embeddings from feature pyramid of student detector that contains appearance feature of perceived objects. We adopt a handy mask pooling to extract object-wise embeddings from the feature maps. Specifically, we pre-compute the object-wise masks: $\mathcal{M} = \{\mathrm{m}_i\}_{i=1}^N \bigcup \{\mathrm{m}_0\}$ at input level for total N objects and a virtual context object with location $(0., 0., 1., 1.)$ covering the entire image. For each object i $(0 \le i \le N)$, $\mathrm{m}_i \in \mathbb{R}^{H\times W}$ is a binary matrix whose values are set as 1 inside the ground-truth region and 0 otherwise. The mask pooling is conducted concurrently for all pyramid levels, at each of which, object-wise masks at input level are down-scaled to corresponding resolution to become scale-specific ones. At p-th scale, the appearance embedding $\mathbf{a}_i \in \mathbb{R}^C$ is obtained by calculating channel-broadcasted Hadamard product between the projected feature map $\mathcal{F}_{proj}(X_p) \in \mathbb{R}^{H_p\times W_p\times C}$ and down-scaled object mask $\in \mathbb{R}^{H_p\times W_p}$, followed by global sum pooling. $\mathcal{F}_{proj}(\cdot)$ is a single 3×3 *conv* layer. Thus, we collect appearance embeddings: $\mathcal{A}_p = \{\mathbf{a}_i \in \mathbb{R}^C\}_{i=0}^N$ for each object at level p (likewise for the other levels).

3.2 Inter-object Relation Adapter

Given label and appearance embeddings, we formulate the inter-object relation adaption by a cross-attention process. In Fig. 2, this process is executed at every student appearance pyramid scale to retrieve the interacted embeddings. We omit the pyramid scale subscript below for brevity.

During the cross attention, a sequence of key and query tokens are leveraged in calculating KQ-attention relation for aggregating value to obtain attention outputs. For achieving

the label-guided information adaption, we exploit the appearance embeddings $\mathcal{A}$ at current scale as query, and the scale-invariant label embeddings $\mathcal{L}$ as key and value. The attention scheme measures the correlation between lower-level structural appearance information and higher-level label semantics among objects then reassembles the informative label embeddings for dynamic adaption.

Before conducting attention, the query, key, and value are transformed by linear layers $f_{\mathcal{Q}}$, $f_{\mathcal{K}}$ and $f_{\mathcal{V}}$, respectively. We then computed the interacted embeddings $\mathbf{u}_i \in \mathbb{R}^C$ for i-th object by weighting each transformed label embedding $f_{\mathcal{V}}(\mathbf{l}_j)$ by label-appearance correlation factor w_{ij}.

$$\mathbf{u}_i = \sum_{j=0}^{N} w_{ij} f_{\mathcal{V}}(\mathbf{l}_j) \tag{1}$$

w_{ij} is calculated by a scaled dot-product between i-th appearance embeddings $\mathbf{a}_i$ and j-th label embeddings $\mathbf{l}_j$ followed by a softmax operation:

$$w_{ij} = \frac{\exp\left(f_{\mathcal{Q}}(\mathbf{a}_i) \cdot f_{\mathcal{K}}(\mathbf{l}_j)/\tau\right)}{\sum_{k=0}^{N} \exp\left(f_{\mathcal{Q}}(\mathbf{a}_i) \cdot f_{\mathcal{K}}(\mathbf{l}_k)/\tau\right)} \tag{2}$$

where $\cdot$ is the notation for inner product and $\tau = \sqrt{C}$ is the denominator for variance rectification (Vaswani et al. 2017).

Specifically, for more robust attention modeling, the paradigm actually involves T set of concurrent operations termed heads to obtain partial interacted embeddings in parallel. By concatenating the partial interacted embeddings from all heads and applying a linear projection $f_{\mathcal{P}}$, we obtain interacted embeddings $\mathbf{E} = \{\mathbf{e}_i \in \mathbb{R}^C\}_{i=0}^{N}$ for all objects:

$$\mathbf{e}_i = f_{\mathcal{P}}([\mathbf{u}_i^1; \mathbf{u}_i^2; \ldots; \mathbf{u}_i^T]) \tag{3}$$

where [;] denotes the concatenation operator that combines the partial embeddings along the channel dimension. The resulting embeddings are also scale-sensitive as the appearance embeddings. As aforementioned, we obtain interacted embeddings across scales by iterating over all feature scales.

Technically, above computation is accomplished by means of multi-head self attention (MHSA) (Vaswani et al. 2017). Note that our framework is decoupled to the specific choice. As will be shown in this paper, LGD shows the efficacy even with the naive transformer. It is likely to perform even better by using advanced variants like focal transformer (Yang et al. 2021) but that is beyond the scope.

3.3 Intra-object Knowledge Mapper

To make the 1D interacted embeddings applicable to widely-used intermediate feature distillation (Li, Jin, and Yan 2017; Wang et al. 2019) for detection, we map the interacted embeddings onto 2D feature map space to fetch instructive knowledge. Naturally, for each pyramid scale $p, (1 \leq p \leq P)$, the resolutions of resulting maps are confined to be identical with corresponding student feature maps.

Intuitively, since spatial topology is not maintained in label encoding for compact representations (Sec. 3.1), it is important to recover the localization information for each object to achieve alignment in geometric perspective. Naturally, object bounding box regions serve as good heuristics. We fill each object-binding interacted embedding within its corresponding ground-truth box region on a zero-initialized feature map. In practice, for each object i, we acquire its feature map at p-th scale by calculating matrix multiplication between the vectorized object mask $\mathbf{m}_i \in \mathbb{R}^{H_p W_p \times 1}$ and the projected, interacted embedding $\mathbf{e}_i$. All these object-wise maps are added up to a unified one followed by a refinement module $\mathcal{F}_{ref}(\cdot)$ to form the instructive knowledge:

$$X_p{}^{\mathcal{I}} = \mathcal{F}_{ref}\left[\mathbf{m}_0 \mathcal{F}_{ctx}^{\top}(\mathbf{e}_0) + \mathcal{G}\left(\sum_{i=1}^{N} \mathbf{m}_i \mathcal{F}_{inst}^{\top}(\mathbf{e}_i)\right)\right] \tag{4}$$

where $\mathcal{F}_{ctx}^{\top}(\mathbf{e}_0)$ and $\mathcal{F}_{inst}^{\top}(\mathbf{e}_i) \in \mathbb{R}^{1\times C}, (1 \leq i \leq N)$ are the transposes of projected context and normal object interacted embeddings, respectively. Both $\mathcal{F}_{ctx}(\cdot)$ and $\mathcal{F}_{inst}(\cdot)$ are single *fc* layers. $\mathcal{G}(\cdot)$ is a single 3×3 *conv* layer. $\mathcal{F}_{ref}(\cdot)$ starts with a *relu* followed by three 3×3 *conv* layers. Thus, we collect the instructive knowledge $\mathcal{X}^{\mathcal{I}} = \{X_p{}^{\mathcal{I}} \in \mathbb{R}^{H_p \times W_p \times C}\}_{p=1}^{P}$ at all scales.

Beyond applicability consideration, the above mapping implies a spirit of *intra-object* regularization (Yun et al. 2020; Law and Deng 2018; Chen et al. 2020b) which enforces activation neurons inside the same foreground region on student appearance representations to be close (through subsequent distillation in Equation 5). Moreover, these instructive representations will be supervised with detection loss for ensuring the representation capability (Equation 6).

Before distillation, an adaption head $\mathcal{F}_{adapt}(\cdot)$ is used to adapt student representations, following FitNet. We conduct knowledge transfer between the instructive representations $X_p{}^{\mathcal{I}}$ and the adapted student features $X_p^{\mathcal{S}} = \mathcal{F}_{adapt}(X_p)$ at each feature scale. We adopt InstanceNorm (Ulyanov, Vedaldi, and Lempitsky 2016) to eliminate the appearance and label style information for both feature maps followed by a Mean-Square-Error (MSE):

$$L_{distill} = \frac{1}{N_{total}} \sum_{p=1}^{P} \left\| X_p^{\mathcal{S}} - X_p^{\mathcal{I}} \right\|^2 \tag{5}$$

where P is the total number of pyramid levels, and $N_{total} = \sum_{p=1}^{P} H_p W_p C$ indicates the total size of the feature pyramid tensors. As gradient stopping technique suggested in previous studies (Hao et al. 2020; Hoffman, Gupta, and Darrell 2016), we detach instructive representations $\mathcal{X}^{\mathcal{I}}$ when calculating distillation loss to avoid model collapse.

Besides the distillation loss and detection loss for optimizing student detector, we further ensure the instructive representation quality and consistency with student representations by sharing the detection head for supervision. The overall detection loss is shown below:

$$L_{det} = L_{det}^{\mathcal{S}}(\mathcal{H}(\mathcal{X}), \mathcal{Y}) + L_{det}^{\mathcal{I}}(\mathcal{H}(\mathcal{X}^{\mathcal{I}}), \mathcal{Y}) \tag{6}$$

where $\mathcal{X}/\mathcal{X}^{\mathcal{I}}$ denote student / instructive representations across scales. $L_{det}^{\mathcal{S}/\mathcal{I}}$ denotes the detection loss (classification and regression) upon them. $\mathcal{H}(\cdot)$ refers to the detection head. $\mathcal{Y}$ stands for the label set (boxes and categories). In summary, the total training objective is:

$$L_{total} = L_{det} + \lambda L_{distill} \tag{7}$$

where λ is a trade-off for distillation term and we simply adopt $\lambda = 1$ throughout all experiments. For stable training, the distillation starts in 30k iterations since it could be detrimental when the instructive knowledge is optimized insufficiently (Hao et al. 2020; Liu et al. 2020). The student detector backbone is frozen in early $10k$ iterations under $1\times$ training schedule and $20k$ for $2\times$ training schedule.

4 Experiment

4.1 Experiments Setup

The proposed framework is built upon Detectron2 (Wu et al. 2019). Experiments are run with batch size 16 on 8 GPUs. Inputs are resized such that shorter sides are no more than 800 pixels. We use SGD optimizer with 0.9 momentum and 10^{-4} weight decay. The multi-head attention in inter-object relation adapter uses $T = 8$ heads following common practice. For brevity, we denote by R-50, R-101 and R-101 DCN for ResNet-50, ResNet-101 and ResNet-101 with deformable convolutions v2 (Zhu et al. 2019). Main experiments are validated on MS-COCO (Lin et al. 2014) dataset that we also testify on others: Pascal VOC (Everingham et al. 2010) and CrowdHuman (Shao et al. 2018).

MS-COCO is a challenging object detection dataset with 80 categories. Mean average precision (AP) is used as the major metric. Following common protocol (He, Girshick, and Dollár 2019), we use the *trainval-115k* and *minival-5k* subsets *w.r.t.* training and evaluation. We denote by $1\times$ the training for 90k iterations where learning rate is divided by 10 at 60k and 80k iterations. By analogy, $2\times$ denotes 180k of iterations with milestones at 120k and 160k. We term the single and multi-scale training by ss and ms for short.

Pascal VOC is a dataset with 20 classes. The union of *trainval-2007* and *trainval-2012* subsets are used for training, leaving *test-2007* for validation. We report mAP and AP50/75 (AP with overlapping threshold 0.5/0.75). Models are trained for $24k$ iterations with milestones at $18k$ and $22k$.

CrowdHuman is the largest crowd pedestrian detection dataset, containing 23 people per image. It includes 15k and 4370 images *w.r.t.* training and validation. The major metric is *average log miss rate over false positives per image* (termed mMR, lower is better). Models are trained for 30 epochs with learning rate decayed at 24^{th} and 27^{th} epoch.

4.2 Comparison to Teacher-free Methods

Detailed Comparison with State-of-the-Art. As shown in Figure 3 and Table 1, we compare our LGD framework with the baseline and previous teacher-free SOTA, *i.e.*, the LabelEnc (Hao et al. 2020) regularization method. We verify the efficacy on MS-COCO on three popular detectors: Faster R-CNN (Ren et al. 2015), RetinaNet (Lin et al. 2017b) and FCOS (Tian et al. 2019). Figure 3 shows the result trending as student detector grows stronger (longer periods: $1\times \rightarrow 2\times$, scale augmentations: $ss \rightarrow ms$ and larger backbones: R-50 $\rightarrow$ R-101 $\rightarrow$ R-101 DCN). Our model compare favorably to or is slightly better than LabelEnc in earlier settings. For RetinaNet or FCOS R-50 at $2\times ss$ setting, the baseline runs into overfitting while our method tackles that and achieves **2.8**% mAP gain. Notably, as the detector setting becomes stronger, the gain of LabelEnc shrinks rapidly while ours still consistently boosts the performance. For Faster R-CNN with R-101 and R-101 DCN, LabelEnc underperforms the baseline (41.4 *vs.* 41.7 and 44.0 *vs.* 44.1). Instead, our method manage to improve and surpasses LabelEnc at around **1**% mAP, verifying higher upper limit. Likewise, for RetinaNet and FCOS with R-101 and R-101 DCN, our method could steadily achieve gains of **1.2**$\sim$**1.5**%. Note that in traditional distillation schemes, it remains unknown to find suitable teacher for such strong students.

Detector	Backbone	Setting	Baseline	LabelEnc	Ours
FRCN	R-50	$1\times\ ss$	37.6	38.1	**38.3**
		$1\times ms$	37.9	38.4	**38.6**
		$2\times\ ss$	38.0	38.9	**39.2**
		$2\times ms$	39.6	39.6	**40.4**
	R-101	$2\times ms$	41.7	41.4	**42.3**
	R-101 DCN	$2\times ms$	44.1	44.0	**44.9**
RetinaNet	R-50	$1\times\ ss$	36.6	37.8	**38.3**
		$1\times ms$	37.4	**38.5**	**38.5**
		$2\times\ ss$	36.2	**39.0**	**39.0**
		$2\times ms$	38.8	39.6	**40.3**
	R-101	$2\times ms$	40.6	41.5	**42.1**
	R-101 DCN	$2\times ms$	43.1	43.5	**44.4**
FCOS	R-50	$1\times\ ss$	38.8	39.6	**39.7**
		$1\times ms$	39.4	40.0	**40.1**
		$2\times\ ss$	38.1	**41.0**	40.9
		$2\times ms$	41.0	41.8	**42.3**
	R-101	$2\times ms$	42.9	43.6	**44.1**
	R-101 DCN	$2\times ms$	44.9	45.6	**46.3**

Table 1: Detailed comparison with previous SOTA.

Method	RetinaNet		FRCN	
	$1\times\ ss$	$1\times\ ms$	$1\times\ ss$	$1\times\ ms$
Baseline	36.6	37.4	37.6	37.9
DML†	37.0	37.4	37.6	37.9
tf-KD†	–	–	37.5	37.8
BAN†, ♠	36.8	38.0	37.6	38.1
Ours	**38.3**	**38.5**	**38.3**	**38.6**

Table 2: Comparison with typical teacher-free methods. † denotes our transfer to detection. ♠ denotes reporting the 3^{rd} generation result in BAN literature which costs $3\times$ longer training schedules far more than regular $1\times$. Also, it is undefined for tf-KD to experiment on RetinaNet with focal loss.

Comparison with Typical Methods. As aforementioned, teacher-free schemes other than LabelEnc are NOT designed for detection. For surplus concern, we transfer and reimplement typical methods like DML, tf-KD and BAN to detection by substituting their logits distillation with intermediate feature distillation in mainstream detection KD literature (except tf-KD). As shown in Table 2, these methods obtain slight improvement or are even harmful (tf-KD). BAN performs the best among them. It obtains 0.6% improvement on RetinaNet $1\times\ ms$ R-50 at a cost of actual $3\times$ training periods. However, it fails to generalize to other settings.

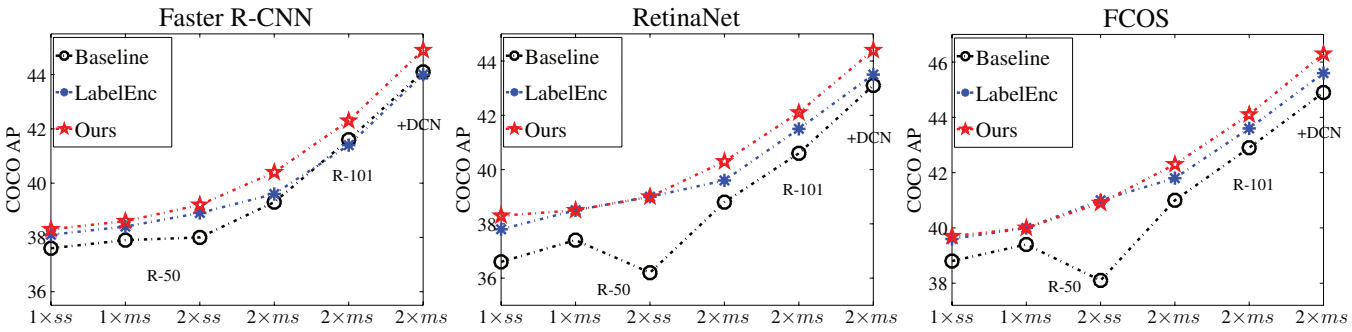

Figure 3: Result tendency as detector grows stronger on three typical detectors by LabelEnc and ours. In each sub-figure, there are six settings from left to right: R-50-{1× ss, 1× ms, 2× ss, 2× ms} → R-101-2× ms → R-101 DCN-2× ms.

4.3 Comparison with Classical Teacher-based KD

Method	Teacher	Student Backbones		
		R-50	R-101	R-101 DCN
Baseline	N/A	38.8	40.6	43.1
LabelEnc	N/A	39.6	41.5	43.5
FGFI	R-101	39.8	40.7	42.4
	R-101 DCN	**40.5**	41.9	43.0
Ours	N/A	40.3	**42.1**	**44.4**

Table 3: Results corresponding to Figure 1. Our method is effective for stronger students compared with others.

We also compare the proposed **teacher-free** LGD with the classical **teacher-based** method, FGFI (Wang et al. 2019). Experiments are conducted on RetinaNet 2× ms with backbones R-50, 101 and 101 DCN respectively. As shown in Figure 1 and Table 3, our framework performs better when student gets stronger. Towards strong detector with R-101 DCN as backbone, LGD is **0.9**% and **1.4**% superior to LabelEnc and FGFI. The reason why the benefits of FGFI diminish might attribute to lack of much stronger teacher (Zhang and Ma 2021; Yao et al. 2021). We believe it is possible that FGFI with larger teacher or other stronger teacher-based detection KD can outperform ours, but such teacher-presumed setting is not the design purpose of our framework.

4.4 Ablation Studies

Method	AP	APs	APm	APL	ΔAP
N/A	36.6	21.2	40.4	48.1	–
MLP	37.9	21.5	41.9	49.7	+1.3
TransEnc	37.9	21.7	41.6	50.2	+1.3
PointNet	38.3	23.2	42.0	50.0	+1.7

Table 4: Label Encoder Ablation

Label Encoding. In this work, we adopt PointNet (Qi et al. 2017) as the label encoding module. In fact, other modules are also applicable. We conduct comparisons on three alternations under 2×ms schedule on MS-COCO with RetinaNet based on ResNet-50 backbone. Specifically, we compare PointNet with a MLP only network, and an encoder network composed of 6 scaled dot-product attention heads (Vaswani et al. 2017), abbreviated as "TransEnc". Similar to the handling we have done upon PointNet, we feed label descriptors into these networks to obtain label embeddings. We respectively input these label embeddings to remaining LGD modules and examine. All variants achieve good results as shown in Table 4, which demonstrates the robustness of our framework. The PointNet we finally adopt is the best among three of them, perhaps owing to its local-global relationship modeling among label descriptors.

Method	Baseline	Interaction Query (Ours)	
		Label	Student
RetinaNet	36.6	37.6 (+1.0)	**38.3** (+1.7)
FRCN	37.6	37.8 (+0.2)	**38.3** (+0.7)
FCOS	38.8	39.6 (+0.8)	**39.7** (+0.9)

Table 5: Inter-object Relation Adaption ablations with RetinaNet, Faster R-CNN and FCOS with R-50 1× ss.

Inter-object Relation Adapter. As aforementioned in Sec 3.2, the proposed method adopts the student appearance embeddings as queries and label embeddings as keys and values to involve in the guided inter-object relation modeling (here abbreviated as "Student"). We also experiment with the reverse option that using label embeddings as queries (abbreviated as "Label"). As shown in Table 5, for RetinaNet and FRCN 1× ss with R-50 as backbone, the adopted "student" mode are 0.7% and 0.5% better than "Label" mode.

Intra-object Knowledge Mapper. As specified in Equation 4, the instructive knowledge is dependent on interacted embeddings of both actual objects and virtual context. We ablate their usage in Table 6a. As expected, the context alone is not helpful since mere context provides nothing useful towards object detection. It manages to enhance the performance when combined with object embeddings (+0.3%).

Object	Context	AP
		36.6
	✓	36.6
✓		38.0
✓	✓	**38.3**

(a) Embedding Participation

Method	Mode	AP
RetinaNet	–	36.6
	unshared	37.8
	shared	**38.3**
FRCN	–	37.6
	unshared	37.7
	shared	**38.3**

(b) Head sharing choice

Table 6: Intra-object knowledge adapter ablations.

Head Sharing. Besides, we also examine the head sharing paradigm as shown in Table 6b. Sharing heads between student and instructive representations is consistently better.

4.5 Training Efficiency

Method	Pre-training	Overall	Method Specific
Baseline	–	12.1	–
FGFI	17.0	35.5	23.4
LabelEnc	14.9	24.5	12.4
Ours	N/A	**23.5**	**11.4**

Table 7: Comparison of Training Cost (hours).

Though all distillation and regularization methods won't affect the inference speed of student, they could be training-inefficient due to prerequisite pretraining and distillation process. This is concerned in practical applications but is seldom discussed. As shown in Table 7, we benchmark the (1) "Overall": overall training cost and (2) "Method Specific": overall except student learning (an inherent part shared by all methods). The examination is run on 8 Tesla V100 GPUs upon RetinaNet $2\times$ ss R-50. We use the corresponding detector with R-101 backbone as teacher for FGFI. Compared with FGFI, we save **34**% (23.5 *vs.* 35.5 hours) and **51**% (11.4 *vs.* 23.4 hours) on overall and method-specific items respectively. In fact, there could be stronger teacher exploitation for FGFI or other modern teacher-based KDs that outperform ours but it might bring about a heavier training burden and is beyond our discussion scope. Analogous to FGFI, LabelEnc introduces a two-stage training paradigm albeit without pretrained teacher. Towards LabelEnc, our method consumes 1 hour less and is trained in one-step fashion. In practice, LabelEnc consumes 3.8 G extra GPU footprints except that of the inherent detector, while ours consumes 2.5 G extra (saving **34**% relatively) yet performs better.

4.6 Versatility

Extended Datasets

(a) ***Pascal VOC***: We conduct experiments with Faster R-CNN and RetinaNet with R-50 under $2\times$ ms setting. As shown in Table 8, our method improves the results by **1.7**% (Faster R-CNN) and **2.3**% (RetinaNet). Notably, the AP75 metric of RetinaNet improves **3.0**%, showing the efficacy.

Method	AP	AP50	AP75
FRCN	55.1	81.9	61.0
+ours	56.8 (+**1.7**)	82.5 (+0.6)	63.3 (+**2.3**)
RetinaNet	56.6	81.4	61.3
+ours	58.9 (+**2.3**)	82.6 (+1.2)	64.3 (+**3.0**)

Table 8: Pascal VOC.

mMR: Method \ Detector	RetinaNet	FRCN
Baseline	57.9	48.7
Ours	56.4 (↑ **1.5**)	46.4 (↑ **2.3**)

Table 9: CrowdHuman. mMR: the lower, the better.

(b) ***CrowdHuman***: We also verify our method on the largest crowded detection dataset, CrowdHuman. As shown in Table 9, our method significantly improves the mMR (lower is better) by **2.3**% and **1.5**% for Faster R-CNN and RetinaNet respectively. It further demonstrates the generality of our proposed LGD method towards real-world applications.

Method	AP_{box}	AP_{mask}
Mask R-CNN (R-50)	38.8	35.2
+ours	39.8 (**+1.0**)	36.2 (**+1.0**)
Mask R-CNN (R-101)	41.2	37.2
+ours	42.0 (**+0.8**)	38.0 (**+0.8**)

Table 10: Comparison on instance segmentation.

Instance Segmentation. To further validate the versatility, we conduct experiments on instance segmentation in MS-COCO. In this task, a detector not only needs to localize each instance but also needs to predict a fine-grained foreground mask. We experiment on Mask R-CNN (He et al. 2017). To fully utilize the labels, we replace the object-wise box masks (Section 3.1 (2)) with the segmentation masks as better spatial prior. As shown in Table 10, our method boosts **1**% and **0.8**% mask-box AP with respect to Mask R-CNN R-50 and 101.

5 Conclusion

In this paper, we propose a brand new self-distillation framework, termed LGD for knowledge distillation in general object detection. It absorbs the spirits of inter-and-intra object relationship into forming the instructive knowledge given regular labels and student representations. The proposed LGD runs in an online manner with decent performance and relatively lower training cost. It is superior to previous teacher-free methods and a classical teacher-based KD method especially for strong student detectors, showing higher potential. We hope LGD could serve as a baseline for future detection KD methods without pretrained teacher.

Acknowledgements

This paper is supported by the National Key R&D Plan of the Ministry of Science and Technology (Project No. 2020AAA0104400) and Beijing Academy of Artificial Intelligence (BAAI). Also, the authors would like to thank Yuxuan Cai and Xiangwen Kong for the proof reading.

References

Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Cai, Q.; Pan, Y.; Ngo, C.; Tian, X.; Duan, L.; and Yao, T. 2019. Exploring Object Relation in Mean Teacher for Cross-Domain Detection. In *CVPR*.

Chen, D.; Mei, J.-P.; Wang, C.; Feng, Y.; and Chen, C. 2020a. Online knowledge distillation with diverse peers. In *AAAI*.

Chen, G.; Choi, W.; Yu, X.; Han, T. X.; and Chandraker, M. 2017. Learning Efficient Object Detection Models with Knowledge Distillation. In *NeurIPS*.

Chen, Y.; Zhang, Z.; Cao, Y.; Wang, L.; Lin, S.; and Hu, H. 2020b. RepPoints v2: Verification Meets Regression for Object Detection. In *NeurIPS*.

Dai, X.; Jiang, Z.; Wu, Z.; Bao, Y.; Wang, Z.; Liu, S.; and Zhou, E. 2021. General Instance Distillation for Object Detection. In *CVPR*.

Everingham, M.; Van Gool, L.; Williams, C. K. I.; Winn, J.; and Zisserman, A. 2010. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*.

Friedman, J.; Hastie, T.; Tibshirani, R.; et al. 2001. *The elements of statistical learning*, volume 1. Springer series in statistics New York.

Furlanello, T.; Lipton, Z. C.; Tschannen, M.; Itti, L.; and Anandkumar, A. 2018. Born-Again Neural Networks. In *ICML*.

Guo, J.; Han, K.; Wang, Y.; Wu, H.; Chen, X.; Xu, C.; and Xu, C. 2021. Distilling Object Detectors via Decoupled Features. In *CVPR*.

Guo, Q.; Wang, X.; Wu, Y.; Yu, Z.; Liang, D.; Hu, X.; and Luo, P. 2020. Online Knowledge Distillation via Collaborative Learning. In *CVPR*.

Hao, M.; Liu, Y.; Zhang, X.; and Sun, J. 2020. LabelEnc: A New Intermediate Supervision Method for Object Detection. In *ECCV*.

He, K.; Girshick, R. B.; and Dollár, P. 2019. Rethinking ImageNet Pre-Training. In *ICCV*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. B. 2017. Mask R-CNN. In *ICCV*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. arXiv:1503.02531.

Hoffman, J.; Gupta, S.; and Darrell, T. 2016. Learning with Side Information through Modality Hallucination. In *CVPR*.

Hu, H.; Gu, J.; Zhang, Z.; Dai, J.; and Wei, Y. 2018. Relation Networks for Object Detection. In *CVPR*.

Huang, Z.; Zou, Y.; Kumar, B. V. K. V.; and Huang, D. 2020. Comprehensive Attention Self-Distillation for Weakly-Supervised Object Detection. In *NeurIPS*.

Ioffe, S.; and Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*.

Jaderberg, M.; Simonyan, K.; Zisserman, A.; and Kavukcuoglu, K. 2015. Spatial Transformer Networks. In *NeurIPS*.

Kim, K.; Ji, B.; Yoon, D.; and Hwang, S. 2020. Self-knowledge distillation: A simple way for better generalization. *arXiv preprint arXiv:2006.12000*.

Lan, X.; Zhu, X.; and Gong, S. 2018. Knowledge Distillation by On-the-Fly Native Ensemble. In *NeurIPS*.

Law, H.; and Deng, J. 2018. CornerNet: Detecting Objects as Paired Keypoints. In *ECCV*.

Li, Q.; Jin, S.; and Yan, J. 2017. Mimicking Very Efficient Network for Object Detection. In *CVPR*.

Lin, T.; Dollár, P.; Girshick, R. B.; He, K.; Hariharan, B.; and Belongie, S. J. 2017a. Feature Pyramid Networks for Object Detection. In *CVPR*.

Lin, T.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2017b. Focal Loss for Dense Object Detection. In *ICCV*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Liu, B.; Rao, Y.; Lu, J.; Zhou, J.; and Hsieh, C.-J. 2020. Metadistiller: Network self-boosting via meta-learned top-down distillation. In *ECCV*.

Mostajabi, M.; Maire, M.; and Shakhnarovich, G. 2018. Regularizing Deep Networks by Modeling and Predicting Label Structure. In *CVPR*.

Park, W.; Kim, D.; Lu, Y.; and Cho, M. 2019. Relational Knowledge Distillation. In *CVPR*.

Peng, H.; Du, H.; Yu, H.; Li, Q.; Liao, J.; and Fu, J. 2020. Cream of the crop: Distilling prioritized paths for one-shot neural architecture search. In *NeurIPS*.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *CVPR*.

Ren, S.; He, K.; Girshick, R. B.; and Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NeurIPS*.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2015. FitNets: Hints for Thin Deep Nets. In *ICLR*.

Shao, S.; Zhao, Z.; Li, B.; Xiao, T.; Yu, G.; Zhang, X.; and Sun, J. 2018. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the Inception Architecture for Computer Vision. In *CVPR*.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. FCOS: Fully Convolutional One-Stage Object Detection. In *ICCV*.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *NeurIPS*.

Wang, T.; Yuan, L.; Zhang, X.; and Feng, J. 2019. Distilling Object Detectors With Fine-Grained Feature Imitation. In *CVPR*.

Wei, Y.; Pan, X.; Qin, H.; Ouyang, W.; and Yan, J. 2018. Quantization mimic: Towards very tiny cnn for object detection. In *ECCV*.

Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.-Y.; and Girshick, R. 2019. Detectron2. https://github.com/facebookresearch/detectron2.

Yang, C.; Xie, L.; Su, C.; and Yuille, A. L. 2019. Snapshot Distillation: Teacher-Student Optimization in One Generation. In *CVPR*.

Yang, J.; Li, C.; Zhang, P.; Dai, X.; Xiao, B.; Yuan, L.; and Gao, J. 2021. Focal self-attention for local-global interactions in vision transformers. *arXiv:2107.00641*.

Yao, L.; Pi, R.; Xu, H.; Zhang, W.; Li, Z.; and Zhang, T. 2021. G-DetKD: Towards General Distillation Framework for Object Detectors via Contrastive and Semantic-guided Feature Imitation. arXiv:2108.07482.

Yuan, L.; Tay, F. E. H.; Li, G.; Wang, T.; and Feng, J. 2020. Revisiting Knowledge Distillation via Label Smoothing Regularization. In *CVPR*.

Yun, S.; Park, J.; Lee, K.; and Shin, J. 2020. Regularizing Class-Wise Predictions via Self-Knowledge Distillation. In *CVPR*.

Zhang, L.; and Ma, K. 2021. Improve Object Detection with Feature-based Knowledge Distillation: Towards Accurate and Efficient Detectors. In *ICLR*.

Zhang, L.; Song, J.; Gao, A.; Chen, J.; Bao, C.; and Ma, K. 2019. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. In *ICCV*.

Zhang, Y.; Lan, Z.; Dai, Y.; Zeng, F.; Bai, Y.; Chang, J.; and Wei, Y. 2020. Prime-Aware Adaptive Distillation. In *ECCV*.

Zhang, Y.; Xiang, T.; Hospedales, T. M.; and Lu, H. 2018. Deep Mutual Learning. In *CVPR*.

Zhu, X.; Hu, H.; Lin, S.; and Dai, J. 2019. Deformable ConvNets V2: More Deformable, Better Results. In *CVPR*.