PKD: General Distillation Framework for Object Detectors via Pearson Correlation Coefficient

Weihan Cao[1] **Yifan Zhang**[1] * **Jianfei Gao**[2] **Anda Cheng**[1]
Ke Cheng[1] **Jian Cheng**[1]
Institute of Automation, Chinese Academy of Sciences[1]
Shanghai AI Laboratory[2]
caoweihan2020@ia.ac.cn
{yfzhang, jcheng}@nlpr.ia.ac.cn
gaojianfei@pjlab.org.cn
{chenganda, chengke}2017@ia.ac.cn

Abstract

Knowledge distillation(KD) is a widely-used technique to train compact models in object detection. However, there is still a lack of study on how to distill between heterogeneous detectors. In this paper, we empirically find that better FPN features from a heterogeneous teacher detector can help the student although their detection heads and label assignments are different. However, directly aligning the feature maps to distill detectors suffers from two problems. First, the difference in feature magnitude between the teacher and the student could enforce overly strict constraints on the student. Second, the FPN stages and channels with large feature magnitude from the teacher model could dominate the gradient of distillation loss, which will overwhelm the effects of other features in KD and introduce much noise. To address the above issues, we propose to imitate features with Pearson Correlation Coefficient to focus on the relational information from the teacher and relax constraints on the magnitude of the features. Our method consistently outperforms the existing detection KD methods and works for both homogeneous and heterogeneous student-teacher pairs. Furthermore, it converges faster. With a powerful MaskRCNN-Swin detector as the teacher, ResNet-50 based RetinaNet and FCOS achieve 41.5% and 43.9% mAP on COCO2017, which are 4.1% and 4.8% higher than the baseline, respectively. Our implementation is available at https://github.com/open-mmlab/mmrazor.

1 Introduction

Knowledge distillation(KD) is a widely-used technique to train compact models in object detection. However, there is still a lack of study on how to distill between heterogeneous detectors. Most previous works [38, 33, 9, 19, 12] rely on detector-specific designs and can only be applied to homogeneous detectors. [41, 46] conduct experiments on detectors with heterogeneous backbones, but detectors with heterogeneous detection heads and different label assignments are always omitted. Object detection is developing rapidly and algorithms with better performance are proposed continuously. Nevertheless, it is not easy to change detectors frequently in terms of stability in practical applications. Furthermore, in some scenarios, only detectors with a specific architecture can be deployed due to hardware limitations (e.g., two-stage detectors are hard to deploy), while most powerful teachers belong to different categories. Thus, it is promising if knowledge distillation can be conducted between heterogeneous detector pairs. In addition, current distillation methods, such as [41, 44],

*Corresponding author

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

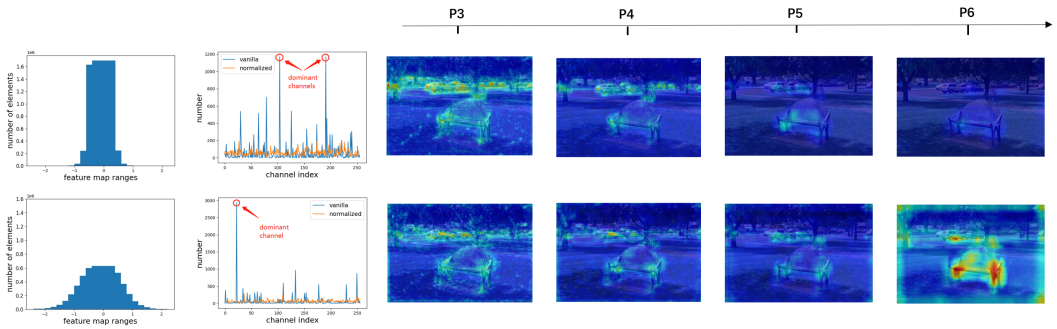

Figure 1: An example to illustrate the problems suffered by directly aligning the feature maps. Left: Feature magnitude of the teacher and the student. Middle: Dominant channels in FPN stage 'P3'. Let $\boldsymbol{s_{l,u,v}} \in \mathbb{R}^C$ denote the feature vector located in pixel (u, v) from l-th FPN stage and omit l for clarity. Then $number_i = \sum_{u,v} \mathbb{1}[\arg\max_c s_{u,v}^{(c)} = i]$ where i is the channel index. We define channels with a larger $number$ as dominant channels. Right: Visualization of the activation patterns from the teacher and the student. More visualization is provided in the Appendix.

usually introduce several complementary loss functions to further improve their performance, so several hyper-parameters are used to adjust the contribution of each loss function which heavily affects their abilities to transfer to other datasets.

In this paper, we first empirically verify that FPN feature imitation does make sense even if the student-teacher detector pairs are heterogeneous. However, directly minimizing the Mean Square Error (MSE) between features of the teacher and the student leads to sub-optimal results, with results shown in Table 4. Similar conclusions are drawn in [6, 12, 46, 41]. In order to explore the limitations of MSE, we elaborately visualize the FPN feature responses of the teacher and student detection models, as shown in Figure 1. Specifically, for an output feature $\boldsymbol{s_l} \in \mathbb{R}^{C\times H\times W}$ from l-th FPN stage, we select the maximum value in the dimension C at each pixel and obtain a 2-D matrix. Then we normalize the values to 0-255 according to the maximum and minimum values of l 2-D matrices. Through these comparisons, we obtain the following observations:

(1) The feature value magnitude of the teacher and the student is different, especially for heterogeneous detectors, as shown in Figure 1 (left). So directly aligning the feature maps between the teacher and the student may enforce overly strict constraints and do harm to the student.

(2) There are always several FPN stages whose values are larger than the others, as shown in Figure 1 (right). It is obvious that FPN stage 'P6' of the teacher is less activated than FPN stage 'P3'. However, for detectors such as RetinaNet [23] and FCOS [35], all FPN stages share the same detection head. Hence, FPN stages with larger values could dominate the gradient of the distillation loss, which will overwhelm the effects of other features in KD and lead to sub-optimal results.

(3) There are always a few channels whose values are significantly larger than the others, as shown in Figure 1 (middle). However, it is mentioned in [33, 45] that the less activated features are still practical for distillation. If they are not correctly balanced, the small gradients produced by these less activated channels can be drowned in the large gradients produced by dominant ones, thus limiting further refinement. Furthermore, from the first column of Figure 1 (right), we observe that there is much noise in the object-irrelevant area . To understand why, recall that we select the maximum value in the dimension C at each pixel, while values of several channels are larger than others. Noise in these channels is visualized. Hence, directly imitating the feature maps may introduce much noise.

According to these observations, we propose **K**nowledge **D**istillation via **P**earson Correlation Coefficient (PKD) shown in Figure 2, which focuses on the linear correlation between features of the teacher and the student. To remove the negative influences of magnitude difference, dominant FPN stages and channels, we first normalize the feature maps to have zero mean and unit variances and minimize the MSE loss between the normalized features. Mathematically, it is equivalent to firstly calculating the Pearson Correlation Coefficient (r) [2] between two original feature vectors, and then using $1 - r$ as the feature imitation loss. In addition to addressing the above three issues, PCC also has a beneficial effect on the gradient flow through student detectors by reducing the dependence of gradients on the feature value magnitude. So the training process is more stable.

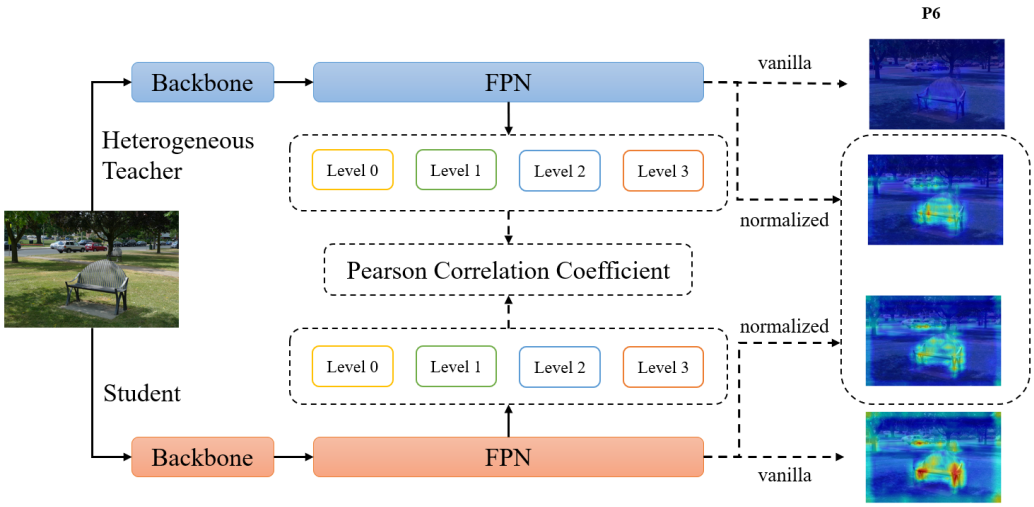

Figure 2: Overview of the proposed distillation via Pearson Correlation Coefficient (PKD). To demonstrate the effectiveness of PKD, we visualize the discord activation patterns of pre-normalized and post-normalized FPN features before distillation. The normalization mechanism bridge the gap between the activation patterns of the student and the teacher.

Compared with the previous methods, our method offers the following advantages. First, as the distillation loss is calculated just on FPN features, it can be easily applied to heterogeneous detector pairs, including models with heterogeneous backbones, heterogeneous detection heads and different training strategies such as label assignment. Second, as there is no need to forward the teacher's detection head, the training time can be reduced significantly, especially for those models with cascaded heads. Besides, PKD converges faster than previous methods. Last but not least, it is not sensitive to the only hyper-parameter - distillation loss weight. So it can be easily applied to other datasets. We conduct extensive experiments to verify the significant performance boosts brought by our approach on COCO dataset [24]. Using the same two-stage detector as the teacher, ResNet50 based RetinaNet [23] and FCOS [23] achieve 41.5% mAP and 43.9% mAP, which surpasses the baseline by 4.1% and 4.8% respectively, and it also outperforms the previous state-of-the-art methods by a large margin.

In summary, the contributions of this paper are threefold:

- We present that FPN feature imitation does make sense although the student-teacher detector pairs are heterogeneous.
- We propose to imitate FPN features with PCC to focus on relational information and give more freedom to the student feature's magnitude. It is capable of distilling knowledge for both homogeneous and heterogeneous detector pairs.
- We verify the effectiveness of our method on various detectors via extensive experiments on the COCO [24], and without bells or whistles achieve state-of-the-art performance. Moreover, our method converges faster and is not sensitive to the only hyper-parameter distillation loss weight. It is simple yet effective.

2 Related Works

2.1 Object Detection

Object detection is considered one of the most challenging vision tasks which aims at detecting semantic objects of a certain class in images. Modern detectors are roughly divided into two-stage detectors [31, 4, 13] and one-stage detectors [23, 35, 21, 11]. In two-stage detectors, a Region Proposal Network(RPN) is usually adopted to generate initial rough predictions refined by a task-specific detection head. A typical example is Faster R-CNN [31]. In contrast, One-stage detectors, such as RetinaNet [23] and FCOS [35], can directly and densely predict bounding boxes on the output feature maps. Among these works, multi-scale features are usually adopted to handle objects

of various scales, *e.g.* FPN [22], which is considered a typical case for our study. The proposed PKD only distills the intermediate features and does not rely on detector-specific designs so that it can be used directly on various detectors.

2.2 Knowledge Distillation

Knowledge Distillation (KD) is a kind of model compression and acceleration approach aiming at transferring knowledge from a teacher model to a student model. It is popularized by [15] and then its effectiveness in image classification has been explored by subsequent works [32, 16, 14, 1, 43, 28, 26, 34, 36]. However, adapting KD to object detectors is nontrivial since minimizing the Kullback–Leibler (KL) divergence between the classification head outputs fails to transfer the spatial information from the teacher and only brings limited performance gain to the student. The following three strategies are usually adopted by previous methods in detection to handle the above challenge. First, the distillation is usually conducted among multi-scale intermediate features [20], which provides rich spatial information for detection. Second, different feature selection methods are proposed to overcome the foreground-background imbalance. Most of these works can be divided into three kinds according to the feature selection approach [18]: proposal-based [6, 20, 9, 42], rule-based [38, 12] and attention-based [44, 46, 19]. Third, as the relation between different objects contains valuable information, many previous works try to improve the performance of detectors by enabling detectors to capture and make use of these relations, such as non-local modules [44] and global distillation [41].

Unlike the previous works, we consider the magnitude difference, dominant FPN stages and channels as the key problem. We hope our method could serve as a solid baseline and help ease future research in knowledge distillation for object detectors.

3 Method

3.1 Preliminaries

In this part, we briefly recap the traditional knowledge distillation for object detection. Recently, feature-based distillation over multi-scale features is adopted to deal with rich spatial information for detection. Different imitation masks S are proposed to form an attention mechanism for foreground features and filter away noises in the background. The objective can be formulated as:

$$\mathcal{L}_{FPN} = \sum_{l=1}^{L} \frac{1}{N_l} \sum_{c}^{C} \sum_{i}^{W} \sum_{j}^{H} S_{lcij} \left(F_{lcij}^{t} - \phi_{adapt} \left(F_{lcij}^{s} \right) \right)^2, \tag{1}$$

where L is the number of FPN layers, l represents the l-th FPN layer, i, j are the location of the corresponding feature map with width W and height H. $N_l = \sum_c^C \sum_i^W \sum_j^H S_{lcij}$. F_l^t and F_l^s are the l-th layer of feature of student and teacher detectors, respectively. Function ϕ_{adapt} is a 1x1 convolution layer to upsample the number of channels for the student network if the number of channels mismatches between the teacher and the student.

The definition of S is different in these methods. For example, FRS [46] uses the aggregated classification score map from the corresponding FPN layer, and FGD [41] considers spatial attention, channel attention, object size and foreground-background information simultaneously.

3.2 Is FPN feature imitation applicable for heterogeneous detector pairs?

Most of the previous works perform distillation on FPN, as FPN integrates multiple layers of the backbone and provides rich spatial information of multi-scale objects. It is reasonable to force the student to imitate FPN features from its homogeneous teacher as they have the same detection head and label assignment, and better features could lead to better performance. However, there is still a lack of study on how to distill between heterogeneous detectors. [41, 46] conduct experiments on detectors with heterogeneous backbones, but detectors with heterogeneous detection heads and different label assignments are always omitted. Thus, we are motivated to investigate whether FPN feature imitation still makes sense for these heterogeneous detector pairs.

We first simulate the extreme case of feature-based knowledge distillation via the following backbone and neck replacement. Without loss of generality, three three popular detectors, including RetinaNet [23], FCOS [35] and GFL [21] are adopted. Their backbones are the same, while the detection heads and training strategies such as label assignment are different. We first train these detectors for 12 epochs, then freeze the pre-trained backbones and necks, and fine-tune the randomly initialized detection heads for 12 epochs. During fine-tuning, the parameters in the backbone and neck are frozen and will not be updated. By analyzing the results in Table 1, we observe that distillation between heterogeneous detectors is possible, but feature magnitude difference needs to be eliminated first.

3.3 Feature Imitation with Pearson Correlation Coefficient

As discussed in Section 3.2, a promising feature distillation approach needs to consider the magnitude difference when constructing them into pairs for imitation. Moreover, by comparing the activation patterns shown in Figure 1, we find the dominant FPN stages and channels can negatively interfere with the training phase of the student and lead to sub-optimal results, which is ignored by previous works.

Table 1: Results of the backbone and neck replacement on COCO.

Backbone & Neck	Head	mAP
FCOS-Res50	FCOS	36.5
GFL-Res50	FCOS	37.6
FCOS-Res50	GFL	38.8
GFL-Res50	GFL	39.7
Retina-Res50	Retina	36.3
FCOS-Res50	Retina	35.2
Retina-Res50	FCOS	36.5
FCOS-Res50	FCOS	36.5

To address the above issues, we propose first normalizing the features of the teacher and the student to have zero mean and unit variances and minimizing the MSE between normalized features. Additionally, we want the normalization to obey the convolutional property - so that different elements of the same feature map, at different locations, are normalized in the same way. In other words, features convolved by the same convolution kernel are normalized in the same way. To accomplish this, we jointly normalize all the activations over all locations in a mini-batch. Let $\mathbb{B}$ be the set of all values in a feature map across both the elements of a mini-batch and spatial locations. So for a mini-batch of size b and feature maps of size $h \times w$, we use the effective mini-batch of size $m = \|\mathbb{B}\| = b \cdot hw$. Let $\boldsymbol{s}^{(\boldsymbol{c})} \in \mathbb{R}^m$ be the c-th channel of a batch of FPN outputs and omit c for clarity. Then we get the normalized values $\hat{s}_{1\ldots m}$ and $\hat{t}_{1\ldots m}$ from the student and the teacher, respectively. Instead of delicately designing imitation masks S in Eq. 1 to choose the important features, our PKD only operates on the full feature map. That is, the imitation mask S is filled with the scalar value 1. Hence we can formulate our distillation loss as the following:

$$\mathcal{L}_{FPN} = \frac{1}{2m} \sum_{i=1}^{m} (\hat{s}_i - \hat{t}_i)^2. \tag{2}$$

Moving forward, minimizing the loss function above is equivalent to maximizing the PCC between the pre-normalized features of the student and the teacher. PCC can be computed as:

$$r(\boldsymbol{s}, \boldsymbol{t}) = \frac{\sum_{i=1}^{m} (s_i - \mu_s)(t_i - \mu_t)}{\sqrt{\sum_{i=1}^{m} (s_i - \mu_s)^2} \sqrt{\sum_{i=1}^{m} (t_i - \mu_t)^2}}. \tag{3}$$

As $\hat{\boldsymbol{s}}, \hat{\boldsymbol{t}} \sim \mathcal{N}(0, 1)$, we get $\frac{1}{m-1} \sum_i \hat{s}_i^2 = 1$, $\frac{1}{m-1} \sum_i \hat{t}_i^2 = 1$. Then, we can reformulate Eq. 2 as:

$$\begin{aligned} \mathcal{L}_{FPN} &= \frac{1}{2m} \left((2m - 2) - 2 \sum_{i=1}^{m} \hat{s}_i \hat{t}_i \right) \\ &= \frac{2m - 2}{2m} (1 - r(\boldsymbol{s}, \boldsymbol{t})) \approx 1 - r(\boldsymbol{s}, \boldsymbol{t}). \end{aligned} \tag{4}$$

The gradient of PCC, $\partial \mathcal{L} / \partial s_i$, with respect to each FPN output s_i, is given by:

$$\frac{\partial \, \mathcal{L}_{FPN}}{\partial s_i} = \frac{1}{m \sigma_s} (\hat{s}_i \cdot r(\boldsymbol{s}, \boldsymbol{t}) - \hat{t}_i), \tag{5}$$

recalling $r(\boldsymbol{s}, \boldsymbol{t})$ is PCC in Eq. 3 and σ_s is the sample standard deviation of the student's feature $\boldsymbol{s}$. So PCC also has a beneficial effect on the gradient flow through student detectors by reducing the dependence of gradients on the feature value magnitude. It makes gradient descent-based training algorithms more stable.

In conclusion, PCC focuses on the relational information from the teacher and gives more freedom to the student feature's magnitude. Moreover, it eliminates the negative impacts of the dominant FPN stages and channels, leading to better performance. As a result, the normalization mechanism bridge the gap between the activation patterns of the student and the teacher (see Figure 2). Hence, feature imitation with PCC works for heterogeneous detector pairs. We train the student detector with the total loss as follows:

$$\mathcal{L} = \mathcal{L}_{GT} + \alpha \mathcal{L}_{FPN}, \tag{6}$$

where $\mathcal{L}_{GT}$ is the detection training loss, α is the hyper-parameter to balance the detection training loss and distillation loss.

3.4 Connection of PCC and KL divergence

As discussed in Section 3.3, the normalization mechanism is the key to addressing the three issues mentioned above. At the same time, KL divergence has been widely used in distillation. They first convert activations into a probability distribution with softmax function and then minimize the asymmetry KL divergence of the normalized activation maps:

$$\mathcal{L}_{KL} = T^2 \sum_{i=1}^{m} \phi(t_i) \cdot log[\frac{\phi(t_i)}{\phi(s_i)}], \tag{7}$$

where T is a hyper-parameter to control the degree of softness of the targets, and ϕ is the softmax function:

$$\phi(t) = \frac{exp(t_i/T)}{\sum_{j=1}^{m} exp(t_j/T)}. \tag{8}$$

Here, we show that minimizing KL divergence between normalized features in the high-temperature limit is equivalent to minimizing MSE between normalized ones, and hence equivalent to maximizing PCC between original ones.

Let $p_i = \phi(\hat{t_i})$ and $q_i = \phi(\hat{s_i})$ denote the probabilities from the teacher and the student, respectively. The KL divergence gradient, $\partial \mathcal{L}_{KL} / \partial \hat{s_i}$, with respect to each normalized activation $\hat{s_i}$ of the student is given by:

$$\frac{\partial \, \mathcal{L}_{KL}}{\partial \hat{s_i}} = T\left(q_i - p_i\right) = T\left(\frac{e^{\hat{s_i}/T}}{\sum_j e^{\hat{s_j}/T}} - \frac{e^{\hat{t_i}/T}}{\sum_j e^{\hat{t_j}/T}}\right). \tag{9}$$

If T is high compared with the magnitude of the normalized activations, we can approximate:

$$\frac{\partial \mathcal{L}_{KL}}{\partial \hat{s_i}} \approx T\left(\frac{1 + \hat{s_i}/T}{N + \sum_j \hat{s_j}/T} - \frac{1 + \hat{t_i}/T}{N + \sum_j \hat{t_j}/T}\right). \tag{10}$$

As $\hat{s_i}$ and $\hat{t_i}$ have been zero-meaned, Eq. 10 simplifies to:

$$\frac{\partial \mathcal{L}_{KL}}{\partial \hat{s_i}} \approx \frac{1}{N}\left(\hat{s_i} - \hat{t_i}\right). \tag{11}$$

Exploiting the relation between MSE and PCC shown in Eq. 4 gives the desired result. Experimental results are provided in Section A.3.

4 Experiments

In order to verify the effectiveness and robustness of our method, we conduct experiments on different detection frameworks on the COCO [24] dataset. We choose the default 120k train images split for training and 5k val images split for the test. We use the standard training settings following [41] and report mean Average Precision (AP) as an evaluation metric, together with AP under different thresholds and scales, $i.e.$, AP_{50}, AP_{75}, AP_S, AP_M and AP_L.

For distillation, the hyper-parameter α is set to 6 when using a two-stage detector as the teacher and 10 when using a one-stage one. Feature levels of heterogeneous detector pairs may not be strictly

Table 2: Results of the proposed method with different detection frameworks on the COCO dataset. T and S mean the teacher and student detector, respectively. * indicates the results reproduced by us.

Method	schedule	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Retina-ResX101 (T)	2x	40.8	60.5	43.7	22.9	44.5	54.6
Retina-Res50 (S)	2x	37.4	56.7	39.6	20.0	40.7	49.7
FKD [44]	2x	39.6 (+2.2)	58.8	42.1	22.7	43.3	52.5
FRS [46]	2x	40.1 (+2.7)	59.5	42.5	21.9	43.7	54.3
FGD [41]	2x	40.4 (+3.0)	59.9	43.3	**23.4**	44.7	54.1
PKD (Ours)	2x	**40.8 (+3.4)**	**60.3**	**43.4**	23.0	**45.1**	**54.7**
FasterRCNN-Res101 (T)	2x	39.8	60.1	43.3	22.5	43.6	52.8
FasterRCNN-Res50 (S)	2x	38.4	59.0	42.0	21.5	42.1	50.3
GID [9]	2x	40.2 (+1.8)	60.7	43.8	22.7	44.0	53.2
FRS [46]	2x	40.4 (+2.0)	60.8	44.0	**23.2**	44.4	53.1
FGD [41]	2x	40.4 (+2.0)	60.7	44.3	22.8	44.5	**53.5**
PKD (Ours)	2x	**40.5 (+2.1)**	**60.9**	**44.4**	22.6	**44.8**	53.1
FCOS-Res101 (T)	2x+ms	41.2	60.4	44.2	24.7	45.3	52.7
FCOS-Res50 (S)	2x+ms	39.1	58.4	41.6	24.0	42.7	48.7
FRS [46] *	2x+ms	42.2 (+3.1)	60.6	45.6	**27.1**	46.5	53.0
FGD [41] *	2x+ms	42.3 (+3.2)	60.8	45.8	26.1	46.7	53.3
PKD (Ours)	2x+ms	**42.8 (+3.7)**	**61.4**	**46.2**	25.9	**47.2**	**54.6**
RepPoints ResNeXt101 (T)	2x	44.2	65.5	47.8	26.2	48.4	58.5
RepPoints Res50 (S)	2x	38.6	59.6	41.6	22.5	42.2	50.4
FGD [41]	2x	41.3 (+2.7)	-	-	**24.5**	45.2	54.0
PKD (Ours)	2x	**42.3 (+3.7)**	63.1	45.4	23.9	**46.6**	**56.5**
TOOD-ResX101 (T)	2x+ms	47.6	68.5	51.6	30.6	51.4	59.7
TOOD-Res50 (S)	1x	42.4	59.7	46.2	25.4	45.5	55.7
PKD (Ours)	1x	45.5 (+3.1)	62.8	49.3	27.4	49.8	58.4

aligned, *e.g.*, FasterRCNN constructs the feature pyramid from $P2$ to $P6$, while RetinaNet uses $P3$ to $P7$. To address the above issue, we upsample the low-resolution feature maps to have the same spatial size as the high-resolution ones. In addition, the adaptive layer (ϕ_{adapt} in Eq. 1) is unnecessary for our PKD.

All experiments are performed on 8 Tesla A100 GPUs with two images in each. Our implementation is based on mmdetection [7] and mmrazor [8] with Pytorch framework [29]. '1x' (namely 12epochs), '2x' (namely 24 epochs) and '2x+ms' (namely 24epochs with multi-scale training) training schedules are used. More details are given in the Appendix.

4.1 Main Results

Our method can be applied to different detection frameworks easily, so we first conduct experiments on five popular detectors, including a two-stage detector (Faster RCNN[31]), two anchor-based one-stage detector (RetinaNet [23], RepPoints [40] and TOOD [11]) and an anchor-free detector (FCOS [35]). Table 2 shows the comparison of the results of state-of-the-art distillation methods on the COCO. Our distillation method surpasses other state-of-the-art methods. All the student detectors gain significant improvements in AP with the knowledge transferred from teacher detectors, *e.g.*, FCOS with ResNet-50 gets a 3.7% mAP improvement on the COCO dataset. These results indicate the effectiveness and generality of our method in both one-stage and two-stage detectors.

4.2 Distilling More Student Detectors with Stronger Heterogeneous Teachers

Most of the current methods are designed for homogeneous detector pairs. PKD is general enough to distill knowledge between both homogeneous and heterogeneous detector pairs. Here we conduct experiments on more detectors and use stronger heterogeneous teacher detectors, as shown in Table 3.

Comparing with Table 2, we find that student detectors perform better with stronger teacher detectors, *e.g.*, Retina-Res50 model achieves 41.5% and 39.6% mAP with Mask RCNN-Swin [27] and Retina-Res101, respectively. Results show that mimicking the feature maps of stronger heterogeneous teacher detectors can further boost the student's performance when applied with our PKD.

Table 3: Results of more detectors with stronger heterogeneous teacher detectors on the COCO dataset.

Method	schedule	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask RCNN-Swin (T)	3x+ms	48.2	69.8	52.8	32.1	51.8	62.7
Retina-Res50 (S)	2x	37.4	56.7	39.6	20.0	40.7	49.7
PKD (Ours)	2x	41.5 (+4.1)	60.6	44.1	22.9	45.2	56.4
Mask RCNN-Swin (T)	3x+ms	48.2	69.8	52.8	32.1	51.8	62.7
FCOS-Res50 (S)	2x+ms	39.1	58.4	41.6	24.0	42.7	48.7
PKD (Ours)	2x+ms	43.9 (+4.8)	62.3	47.5	27.2	48.0	57.1
GFL-Res101 (T)	2x+ms	44.9	63.1	49.0	28.0	49.1	57.2
FCOS-Res50 (S)	2x+ms	39.1	58.4	41.6	24.0	42.7	48.7
PKD (Ours)	2x+ms	43.5 (+4.4)	61.9	47.1	26.5	47.7	55.7

Table 4: Comparisons between feature imitation with MSE and our PKD. We address the above three issues and achieve better performance.

Detector Pairs	Methods	mAP	AP_S	AP_L
FCOS-ResX101 Retina-Res50	MSE	36.3	20.0	47.1
	Ours	41.3	24.2	55.4
GFL-Res101 FCOS-Res50	MSE	43.0	27.1	54.3
	Ours	43.5	26.5	55.7
Retina-ResX101 Retina-Res50	MSE	40.4	22.2	54.4
	Ours	40.8	23.0	54.7

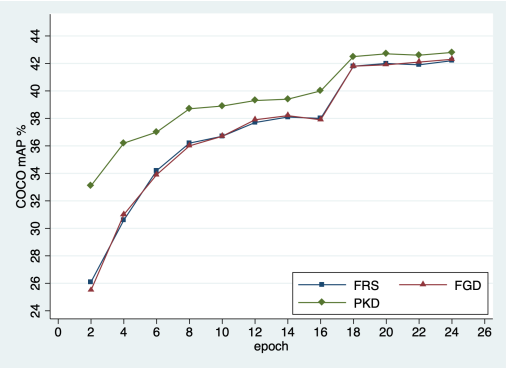

Figure 3: Comparison of the convergence speed of FCOS-Res50 among other state-of-the-art distillation methods and ours.

4.3 Other feature-based distillation methods with the normalized features

In object detection, some feature-based distillation methods [46, 41] transfer the knowledge within some pre-defined imitation regions, and achieved competitive results. While we believe full map feature imitation with our PKD can already outperform them. In order to show the universality of our method, we take FitNet [32], GT Mask, FRS [46] and FGD [41] as baselines and build PKD on them.

Recaps. Full feature map distillation with MSE loss is used in FitNet. While the 2nd method only imitates features that overlap with ground truth bounding boxes. Feature imitation and logit mimicking with FRS uses the aggregated classification score, takes maximum operation in channel direction, of the classification head output as a weighted score mask, to guide the distillation of the FPN and classification head. And Focal distillation in FGD forces the student to focus on the teacher's pixels and channels with large absolute mean values. Global distillation in FGD rebuilds the relation between different pixels and transfers it from teachers to students through GcBlock [5].

Results. For a fair comparison, all the hyper-parameters are copied from the original paper. Results in Table 5 show that combining PKD with these feature imitation methods can further improve their performance. Besides, we do not combine PKD with FGD focal distillation as focal distillation forces the student to focus on the teacher's pixels and channels with large absolute mean values. That is, features with larger values are given larger weights, which seems not to agree with the argument that

"The small gradients produced by less activated features can be drowned into the large gradients produced by highly activated ones, thus limiting further refinement." in [33, 45] and us.

Table 5: Results of other feature-based distillation methods with the normalized feature maps.

FitNet	GT Mask	FRS feature imitation	FGD Focal	FGD Global	PKD	GFL101 (44.7) GFL50 (40.2)
✓						41.3
✓					✓	43.3
	✓					41.4
	✓				✓	43.0
		✓				43.2
		✓			✓	43.4
			✓			41.8
				✓		40.2
				✓	✓	43.3

4.4 Analysis

4.4.1 Effectiveness of Pearson Correlation Coefficient

Among most previous works, feature-based distillation over multi-scale features is adopted for distillation. In this study, we argue that the magnitude difference, dominant FPN stages and channels can negatively interfere with the training phase of the student and lead to sub-optimal results. To study this empirically, we conduct the following three experiments to explore the effects of MSE loss which is a widely-used loss function in distillation and suffers from the above three issues simultaneously. For a fair comparison, we tune the loss weight of MSE for all experiments. More details are listed in Section A.4.

For the first pair, FCOS-ResX101 is used as the teacher whose feature magnitude is significantly different from that of the student Retina-Res50 (refer to Fig. 4). In this case, what constitutes the knowledge is better presented by relational information from the teacher's features than the absolute values. For the second pair, GFL-Res101 is the teacher and FCOS-Res50 is the student. For both teacher and student, features in FPN stages 'P5' and 'P6' are less activated than those in stages 'P3' and 'P4' (refer to Fig. 5). Hence, features in stages 'P3' and 'P4' could dominate the gradient of the distillation loss, which will overwhelm the effects of other features. Since larger objects are usually assigned to higher feature levels, the student's performance for large objects is significantly lower than that of ours when MSE is used as the distillation loss. For the last pair, Retina-ResX101 is the teacher and Retina-Res50 is the student. There are always a few channels with greater values (refer to Fig. 6 and Fig. 7) , so directly imitating the feature maps may introduce much noise in dominant channels.

By comparing the results in Table 4, we find that our proposed PKD addresses the above three issues and achieves better performance. Hence, an effective distillation loss function should have the ability to handle the above three problems. We hope PKD could serve as a solid baseline and help ease future research in knowledge distillation community.

4.4.2 Sensitivity study of loss weight α

In Eq. 6, we use the loss weight hyper-parameter α to balance the detection training loss and distillation loss. Here, we conduct several experiments to investigate the influence of α. As shown in Table 6, the worst result is just a 0.3 mAP drop compared with the best result, indicating our method is not sensitive to the only hyper-parameter α. Hence it is easy for practical use. Moving forward, as the normalization mechanism has a beneficial effect on the gradient flow through student detectors, by reducing the dependence of gradients on the feature value magnitude, we can train different networks with relatively uniform hyper-parameters α.

Table 6: Ablation study of loss weight hyper-parameter α on FCOS ResX101 - RetinaNet Res50.

α	3	5	8	10	13
mAP	41.0	41.1	41.1	41.3	41.1

4.4.3 Convergence Speed

In this subsection, we conduct experiments with FCOS to compare the convergence speed of our method with other state-of-the-art distillation methods on the COCO benchmark. As shown in Fig. 3, the training convergence can significantly speed up in the early training stage with our PKD. Meanwhile, the final performance of the student detector FCOS-Res50 is about 0.5%-0.6% mAP higher than that of FGD and FRS. In addition, there is no need to forward the detection head of the teacher, which reduces the training time significantly, especially for those models with cascaded heads.

4.4.4 Mathematical properties of Pearson Correlation Coefficient

The Pearson correlation coefficient is essentially a normalized measurement of the covariance, such that the result always has a value between -1 and 1. Hence, $L_{FPN} = 1 - r$ in Eq. 4 always has a value between 0 and 2. It focuses on the linear correlation between the features of the teacher and the student, and relaxes constraints on the magnitude of the features. Let $s^c, t^c \in R^{bhw}$ denote the c-th channel of the student's and teacher's features, and omit c for clarification. Then we get bhw data points (s_i, t_i). $L_{FPN} = 0$ implies that all data points lie on a line for which s increases as t increases. Hence the student is well-trained. And vice versa for $L_{FPN} = 2$. A value of 1 implies that there is no linear dependency between the features of the student and the teacher.

5 Conclusion

This paper empirically finds that FPN feature imitation is applicable for heterogeneous detector pairs although their detection heads and label assignments are different. Then we propose feature imitation with Pearson Correlation Coefficient to focus on the relational information from the teacher and give more freedom to the student's feature value magnitude. Furthermore, a general KD framework is proposed, capable of distilling knowledge for both homogeneous and heterogeneous detector pairs. It converges faster and only introduces one hyper-parameter, which can easily be applied to other datasets. However, our understanding of whether our proposed PKD is capable of other tasks such as text recognition is preliminary and left as future works.

Acknowledgments

This work is supported in part by the Shanghai Committee of Science and Technology, China (Grant No. 20DZ1100800), NSFC 62273347, the National Key Research and Development Program of China (2020AAA0103402), Jiangsu Key Research and Development Plan (No.BE2021012-2) and NSFC 61876182.

References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9163–9171, 2019.

[2] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.

[3] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: high quality object detection and instance segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1483–1498, 2019.

[5] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0, 2019.

[6] 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.

[7] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[8] MMRazor Contributors. Openmmlab model compression toolbox and benchmark. `https://github.com/open-mmlab/mmrazor`, 2021.

[9] Xing Dai, Zeren Jiang, Zhao Wu, Yiping Bao, Zhicheng Wang, Si Liu, and Erjin Zhou. General instance distillation for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2021.

[10] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6569–6578, 2019.

[11] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R Scott, and Weilin Huang. Tood: Task-aligned one-stage object detection. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3490–3499. IEEE Computer Society, 2021.

[12] 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.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[14] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3779–3787, 2019.

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

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

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[18] 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, 2021.

[19] Gang Li, Xiang Li, Yujie Wang, Shanshan Zhang, Yichao Wu, and Ding Liang. Knowledge distillation for object detection via rank mimicking and prediction-guided feature imitation. *arXiv preprint arXiv:2112.04840*, 2021.

[20] 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.

[21] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020.

[22] 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.

[23] 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.

[24] 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. Springer, 2014.

[25] 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.

[26] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. Knowledge distillation via instance relationship graph. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7096–7104, 2019.

[27] 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. *arXiv preprint arXiv:2103.14030*, 2021.

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

[29] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[30] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[31] 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.

[32] 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.

[33] Ruoyu Sun, Fuhui Tang, Xiaopeng Zhang, Hongkai Xiong, and Qi Tian. Distilling object detectors with task adaptive regularization. *arXiv preprint arXiv:2006.13108*, 2020.

[34] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

[35] 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.

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

[37] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[38] 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.

[39] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[40] 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.

[41] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. *arXiv preprint arXiv:2111.11837*, 2021.

[42] 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.

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

[44] 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.

[45] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. *arXiv preprint arXiv:2203.08679*, 2022.

[46] Du Zhixing, Rui Zhang, Ming Chang, Shaoli Liu, Tianshi Chen, Yunji Chen, et al. Distilling object detectors with feature richness. *Advances in Neural Information Processing Systems*, 34, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 3.3 and Section 3.4
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] To the best of our knowledge, there is no potential negative societal impacts of our work.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3.3 and Section 3.4
   (b) Did you include complete proofs of all theoretical results? [Yes] See Section 3.3 and Section 3.4
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Codes and instructions are provided in the Appendix. All datasets used are public datasets.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4. More details are given in the Appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We follow the common practice used in our baseline methods, where no error bars are reported.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Details are given in the Appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] All datasets used are public datasets.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

A.1 Motivation to build a general KD framework

Modern detectors are roughly divided into two-stage detectors [31, 13, 3] and dense prediction detectors (*e.g.*, anchor-based one-stage detectors [23, 25, 30] and anchor-free one-stage detectors [10, 35, 40]). Each family has its own advantages and weakness. In particular, two-stage detectors usually have higher performance, while being slower in inference speed and harder to be deployed due to the region proposal network (RPN) and RCNN head. On the other hand, dense prediction detectors are faster than two-stage detectors while being less accurate. In practice, it is a natural idea to use two-stage detectors as teachers to enhance dense prediction detectors.

Moreover, knowledge distillation between heterogeneous dense prediction detector pairs is also promising. In some scenarios, only the detectors with a specific architecture can be deployed due to hardware limitations. For example, compared with Batch Normalization [17] and Instance Normalization [37], Group Normalization [39] is hard to deploy. However, the most powerful teachers may belong to different categories. Furthermore, object detection is developing rapidly and algorithms with better performance are proposed continuously. Nevertheless, it is not easy to change detectors frequently in terms of stability and hardware runtime limitations in practical applications. So it is helpful if knowledge distillation can be conducted between the latest high-capacity detectors and the widely-used compact detectors.

Hence, we are motivated to design a general distillation method capable of distilling knowledge between both homogeneous and heterogeneous detector pairs.

A.2 Details of Training Recipe

We conduct experiments on different detection frameworks, including two-stage models, anchor-based one-stage models and anchor-free one-stage models. [18] proposes inheriting strategy which initializes the student with the teacher's neck and head parameters and gets better results. Here we use this strategy to initialize the student which has the same head structure as the teacher and find that it helps students converge faster.

All experiments are performed on 8 Tesla A100 GPUs with 2 images in each. Our implementation is based on mmdetection [7] and mmrazor [8] with Pytorch framework [29]. '1x' (namely 12 epochs), '2x' (namely 24 epochs) and '2x+ms' (namely 24 epochs with multi-scale training) training schedules with SGD optimizer are used. Momentum and weight decay are set to 0.9 and 1e-4. The initial learning rate is set to 0.02 for Faster RCNN and 0.01 for others. We train FCOS [35] with tricks including GIoULoss, norm-on-bbox and center-sampling which is the same as FGD [41] and GID [9]. For distillation, only one hyper-parameter α is introduced to balance the supervised learning loss and distillation loss, and it is set to 6 when using a two-stage detector as the teacher and 10 when using a one-stage one as the teacher. The teacher network is well-trained previously and fixed during training.

A.3 Connection of PCC and KL divergence

Section 3.4 in the main text shows the connection between PCC and KL divergence. We conduct two experiments on RetinaNet and GFL to verify this empirically. For both experiments, we set loss weight $\alpha = 10$ and temperature $T = 50$. As shown in Table 7, minimizing KL divergence between post-normalized features in the high-temperature limit can also achieve similar results.

A.4 Details of feature imitation with MSE

As shown in Table 4 in the main text, we compare the results of MSE with our proposed PKD. As the gradient of MSE loss depends on feature value magnitude and it is usually different among different detectors, we have to tune the loss weight carefully to achieve relatively good results, as shown in Table 8. And we put the best results down in Table 4.

Table 7: Results of the KL divergence with normalization mechanism.

Method	schedule	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Retina-ResX101 (T)	2x	40.8	60.5	43.7	22.9	44.5	54.6
Retina-Res50 (S)	2x	37.4	56.7	39.6	20.0	40.7	49.7
Norm+KL	2x	40.9	60.3	43.6	22.9	45.2	55.1
GFL-Res101 (T)	2x+ms	44.9	63.1	49.0	28.0	49.1	57.2
GFL-Res50 (S)	1x	40.2	58.4	43.3	23.3	44.0	52.2
Norm+KL	1x	43.1	60.9	46.7	25.1	47.8	55.9

Table 8: Results of MSE on various detector pairs.

Teacher	Student	Schedule	Baseline	Loss Weight	mAP
FCOS-ResX101	Retina-Res50	1x	36.5	5	33.9
				10	31.4
				20	29.7
FCOS-ResX101	Retina-Res50	2x	37.4	5	36.3
				10	35.6
				20	34.9
GFL-Res101	FCOS-Res50	1x	36.6	50	38.7
				70	39.2
GFL-Res101	FCOS-Res50	2x	39.1	10	41.5
				30	42.5
				50	42.7
				70	43.0
				80	42.9
Retina-ResX101	Retina-Res50	2x	37.4	5	40.0
				10	40.4
				15	40.3
MaskRCNN-Swin	FasterRCNN-Res50	2x	38.4	3	41.7
				5	41.7
				6	41.6
				10	41.7

A.5 Effectiveness of Pearson Correlation Coefficient

In this paper, we argue that the magnitude difference, dominant FPN stages and channels can negatively interfere with the training phase of the student. To clearly show that these three issues do exist, we elaborately visualize the FPN feature responses of the teacher and student detectors before distillation, as shown in Figure 4, Figure 5, Figure 6 and Figure 7. We follow the visualization method in Section 1.

Through these comparisons, we obtain the following three observations:

(1) The feature value magnitude of different detectors could be significantly different, especially for heterogeneous detectors, as shown in Figure 4. Directly aligning the feature maps between the teacher and the student may enforce overly strict constraints and do harm to the student (see Table 4 top in the main text).

(2) As shown in Figure 5, compared with features in FPN stage 'P3' and 'P4', features in stage 'P5' and 'P6' are less activated in both GFL-Res101 and FCOS-Res50. And it is a common case among different detector pairs. FPN stages with larger values could dominate the gradient of the distillation loss, which will overwhelm the effects of other features in KD.

(3) As shown in Figure 6 (right), the feature magnitude of the 210-th channel of MaskRCNN-Swin FPN stage 'P6' is significantly larger than others. Similar phenomena also exist in other detectors such as RetinaNet (see Figure 7). The small gradients produced by those less activated channels can be drowned in the large gradients produced by dominant ones, thus limiting further refinement. Furthermore, there is much noise in the object-irrelevant area, as depicted in Figure 6 (left). Directly imitating the feature maps may introduce much noise.

Comparing Table 8 and Table 4 in the main text, we find that our proposed PKD addresses the above three issues and achieves better performance. Hence, an effective distillation loss function should have the ability to handle the above three problems. We hope PKD could serve as a solid baseline and help ease future research in knowledge distillation community.

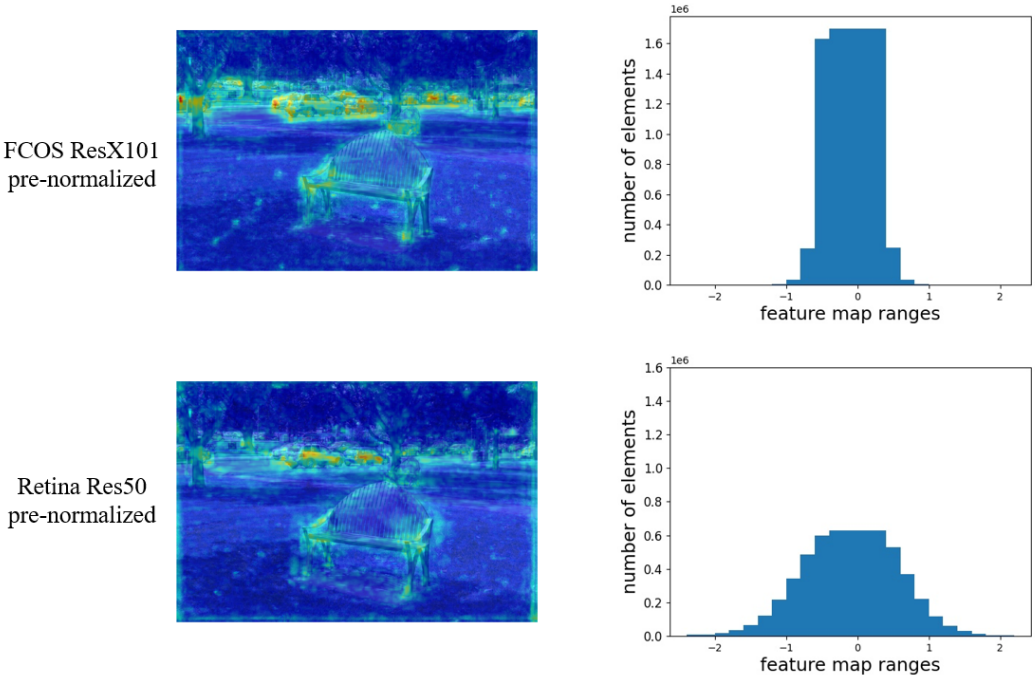

Figure 4: Visualization of the activation patterns and activation distribution of FPN stage 'P3'.

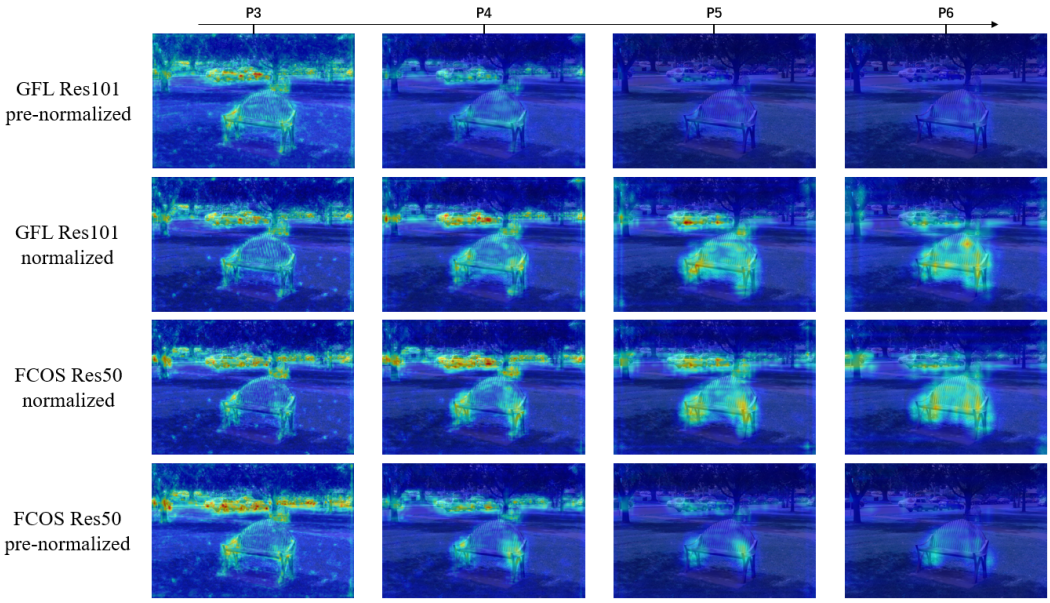

Figure 5: Visualization of dominant FPN stages. From left to right, they correspond to the activation patterns in FPN stage 'P3' to 'P6'. The leftmost corresponds to the lowest stage of FPN, and the rightmost corresponds to the highest stage of FPN.

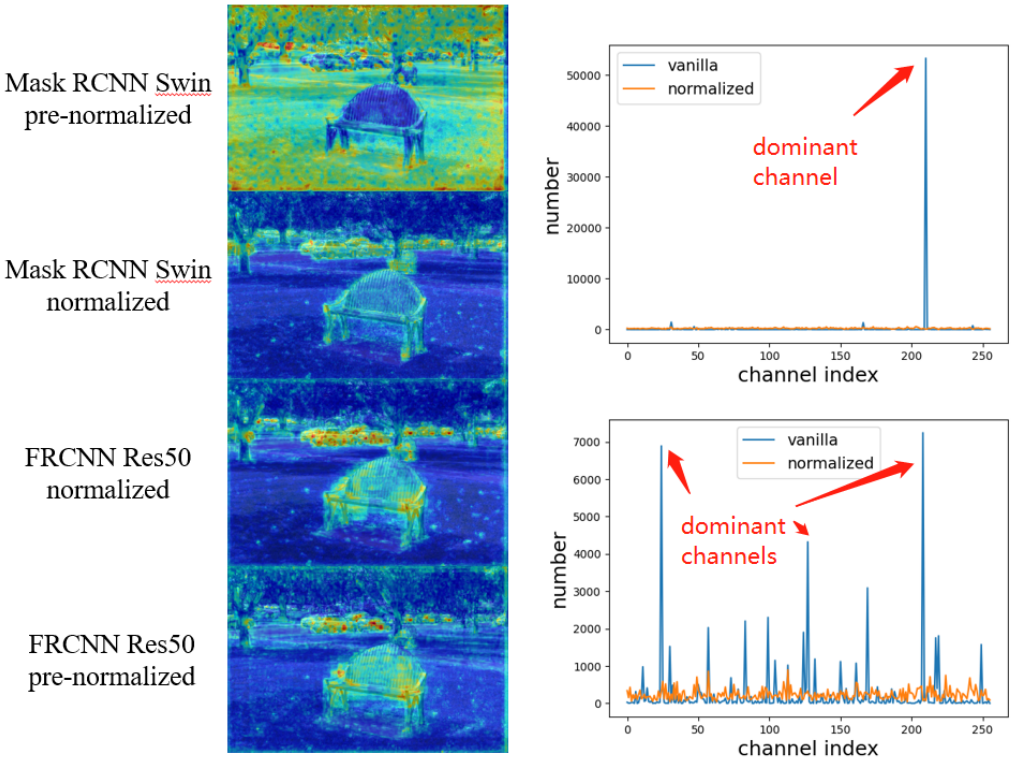

Figure 6: Visualization of dominant channels in MaskRCNN-Swin and FasterRCNN-Res50. Left: Visualization of the activation patterns of FPN stage 'P3'. Right: Dominant channels in pre-normalized FPN stage 'P3'. Let $\boldsymbol{s_{l,u,v}} \in \mathbb{R}^C$ denote the feature vector located in pixel (u, v) from l-th FPN stage and omit l for clarity. Then $number_i = \sum_{u,v} \mathbb{1}[\arg\max_c s_{u,v}^{(c)} = i]$ where i is the channel index. We define channels with a larger $number$ as dominant channels.

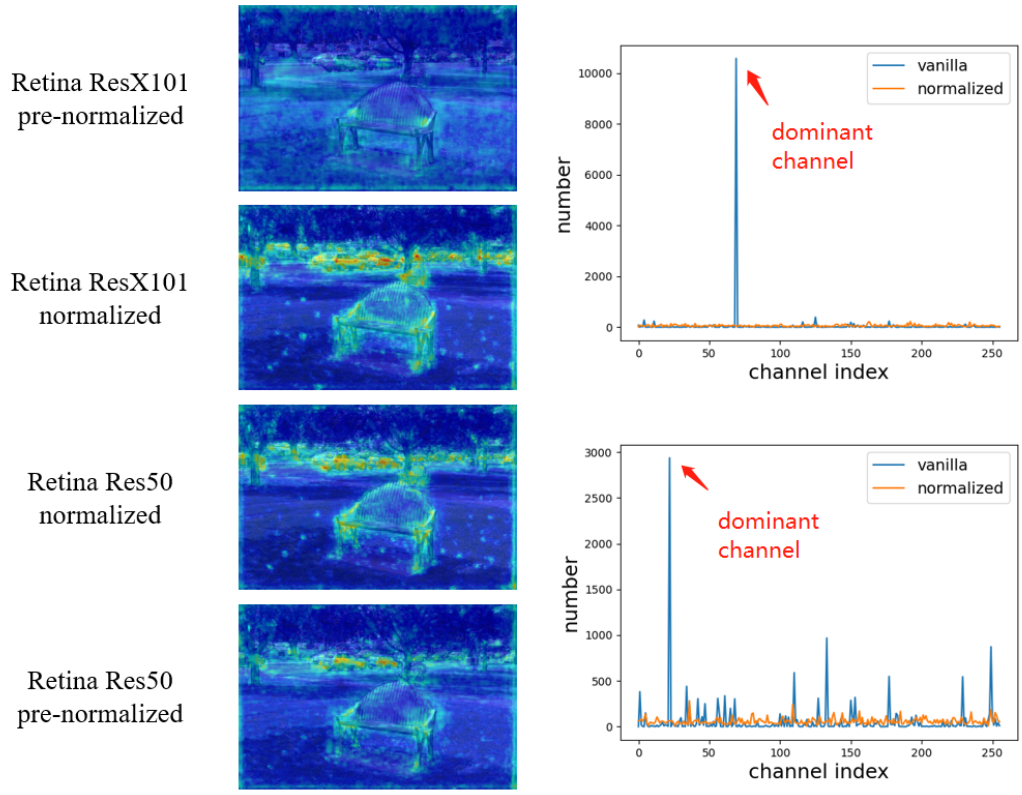

Figure 7: Visualization of dominant channels in Retina-ResX101 and Retina-Res50.

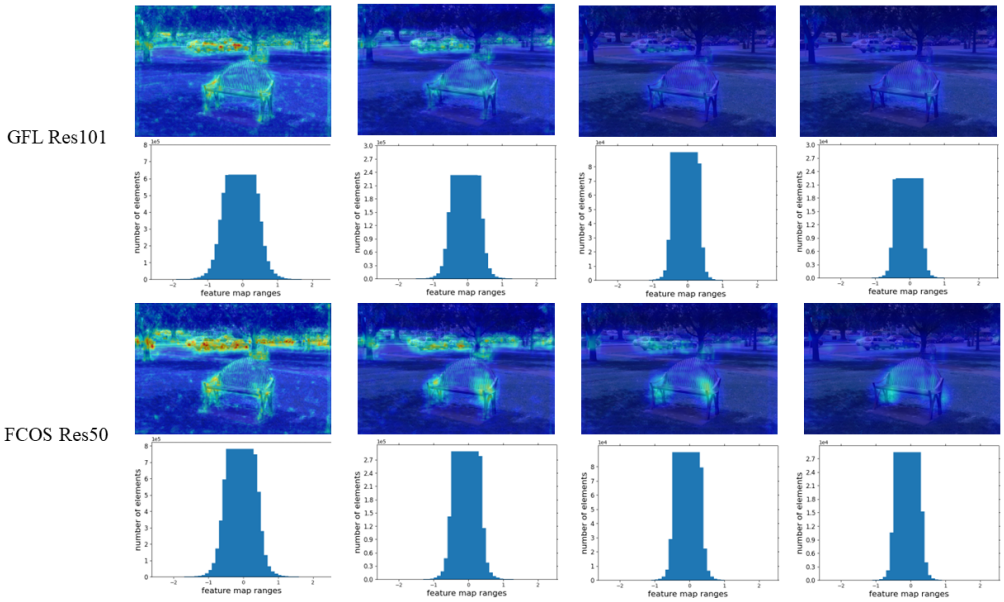

Figure 8: Visualization of the activation patterns and activation distribution of GFL-Res101 and FCOS-Res50. From left to right, they correspond to the activation patterns and activation distribution in FPN stage 'P3' to 'P6'. The leftmost corresponds to the lowest stage of FPN, and the rightmost corresponds to the highest stage of FPN. The feature value magnitude of GFL and FCOS is similar.